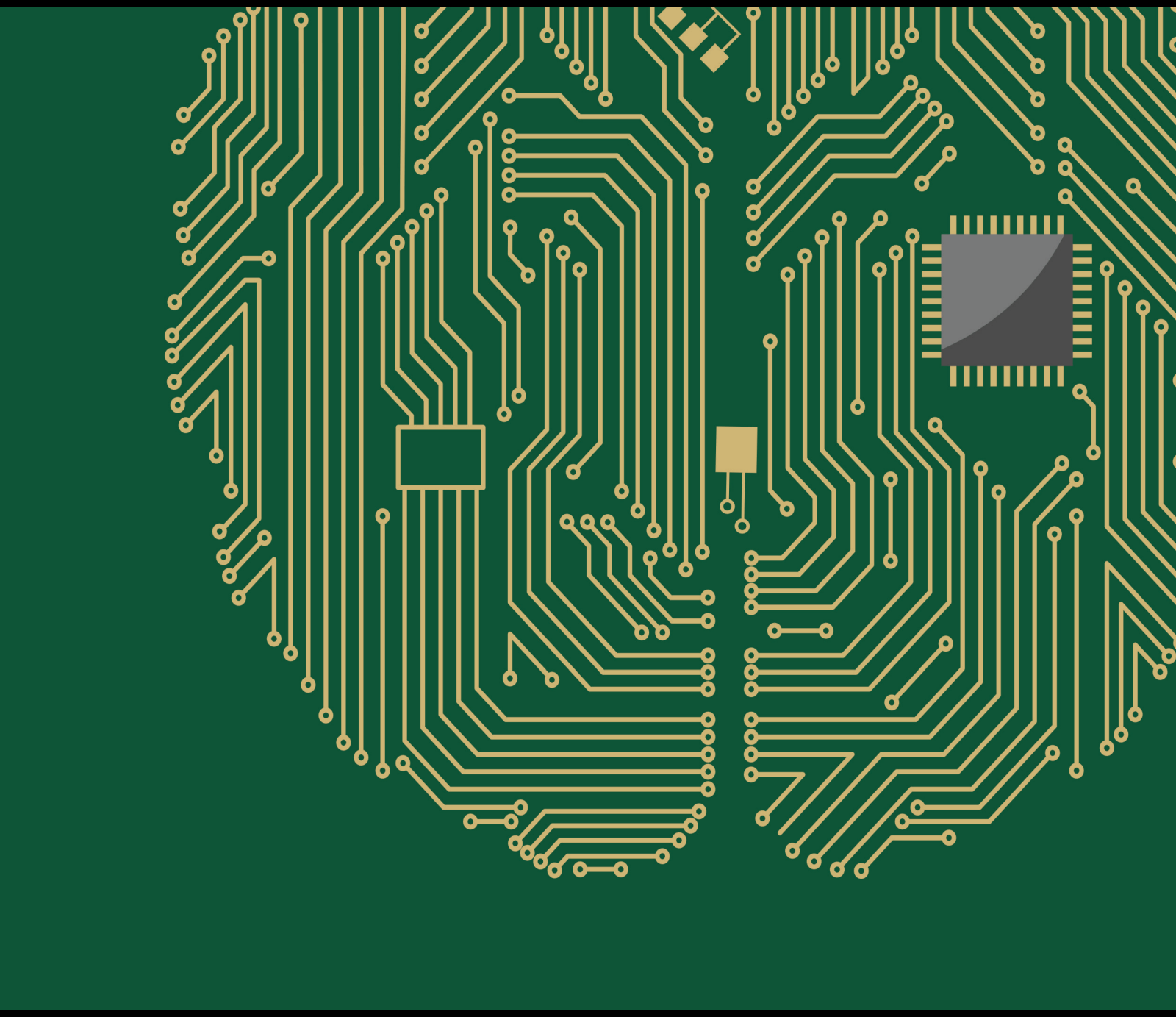
Computational Intelligence and Neuroscience

Artificial Intelligence and Machine Learning in Cyber Defense

Lead Guest Editor: Konstantinos Demertzis
Guest Editors: Lazaros Iliadis and Panayotis Kikiras



Artificial Intelligence and Machine Learning in Cyber Defense

Computational Intelligence and Neuroscience

Artificial Intelligence and Machine Learning in Cyber Defense

Lead Guest Editor: Konstantinos Demertzis
Guest Editors: Lazaros Iliadis and Panayotis Kikiras

Contents

Contents

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9840420, 1 page
https://doi.org/10.1155/2023/9840420

Retraction

Retracted: Cyber Risk Recommendation System for Digital Education Management Platforms

Computational Intelligence and Neuroscience

Received 26 September 2023; Accepted 26 September 2023; Published 27 September 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] X. Yin and Y. Chen, "Cyber Risk Recommendation System for Digital Education Management Platforms," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 8548534, 11 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9828602, 1 page
https://doi.org/10.1155/2023/9828602

Retraction

Retracted: Tech Optimization in Cybersecurity Defenses by Advanced ML Methods: The Use Case of Volleyball Industry

Computational Intelligence and Neuroscience

Received 15 August 2023; Accepted 15 August 2023; Published 16 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Xiao, Z. Bi, and Z. Chen, "Tech Optimization in Cybersecurity Defenses by Advanced ML Methods: The Use Case of Volleyball Industry," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 9907427, 7 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9821676, 1 page
https://doi.org/10.1155/2023/9821676

Retraction

Retracted: Tackling Explicit Material from Online Video Conferencing Software for Education Using Deep Attention Neural Architectures

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Yang and S. Xu, "Tackling Explicit Material from Online Video Conferencing Software for Education Using Deep Attention Neural Architectures," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6334802, 11 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9869278, 1 page
https://doi.org/10.1155/2023/9869278

Retraction

Retracted: Online Anomaly Detection of Industrial IoT Based on Hybrid Machine Learning Architecture

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] J. Guo and Y. Shen, "Online Anomaly Detection of Industrial IoT Based on Hybrid Machine Learning Architecture," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 8568917, 10 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9856503, 1 page
https://doi.org/10.1155/2023/9856503

Retraction

Retracted: A Deep Spiking Neural Network Anomaly Detection Method

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] L. Hu, Y. Liu, and W. Qiu, "A Deep Spiking Neural Network Anomaly Detection Method," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6391750, 13 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9854650, 1 page
https://doi.org/10.1155/2023/9854650

Retraction

Retracted: Privacy-Preserving Sports Wearable Data Fusion Framework

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] J. Li and J. Zhang, "Privacy-Preserving Sports Wearable Data Fusion Framework," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6131971, 7 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 1992429, 9 pages
https://doi.org/10.1155/2022/1992429

Research Article

Preschool Cyber Security Management System Based on Intelligent Agents

Jing Song

Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Jing Song; songjing@zzpec.edu.cn

Received 29 August 2022; Revised 21 September 2022; Accepted 23 September 2022; Published 7 October 2022

Academic Editor: Konstantinos Demertzis

As information and communication technologies create an ever-increasing complexity in interconnected systems and devices, cybersecurity and privacy issues are constantly at the fore, highlighting the need to strengthen the protection and resilience of these systems against the ever-evolving threats of modern cyberspace. This particular work, taking into account that preschool children now have significant needs to ensure their digital identity and, in general, their protection from their contacts with the internet, aspires to provide an understandable and practical guide to strengthen the security of information systems and information from both public and private school agencies. Specifically, a preschool cybersecurity management system based on intelligent agents is proposed. Using sophisticated, intelligent techniques, it aims to improve the ability of preschools to resist modern threats adequately, respond to cyber-attack incidents with the least possible impact, and protect their critical systems, services offered, and the personal data they hold and process. The system intends to link and control distributed systems that currently exist, as well as to solve issues that are beyond the knowledge and skills of a single agent. This novel research idea has never been offered in the relevant literature, and we think it has the potential to advance the state of the art in cybersecurity significantly.

1. Introduction

Today's typical structure of the information systems of an educational institution such as preschools has reached an exceptionally high degree of complexity [1]. Their essential characteristics include at least a central building infrastructure with servers that have public IP (web, mail, DNS, etc.) and with various internal networks that host employees' office computers or other infrastructures used in education [2]. Sometimes, employees bring their own portable devices (laptops, tablets, and smartphones) connected to the carrier's network to the workplace and their mobile storage media (USB, external hard drives, etc.) [3].

The remote offices of the same organization in other regions with their own corresponding internal network infrastructure should be added to the infrastructures in question. The operator's applications assist the specific computing systems, usually web, hosted in the data centers of one or more cloud service providers [4]. Recently, due to the pandemic, the institution's employees, as well as the teachers, work from home (teleworking), connect remotely to the institution's internal network and handle critical data using a home network and computers that have not been tested and certified in terms of their safety [5, 6]. Also, third-party providers and suppliers who have undertaken the development of applications and the technical support of the organization's systems connect remotely to its internal network through their infrastructure or have assigned the work to their subcontractor [7].

As is logical, the sensitive data of educational institutions, especially those related to preschool children, should be supported by information systems that are at least aligned with in-depth defense architecture. In this model, security measures and mechanisms are applied in successive layers

across the entire scope of an operator's network and data to protect them from threats [8]. Each layer individually does not deal with all threats, while they deal with a wide variety of offensive techniques. If a threat manages to bypass a layer, it must deal with the defense mechanisms of the next layer.

An effective defense-in-depth strategy includes mechanisms at the purely technical level, as well as organizational or administrative measures, such as policies and procedures (risk analysis, user training, personal data management, etc.), access restrictions (least privilege, need-to-know, etc.), network security (network segmentation, firewalls, intrusion detection systems, VPNs, etc.), device protection (antivirus, application whitelisting, etc.), and application and data protection (patching, data backup, encryption, etc.) [9]. Figure 1 graphically illustrates an example of the sequential layering of defense-in-depth architecture.

We observe a broad and difficult-to-control dispersion in an operator's data processing, storage, and circulation. At the same time, the traditional network perimeter is no longer demarcated. The above is happening in an interconnected world becoming increasingly vulnerable to malicious activity as connectivity, device richness, distributed applications and services, and complexity in cloud and multicloud environments increase [4, 10, 11]. It is clear that such complexity dramatically increases the security requirements to protect the carrier's critical data from leakage, intentional alteration, or even disruption of availability. Various architecture models have been proposed for the effective defense against constantly evolving threats, which are found in the relevant research literature. But this novel research idea has never been offered in the relevant literature, and we think it has the potential to advance the state of the art in cyber security significantly.

The rest of this paper is structured as follows: Section 2 presents the relevant research studies. Section 3 is allocated to the presentation of the proposed system. Section 4 presents the protocol of the agreement, and finally, section 5 concludes the research.

2. Relevant Research Studies

The literature on the utilization of intelligent agents keeps expanding and covers not only the security aspect of research but also the safety or other perspectives.

Kasereka et al. [12] suggested in 2018 a smart agent-based model to model and simulate the removal of individuals from a burning building. Their concept has been founded on four criteria that allow for her realistic evaluation. In a simulated case study conducted in a building with the general layout of a supermarket, it was determined that the presence of multiple individuals to be removed, the consideration of fire propagation speed, and other aspects significantly impacted the model. This concept is sufficiently broad to be applied in multiple kinds of business buildings without significant modification. We seek to integrate these perspectives into the framework by incorporating a fuzzy approach into the system.

Kotenko et al. [13] presented a strategy for implementing intelligent agents for internet and vulnerability risk assessments in cyber-physical systems. The proposed method has a much smaller sliding window size than the sliding window technique with the same accuracy of assessing traffic characteristics, runs in real-time, and uses fuzzy logical inference to regulate parameters. The experimental evaluation supports the method's fast speed and appropriate precision for analyzing network data. Simultaneously with time, it offers evaluation accuracy equivalent to that of established algorithms. Methods of dynamic supervision of intelligent agents in cyber-physical security systems are the subject of more study.

Kushal et al. [14] developed a two-pronged technique to minimize the consequences of an unusual false data injection attempt, in which an attacker uses batteries to actively lower load curtailment to introduce updates to the central control system. An intelligent agent system examines directives from the primary energy administration system. A bilevel technique is constructed to describe the relationship between the cell and the hacked shipboard power system to discover symptoms of fraudulent data. They developed a heuristic defensive parameter to enhance the detection of tainted instructions. A danger assessment model is used to assess the advantages of the proposed strategy. The findings of the case studies demonstrate that a mixture of an autonomous battery and a heuristic strategy helps reduce the impacts of a cyberattack.

Manbachi and Ordonez [15] suggested an advanced agent-based method for the power management of AC-DC microgrids on isolated islands. This strategy supplied islanded ac-dc microgrids with three major operations interacting at each functioning time interval to improve system performance, efficiency, and dependability. Between the agents, bidirectional communication enabled data gathering and command flow control. They employed an innovative multiobjective particle optimization engine to successfully address each agent's issue for the goal of optimization. A microgrid with various ac-dc producing assets and loads was analyzed to evaluate the accuracy and practical efficiency of the proposed solution.

Biregani and Fotohi [16] presented a countermeasure against malicious UAV assaults. In the first stage, numerous criteria and principles were implemented to identify malicious UAVs. In phase two, a mobile agent was employed to destroy malicious UAVs by alerting regular neighbor UAVs not to listen to the data produced by malicious UAVs. The conjunction of these two steps resulted in secure interactions between the UAVs, allowing packets to exchange data safely. The simulation results demonstrated that the suggested strategy is superior to previous approaches. To detect hostile UAVs in future work, they propose integrating two or more innovative and optimum algorithms, such as earthworm efficiency algorithm, moth search method, monarch butterfly optimum, and elephant herding utilization.

Alhayani et al. [17] investigated the efficacy of artificial intelligence solutions against cyber security threats. They mostly used quantitative research methods and collected primary data from IT sector personnel. Using confirmatory pattern evaluation, discriminant validity, fundamental model analysis, and hypothesis testing, they determined that

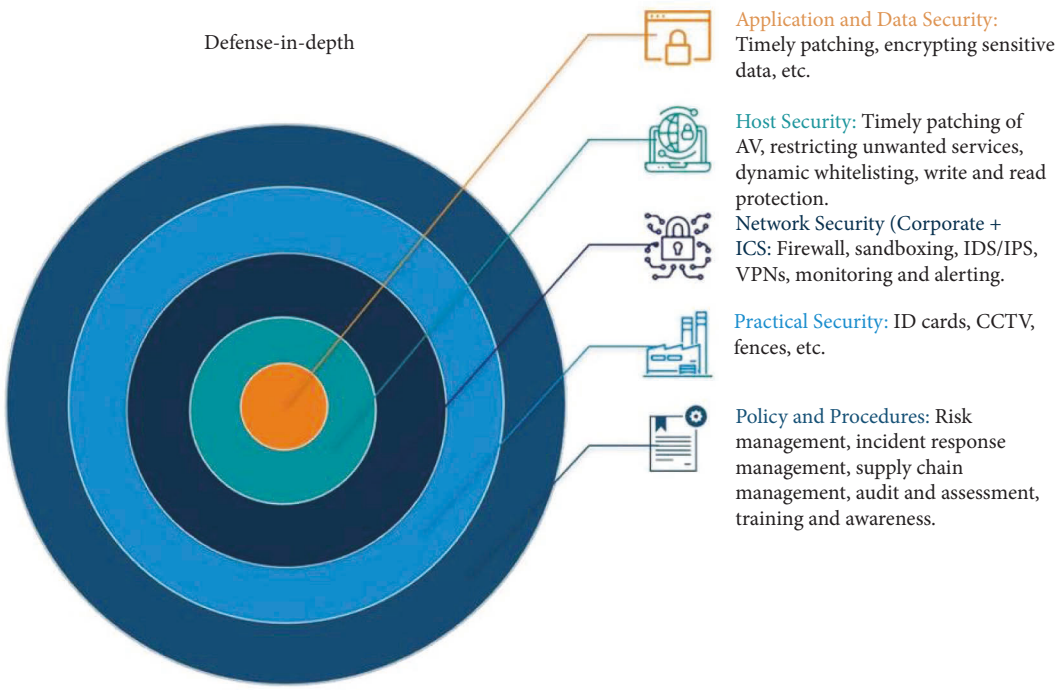


Figure 1: Defense in depth (https://modernciso.com/).

intelligence agents, and artificial neural networks strongly influenced synthetic intelligence approaches. The development of technology has expanded data storage, necessitating better data security.

3. Proposed System

Education agencies increasingly depend on information and communication technologies to carry out their day-to-day operations and mission [1, 2]. These technologies are subject to threats, which exploit known and unknown system vulnerabilities with possible severe effects on business operations, persons, infrastructures, and the safeguarding of sensitive personal data, due to the violation of the confidentiality, integrity, and availability of the information that these systems process, store, or transmit. Threats to IT include cyber-attacks, human errors, and structural failures [8, 18, 19].

For the above reasons, it is imperative for educational institutions, especially preschool education institutions, to realize their responsibility and establish a comprehensive organizational approach to risk management related to the operation and use of information systems.

A vital component of a risk management framework is risk assessment, which consists of the following series of actions [3]:

(1) The sources of threats related to the operator are identified (malicious groups, competitors, natural threats, errors, etc.)

(2) Actions/events (threat events) that could occur from the above sources (cyber-attacks, hardware failures, etc.) are identified

(3) The vulnerabilities of the organization that a source could exploit through specific actions/events are identified

(4) The probability that the identified sources will initiate specific actions and the probability of successful realization of the events are estimated

(5) The adverse effects (on the operations and systems of the entity, on persons, or other organizations) if the actions/events take place are assessed

(6) The risk to the operator's security is determined as a combination of the probability of the events and the adverse effects if the events occur

Based on the calculated risk, the operator should choose the corresponding protection measures to address the risks adequately. Also, the organization should develop a security policy [20, 21], which will define at a high level the security goals and the organization's approach to achieving them while referring to more specific thematic policies and procedures that will specify the implementation and application of the selected protection measures.

Risk management is always the starting point for a practical approach to cyber security. Thus, the agencies must establish an information security management system, which is as follows:

(1) Will be implemented by implementing technical and organizational security measures that will be based on risk management

(2) Will have the full financial and organizational support of the administrative leadership

(3) Will be inspected and renewed at regular intervals and

(4) Will shape a cyber security culture for all involved (from senior management to all involved staff)

An appropriate organizational structure with responsibility for the security of information systems should be created for an information security management system to be implemented effectively [22]. In this structure, it should

(1) Define the appropriate roles and responsibilities
(2) Be adequately staffed with persons possessing technical and legal expertise in cyber security issues and
(3) Allocate the required resources for the implementation of the goals set for cyber security

To automate the above functions and make them independent of human experience and knowledge, this work is proposed to develop a preschool cyber security management system based on intelligent agents [13]. Intelligent agents are modern artificial Intelligence systems that can be used selectively and combined with knowledge representation and problem-solving methods with advanced modern computing technologies [17]. Intelligent agents are computational systems that operate in a complex environment and perceive and act autonomously. In this way, they achieve a set of goals and perform tasks for which they are designed. Intelligent agents continuously perform three functions: they perceive the dynamic conditions of the environment, they act on the background to change it, and they reason to interpret what they perceive, solve problems, and draw conclusions to determine their actions [23].

The proposed system is a multiagent intelligent system that consists of a set of agents that act together to solve the given problem of cyber security management [24, 25]. The system aims to interconnect and operate already existing systems that are distributed, as well as to solve problems that are beyond the capabilities and knowledge of a single agent. Multiagent systems are a vital domain of distributed AI from a loosely considered view of agents, where relevant knowledge is distributed across discrete sources, such as existing experience in individual agent systems.

The proposed multiagent network is a set of agents with dynamic behavior interacting to achieve a common goal. The system in question includes any type of network or system consisting of spatially distributed autonomous devices that collectively record conditions and communicate with each other with wireless or wired devices, exchanging information to achieve an accurate estimate for the desired variable [15]. The system proposes to connect and control existing dispersed systems, as well as to solve problems that are beyond the knowledge and talents of a single person.

An essential element of networked systems, which separates them from the systems that have traditionally been considered in systems theory, is the existence of the network and its effect on the whole system's behavior. The geometry of the network imposes constraints on its behavior, as well as the interactions between agents, described by the graph theory translation of agents as nodes and interactions as branches of a graph representing the network. In such a graph, the existence of a branch indicates that the connected nodes interact with each other.

The agreement is one of the fundamental problems of multiagent coordination, in which a collection of agents must agree on a common state value [24, 26]. In this work, the dynamics of the agreement protocol for undirected static networks are studied to implement the multiagent network of cyber security management.

4. The Protocol of the Agreement

Let there be a multiagent network in which the agents must perform some measurement [27]. Although each measurement made by the individual agents will differ due to its location, it is necessary to reach an agreement on a specific value, which will be achieved by sharing the agents' information. For this purpose, the agents need some communication protocol that will act on the network and allow it to achieve the agreement. The agreement protocol includes n dynamic units, denoted as $1, 2, ..., n$ and are connected by a communication bus between them. Let the state of unit i be $x_i \in R$. Then, the protocol has the form [28]:

$$\dot{x}_i = -\sum_{j \in N(i)} \left(x_i(t) - x_j(t)\right), i = 1, \cdots, n, \tag{1}$$

where $N(i)$ is the set of neighbors of i in the network [29]. The overall network then has momentum

$$\dot{x}(t) = -L(G)x(t), \tag{2}$$

where the positive semidefinite matrix $L(G)$ is the Laplacian of the interaction network of agents G and $x(t) = (x_1(t), \ldots, x_n(t))^T \in \mathbb{R}^n$. The above equation will be referred to as agreement dynamics. With this protocol in action, node potentials are drawn toward the states of neighboring nodes [30]. The value they finally arrive at, i.e., the agreement state, is defined by the agreement set $A \subseteq \mathbb{R}^n$ which is the subset span {1} which is

$$A = \left\{x \in \mathbb{R}^n \,\middle|\, x_i = x_j, \forall i, j\right\}, \tag{3}$$

To clarify the mechanism by which dynamic agreement in an undirected graph drives network nodes to the agreement state, one should consider the eigenvalues of the Laplacian of a connected and undirected graph, which take the form [31]

$$0 = \lambda_0(G) \leq \lambda_1(G) \leq \ldots \leq \lambda_{n-1}(G), \tag{4}$$

where 1, the vector with all elements equal to unity, is the eigenvector corresponding to the zero eigenvalue $\lambda_0(G)$. Recall that the Laplacian is symmetric and $L(G)1 = 0$ for undirected G. Let $U = [u_0 u_1 \ldots u_{n-1}]$ be the matrix consisting of normalized and mutually orthogonal eigenvectors of $L(G)$ which are assigned to the ordered eigenvalues. In addition, let [32, 33]

$$\Lambda(G) = \mathrm{Diag}\left(\left[\lambda_0(G), \ldots, \lambda_{n-1}(G)\right]^T\right). \tag{5}$$

Applying the spectral theorem to the Laplacian, it yields

$$e^{-L(G)t} = e^{-\left(U\Lambda(G)U^T\right)t} = Ue^{-\Lambda(G)t}U^T = e^{-\lambda_0(G)t_0}u_0^T + e^{-\lambda_1(G)t_1}u_1u_1^T + \ldots + e^{-\lambda_{n-1}(G)t}u_{n-1}u_{n-1}^T. \tag{6}$$

Therefore, the solution of $\dot{x}(t) \in F[X](x(t))$, with initial value $x(0) = x_0$ is

$$x(t) = e^{-L(G)t}x_0, \tag{7}$$

which can be decomposed along each eigen-axis as [34]:

$$x(t) = e^{-\lambda_0(G)t}\left(u_0^T x_0\right)u_0 + e^{-\lambda_1(G)t}\left(u_1^T x_0\right)u_1 + \ldots + e^{-\lambda_n(G)t} \cdot \left(u_{n-1}^T x_0\right)u_{n-1}. \tag{8}$$

The problem to be solved is to achieve agreement in a multiagent network with unknown disturbances for a stable network topology. Agents follow simple integrator dynamics. With blocked and continuous second-order derivatives, perturbations are considered blocked. The undirected network description graph is connected. The following solution employs discontinuous systems theory and distributed continuous control [28, 35].

The input of the individual agents is first filtered using the variables T and β as well as the sign of the error ξ. Then, using the filtered information, the final one is generated, which has an additional summation term using ξ. This control is fully distributed, that is, individual agents operate autonomously using information only from their neighbors. It is shown to asymptotically achieve agreement on a stable graph topology [36]. In practice, achieving convergence means that the input learns the perturbation and copes with it.

Specifically, the agent network topology is described using an undirected, connected graph where $G = (V, E)$, where$V = \{u_1, u_2, \ldots, u_n\}$ is the set of its nodes and $E \subseteq V \times V$ all its branches [37, 38]. The set of neighbors of agent i is $N(i) = \left\{u_j \in V \mid u_i u_j \in E\right\}$.

The dynamics of the agents are given by the equation [39]:

$$\dot{x}_i = u_i + d_i, i \in V, \tag{9}$$

where $\dot{x}_i$ are the one-dimensional state variables of the agents, and d_i are unknown blocked perturbations with continuous blocked derivatives up to the second degree and

$$u_i = -K_p\xi_i + u_{fi}, i \in V, \tag{10}$$

u_{fi} is the filtered input

$$T\dot{u}_{fi} + u_{fi} = -\beta \operatorname{sgn}(\xi_i), i \in V, \tag{11}$$

where sgn is the sign function and T, β, Kp are control gains. ξ_i contain the information of agent i about its neighbors and are given by the equation [40]

$$\xi_i = \sum_{j \in N(i)} \alpha_{ij}\left(x_i - x_j\right), i \in V, \tag{12}$$

The problem is to show that agreement is reached, i.e., that $x_i - x_j \longrightarrow 0, \forall i, j \in V$ for fixed graph topology [41].

For the existence of solutions of the system, a Lyapunov function will be used as follows [24, 31, 42, 43]:

$$V\left(x_{ag}, t\right) = \frac{1}{2}\left(\dot{x} + \frac{1}{T}x\right)^T \mathscr{L}\left(\dot{x} + \frac{1}{T}x\right) + \frac{\beta}{T}\sum_{i \in V}|\xi_i| - \sum_{i \in V}\left(\dot{d}_i + \frac{1}{T}d_i\right)\xi_i, \tag{13}$$

where $x_{ag} = (x, \dot{x}, d, \dot{d})$ the augmented state vector with $x, \dot{x}, d, \dot{d} \in \mathbb{R}^n$ and $\mathscr{L}$ the Laplacian of the network graph. All terms of the function are continuously differentiable and therefore normal, except for the term containing the absolute value of ξ_i. By the original hypothesis theorem, this term is also normal, so the Lyapunov function is also normal [44–46].

Using the eigenvalue decomposition of the Laplacian and since the Laplacian has a unique zero eigenvalue, we get [27, 45]

$$\mathscr{L} = \begin{bmatrix} U & \frac{1_n}{\sqrt{n}} \end{bmatrix} \begin{bmatrix} \sum & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U^T \\ \frac{1_n^T}{\sqrt{n}} \end{bmatrix}, \tag{14}$$

where $U \in \mathbb{R}^{n \times (n-1)}$ with property $U^T U = \mathbb{I}_{(n-1)\times(n-1)}$ and $\Sigma \in \mathbb{R}^{(n-1)\times(n-1)}$ has in positions $(i, i), i = 1, \ldots, n-1$ the values $\sigma_i(L) = \sqrt{\lambda_i(L^T L)}$.

In the form presented, the matrix Σ has all nonzero eigenvalues of the Laplacian on its diagonal since the unique zero eigenvalue of the Laplacian has been removed. So, the Laplacian is written in the form [47, 48]

$$\mathscr{L} = U\Sigma U^T = U\Sigma\Sigma^{-1}\Sigma U^T = U\Sigma U^T U\Sigma^{-1}U^T U\Sigma U^T = \mathscr{L}U\Sigma^{-1}U^T\mathscr{L}. \tag{15}$$

Since the Laplacian has an eigenvector of 1, it also holds that

$$\mathscr{L} = \mathscr{L}\left(U\Sigma^{-1}U^T + r\frac{1^T}{N}\right)\mathscr{L}, \tag{16}$$

where r is a positive constant, which is chosen so that $\lambda_1(\mathscr{L}) < r < \lambda_{\max}(\mathscr{L})$. If set

$$\mathscr{L}_0 = \begin{bmatrix} U & \frac{1_n}{\sqrt{n}} \end{bmatrix} \begin{bmatrix} \sum^{-1} & 0 \\ 0 & r \end{bmatrix} \begin{bmatrix} U^T \\ \frac{1_n^T}{\sqrt{n}} \end{bmatrix} = U\sum^{-1}U^T + r\frac{11^T}{N}. \tag{17}$$

The request is satisfied.

Therefore, from the property $\xi = \mathscr{L}x$, the function V_0 can be written in the form [49, 50]

$$V_0 = \frac{1}{2}\left(\dot{\xi} + \frac{1}{T}\xi\right)^T \mathscr{L}_0\left(\dot{\xi} + \frac{1}{T}\xi\right). \tag{18}$$

So the function V_0 is [49, 51] continuous on its set of values as a quadratic function and so holds

$$\frac{d}{dt}\left(\dot{x}_i + \frac{1}{T}x_i\right) = \dot{d}_i + \frac{1}{T}d_i - K_p\left(\dot{\xi}_i + \frac{1}{T}\xi_i\right) + \frac{1}{T}\beta \operatorname{sgn}\ (\xi_i), \forall i \in V, \tag{19}$$

and so

$$\frac{dV_0}{dt} = -K_p \sum_{i\in V}\left(\dot{\xi}_i + \frac{1}{T}\xi_i\right)^2 + \sum_{i\in V}\left(\dot{d}_i + \frac{1}{T}d_i\right)\left(\dot{\xi}_i + \frac{1}{T}\xi_i\right) - \sum_{i\in V}\frac{\beta}{T}\operatorname{sgn}(\xi_i)\left(\dot{\xi}_i + \frac{1}{T}\xi_i\right). \tag{20}$$

Therefore V_0 is blocked and so

$$V_0 = \frac{1}{2}\left(\dot{x} + \frac{1}{T}x\right)^T \mathscr{L}\left(\dot{x} + \frac{1}{T}x\right) \geq \frac{\lambda_{\min}(\mathscr{L}_0)}{2}\left\|\dot{\xi} + \frac{1}{T}\xi\right\|^2. \tag{21}$$

So since V_0 is closed and locally Lipschitz, it is Lipschitz continuous on the set of values of x_{ag}.

So the Lyapunov function is [52, 53]

$$V(x_{ag}, t) = \frac{1}{2}\left(\dot{x} + \frac{1}{T}x\right)^T \mathscr{L}\left(\dot{x} + \frac{1}{T}x\right) + \frac{\beta}{T}\sum_{i\in V}|\xi_i| - \sum_{i\in V}\left(\dot{d}_i + 1/Td_i\right)\xi_i. \tag{22}$$

and is absolutely continuous, over the set of values of x_{ag}. Therefore, it is derivable almost everywhere, and its derivative is

$$\dot{V}(x_{ag}, t) = -K_p \sum_{i\in V}\left(\dot{\xi}_i + 1/T\xi_i\right)^2 - \beta/T^2 \cdot \sum_{i\in V}\operatorname{sgn}(\xi_i)\xi_i + 1/T^2 \sum_{i\in V}\xi_i d_i - \sum_{i\in V}\xi_i \ddot{d}_i. \tag{23}$$

So, the system with a fixed network topology for connected and undirected graphs ensures that $\lim_{t\longrightarrow\infty}\xi(t) = 0$ using distributed continuous control for appropriate control gain β. Thus, it is true that [54]

$$W_1(\xi_{ag}) \leq V(x_{ag}, t) \leq W_2(\xi_{ag}), \tag{24}$$

where

$$W_1(\xi_{ag}) = \frac{1}{2}\lambda_{\min}(\mathscr{L}_0)\left\|\dot{\xi} + \frac{1}{T}\xi\right\|^2 + \sum_{i\in V}|\xi_i|\left(\frac{\beta}{T} + \max_{i\in V}\left|\dot{d}_i + \frac{1}{T}d_i\right|\right), \tag{25}$$

and

$$W_2(\xi_{ag}) = \frac{1}{2}\lambda_{\max}(\mathscr{L}_0)\left\|\dot{\xi} + \frac{1}{T}\xi\right\|^2 + \sum_{i\in V}|\xi_i|\left(\frac{\beta}{T} + \max_{i\in V}\left|\dot{d}_i + \frac{1}{T}d_i\right|\right), \tag{26}$$

W_1 and W_2 are positive definite and continuous, for $\xi_{ag} = (\xi, \dot{\xi})$ for suitable β which exceeds the term $\max|T\dot{d}_i + d_i|$. Moreover, it is true that [38, 43, 49]

$$\dot{V}(x_{ag}, t) \leq^{a.e.} -W(\xi_{ag}(t)). \tag{27}$$

where

$$W(\xi_{ag}) = K_p \sum_{i\in V}\left(\dot{\xi}_i + 1/T\xi_i\right)^2 + \sum_{i\in V}|\xi_i|\left(\beta/T^2 - \max_{i\in V}\left|d_i/T^2 - \ddot{d}_i\right|\right). \tag{28}$$

Eigenvalue decomposition of the Laplacian $\mathscr{L}$ of the network graph yields

$$\mathscr{L} = U\Sigma U^T, \tag{29}$$

and using the equation $\xi = \mathscr{L}x$ holds

$$U\Sigma^{-1}U^T\xi = UU^Tx = \left(\mathbb{I}_n - \frac{1_n 1_n^T}{n}\right)x = x - \frac{\sum_{i=1}^n x_i}{n}1_n, \tag{30}$$

and consequently

$$x = U\Sigma^{-1}U^T\xi + \frac{\sum_{i=1}^n x_i}{n}1_n. \tag{31}$$

Multiplying the above equality by $(e_i - e_j)^T$ results

$$x_i - x_j = \left(e_i - e_j\right)^T U\Sigma^{-1}U^T\xi, \tag{32}$$

and taking the limit at $t \longrightarrow \infty$ implies that $\lim_{t\longrightarrow\infty}(x_i(t) - x_j(t)) = 0$ for every $i, j \in V$.

So, the collective price of the agents converges to the average price of their situations (average consensus).

Unfortunately, no other comparable model exists to serve as a benchmark. As a result, to prevent prejudice or false perceptions, we report the performance of the suggested model without comparing it to any other potential models.

5. Conclusion

In this paper, we suggested a preschool cyber security management system based on intelligent agents. We used cutting edge, intelligent techniques, and it aims to improve the ability of preschools to resist modern threats adequately, respond to cyber-attack incidents with the least possible impact, and protect their critical systems, services offered, and the personal data they hold and process.

The suggested system is a multiagent intelligent system comprising a group of agents that collaborate to solve the cyber security management challenge. The system intends to link and control distributed systems that currently exist, as well as to solve issues that are beyond the knowledge and skills of a single agent. Multiagent systems are a crucial area of distributed AI, where information is dispersed among distinct sources, such as previous experience in individual agent systems. The suggested multiagent network consists of a collection of agents with dynamic behavior collaborating to accomplish a common objective. The system in issue encompasses any network or system composed of geographically dispersed autonomous devices that collectively record circumstances and interact with each other through wireless or wired devices, exchanging data to provide an accurate estimate of the desired variable.

One of the core issues of multiagent coordination is the agreement, in which a group of agents must agree on a shared state value. The dynamics of the agreement protocol for undirected static networks are investigated in this work to implement the multiagent network of cyber security management.

Intelligent applications, such as the ones presented here, offer network defenders command over their environment, enabling them to turn the tables on even the most sophisticated attackers and stop them before they damage them. The need for interdisciplinary knowledge and deep experience in foundational cyber science skills such as understanding crypto analysis methods, building out

security pipelines, and statistics, as well as computer science fundamentals and software engineering skills such as understanding computer architecture, proficiency with programming languages, and the ability to program software solutions is a significant disadvantage of multiagent applications.

Further research will be related to mechanisms of distributed control of intelligent agents in secure communications for other sector specific implementations. Also, I will be studying the multiagent reinforcement learning method that focuses on studying the behaviour of multiple learning agents that coexist in a shared environment.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This study was supported by the Key Project of Comprehensive Education Reform in Henan Province and reform of quality evaluation of kindergarten care and education in Henan province (No. 2022CG0265).

References

[1] L. Tetzlaff, F. Schmiedek, and G. Brod, "Developing personalized education: a dynamic framework," *Educational Psychology Review*, vol. 33, no. 3, pp. 863–882, 2020.

[2] A. Klašnja-Milićević and M. Ivanović, "E-Learning personalization systems and sustainable education," *Sustainability*, vol. 13, no. 12, p. 6713, Jan. 2021.

[3] Z. Amin, "A practical road map for assessing cyber risk," *Journal of Risk Research*, vol. 22, no. 1, pp. 32–43, Jan. 2019.

[4] A. S. Al-Ahmad and H. Kahtan, "Cloud computing review: features and issues," in *Proceedings of the 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, pp. 1–5, Shah Alam, Malaysia, July 2018.

[5] Z. Lassoued, M. Alhendawi, and R. Bashitialshaaer, "An exploratory study of the obstacles for achieving quality in distance learning during the COVID-19 pandemic," *Education Sciences*, vol. 10, no. 9, p. 232, Sep. 2020.

[6] D. Turnbull, R. Chugh, and J. Luck, "Transitioning to E-learning during the COVID-19 pandemic: how have higher education institutions responded to the challenge?" *Education and Information Technologies*, vol. 26, no. 5, pp. 6401–6419, 2021.

[7] H. Alamleh and A. A. S. AlQahtani, "Analysis of the design requirements for remote internet-based E-voting systems," in *Proceedings of the 2021 IEEE World AI IoT Congress (AIIoT)*, pp. 0386–0390, Seattle, WA, USA, May 2021.

[8] P. Ganguly, M. Nasipuri, and S. Dutta, "Challenges of the existing security measures deployed in the smart grid framework," in *Proceedings of the 2019 IEEE 7th International Conference on Smart Energy Grid Engineering (SEGE)*, pp. 1–5, Oshawa, ON, Canada, August 2019.

[9] M. N. Dazahra, F. Elmariami, A. Belfqih, and J. Boukherouaa, "A defense-in-depth cybersecurity for smart substations," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 6, p. 4423, 2018.

[10] M. T. Amron, R. Ibrahim, and S. Chuprat, "A review on cloud computing acceptance factors," *Procedia Computer Science*, vol. 124, pp. 639–646, Jan. 2017.

[11] S. Srivastava, A. Bisht, and N. Narayan, "Safety and security in smart cities using artificial intelligence — a review," in *Proceedings of the 2017 7th International Conference on Cloud Computing, Data Science Engineering - Confluence*, pp. 130–133, Noida, India, January 2017.

[12] S. Kasereka, N. Kasoro, K. Kyamakya, E.-F. Doungmo Goufo, A. P. Chokki, and M. V. Yengo, "Agent-Based Modelling and Simulation for evacuation of people from a building in case of fire," *Procedia Computer Science*, vol. 130, pp. 10–17, 2018.

[13] I. Kotenko, S. Ageev, and I. Saenko, "Implementation of intelligent agents for network traffic and security risk analysis in cyber-physical systems," in *Proceedings of the 11th International Conference on Security of Information and Networks*, pp. 1–4, Cardiff, UK, September 2018.

[14] T. R. B. Kushal, K. Lai, and M. S. Illindala, "Risk-based mitigation of load curtailment cyber attack using intelligent agents in a shipboard power system," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 4741–4750, 2019.

[15] M. Manbachi and M. Ordonez, "Intelligent agent-based energy management system for islanded AC–DC microgrids," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4603–4614, 2020.

[16] M. F. Biregani and R. Fotohi, "Secure communication between UAVs using a method based on smart agents in unmanned aerial vehicles," *The Journal of Supercomputing*, vol. 77, no. 5, pp. 5076–5103, May 2021.

[17] B. Alhayani, H. Jasim Mohammed, I. Zeghaiton Chaloob, and J. Saleh Ahmed, "Effectiveness of artificial intelligence techniques against cyber security risks apply of IT industry," *Materials Today Proceedings*, Article ID S2214785321016722, 2021.

[18] S. Alromaihi, W. Elmedany, and C. Balakrishna, "Cyber security challenges of deploying IoT in smart cities for healthcare applications," in *Proceedings of the 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, pp. 140–145, Barcelona, Spain, December 2018.

[19] B. Hamid, N. Jhanjhi, M. Humayun, A. Khan, and A. Alsayat, "Cyber Security Issues and Challenges for Smart Cities: A Survey," in *Proceedings of the 2019 13th International Conference On Mathematics, Actuarial Science, Computer Science And Statistics (MACS)*, pp. 1–7, Karachi, Pakistan, December 2019.

[20] L. Schaefer and D. Millner, "Flight delay propagation analysis with the detailed policy assessment tool," in *Proceedings of the 2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236)*, vol. 2, pp. 1299–1303, Tucson, AZ, USA, Jul. 2001.

[21] R. Telang, "Policy framework for data breaches," *IEEE Security & Privacy*, vol. 13, no. 1, pp. 77–79, 2015.

[22] Z. Tang, "Analysis of information security problems and countermeasures in big data management of colleges and universities under smart campus environment," in *Proceedings of the 2021 2nd International Conference on Information Science and Education (ICISE-IE)*, pp. 912–915, Chongqing, China, November 2021.

[23] M. Dhingra, M. Jain, and R. S. Jadon, "Role of artificial intelligence in enterprise information security: a review," in *Proceedings of the 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pp. 188–191, Waknaghat, India, December 2016.

[24] D. Zhang and X. Wang, "Opinion dynamics based on game theory in multi-agent network," in *Proceedings of the 2021 3rd International Symposium on Robotics Intelligent Manufacturing Technology (ISRIMT)*, pp. 282–286, Chongqing, China, September 2021.

[25] J. C. Bansal, H. Sharma, K. Deep, K. N. Das, and A. Nagar, "Special issue on swarm intelligence and its applications to engineering," *Int. J. Syst. Assur. Eng. Manag.*, vol. 9, no. 4, pp. 739-740, 2018.

[26] Y.-L. Wang, Z.-Z. Li, and H.-P. Zhu, "Mobile-agent-based distributed and incremental techniques for association rules," in *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693)*, vol. 1, pp. 266–271, Xi'an, China, November 2003.

[27] Z. Cheng, T. Wang, and Y. Xin, "High-order distributed consensus in multi-agent networks," in *Proceedings of the 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS)*, pp. 965–969, Enshi, China, May 2018.

[28] S. T. Arzo, R. Bassoli, F. Granelli, and F. H. P. Fitzek, "Multi-agent based autonomic network management architecture," *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 3595–3618, 2021.

[29] H. Ye, H. Lv, and Q. Sun, "An improved clustering algorithm based on density and shared nearest neighbor," in *Proceedings of the 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, pp. 37–40, Chongqing, China, May 2016.

[30] X. Wang, M. Dai, Y. Chen, and Y. Zong, "Laplacian spectra for a family of treelike networks," in *Proceedings of the IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, pp. 5855–5858, Beijing, China, July 2017.

[31] S. Feng, L. Wang, S. Sun, and C. Xia, "Effect of network structure to the convergence rate of agents in multi-agent systems," in *Proceedings of the 2017 36th Chinese Control Conference (CCC)*, pp. 1408–1412, Dalian, China, July 2017.

[32] L. Zhiwei, Z. Chonghu, S. Jie, L. Juan, and Z. Songhao, "A multi-agent task allocation strategy based on artificial immune system," in *Proceedings of the 2013 25th Chinese Control and Decision Conference (CCDC)*, pp. 3486–3491, Guiyang, China, Feb 2013.

[33] S. Ouiazzane, M. Addou, and F. Barramou, "A multi-agent model for network intrusion detection," in *Proceedings of the 2019 1st International Conference on Smart Systems and Data Science (ICSSD)*, pp. 1–5, Rabat, Morocco, October 2019.

[34] S. M. Amrr and M. Nabi, "Attitude regulation of spacecraft using large angle eigen-axis rotations," in *Proceedings of the 2020 International Conference on Decision Aid Sciences and Application (DASA)*, pp. 783–787, Bahrain, November 2020.

[35] C. Ma, W. Wu, H. Fu, and C. Wang, "Distributed leader-follower consensus of multi-agent systems with unreliable networks," in *Proceedings of the 2020 7th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS)*, pp. 581–584, Guangzhou, China, November 2020.

[36] C. Ma and Y. Gao, "Study on mesh segmentation of topology optimization results using Reeb graph," in *Proceedings of the 2021 International Conference on Artificial Intelligence and Electromechanical Automation (AIEA)*, pp. 277–280, Guangzhou, China, Feb 2021.

[37] M. Mohammadian, "Network security risk assessment using intelligent agents," in *Proceedings of the 2018 International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)*, pp. 1–6, Putrajaya, Malaysia, December 2018.

[38] H. Liu, R. Gu, Z. Li, and Y. Ji, "Multi-agent federated reinforcement learning for privacy-enhanced service provision in multi-domain optical network," in *Proceedings of the 2021 Asia Communications and Photonics Conference (ACP)*, pp. 1–3, Shanghai, China, July 2021.

[39] N. A. Musa, M. Z. M. Yusoff, R. Ismail, and Y. Yusoff, "Issues and challenges of forensics analysis of agents' behavior in multi-agent systems: a critical review," in *Proceedings of the 2015 International Symposium on Agents, Multi-Agent Systems and Robotics (ISAMSR)*, pp. 122–125, Putrajaya, Malaysia, December 2015.

[40] C. C. Aggarwal, "Neighborhood-based collaborative filtering," in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed., Springer International Publishing, New York, NY, USA, 2016.

[41] V. Alieksieiev and B. Andrii, "Information analysis and knowledge gain within graph data model," in *Proceedings of the 2019 IEEE 14th International Conference on Computer Sciences and Information Technologies (CSIT)*, vol. 3, pp. 268–271, Lviv, Ukraine, September 2019.

[42] H. K. Dambanemuya and E. -Á. Horvát, "Network-aware multi-agent simulations of herder-farmer conflicts," in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 721-722, Vancouver, BC, Canada, December 2019.

[43] S. E. Benton, E. Rogers, and D. H. Owens, "Lyapunov stability theory for linear repetitive processes — the 2D equation approach," in *Proceedings of the 1999 European Control Conference (ECC)*, pp. 4768–4773, Karlsruhe, Germany, December 1999.

[44] S. Chen, M. Fazlyab, M. Morari, G. J. Pappas, and V. M. Preciado, "Learning Lyapunov Functions for Hybrid Systems," in *Proceedings of the 2021 55th Annual Conference On Information Sciences And Systems (CISS)*, p. 1, Nashville, Tennessee, March 2021.

[45] Y. Ji, J. Huang, and J. Hu, "Optimal Lyapunov-based states transfer for superconducting qubits," in *Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA)*, pp. 849–852, Ningbo, China, December 2016.

[46] N. Wang, H. Liu, and W. Chen, "Lyapunov-based excitation control for the synchronous generator unit," in *Proceedings of the 32nd Chinese Control Conference*, pp. 899–903, Xi'an, China, July 2013.

[47] S. Gao, I. W.-H. Tsang, and L.-T. Chia, "Laplacian sparse coding, hypergraph laplacian sparse coding, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 92–104, 2013.

[48] Y. Liu and S. Liao, "An error bound for eigenvalues of graph laplacian with bounded kernel function," in *Proceedings of the 2011 Seventh International Conference on Computational Intelligence and Security*, pp. 436–440, Sanya, China, September 2011.

[49] G. Alessandrini, M. V. de Hoop, R. Gaburro, and E. Sincich, "Lipschitz stability for a piecewise linear Schrödinger potential from local Cauchy data," *Asymptotic Analysis*, vol. 108, no. 3, pp. 115–149, 2018.

[50] V. Sokolov, "Adaptive suboptimal tracking under bounded lipshitz uncertainty and disturbance in discrete-time minimum-phase plant," in *Proceedings of the 2016 International Conference Stability and Oscillations of Nonlinear Control*

Systems (Pyatnitskiy's Conference), pp. 1–4, Moscow, Russia, June 2016.

[51] C. Freer, B. r. Kjos-Hanssen, A. Nies, and F. Stephan, "Algorithmic aspects of Lipschitz functions," *Computability*, vol. 3, no. 1, pp. 45–61, Jan. 2014.

[52] L. Hu and L. Wang, "H Fuzzy filtering design via membership function dependent Lyapunov function," in *Proceedings of the 2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS)*, pp. 348–353, Jinzhou, China, December 2016.

[53] W. Junqiang and X. Meiqing, "Stochastic stability analysis of the power systems based on Lyapunov function," in *Proceedings of the 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, pp. 1–9, Beijing, China, July 2018.

[54] Y. Gao and L. Jia, "Stability in measure for uncertain delay differential equations based on new Lipschitz conditions," *Journal of Intelligent and Fuzzy Systems*, vol. 41, no. 2, p. 2997, 2021.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4043309, 11 pages
https://doi.org/10.1155/2022/4043309

Research Article

A Semi-Self-Supervised Intrusion Detection System for Multilevel Industrial Cyber Protection

Fuchuan Ye[1] **and Weiqiong Zhao**[2]

[1]*Information and Educational Technology Center, Southwest Minzu University, Chengdu 610041, China*
[2]*School of Intelligent Technology, Geely University of China, Chengdu 641423, China*

Correspondence should be addressed to Fuchuan Ye; fuchuan_ye@163.com

Received 30 June 2022; Revised 12 August 2022; Accepted 12 August 2022; Published 21 September 2022

Academic Editor: Konstantinos Demertzis

Industry 4.0 affects all components of the modern industry value chain. The accelerating use of the Internet and the convergence of industrial and operational networks constantly increase the need for secure industrial communication solutions. Therefore, "multilevel industrial cyber protection" is critical to Industry 4.0. In general, industrial protection refers to safeguarding information and data and the intellectual property rights of production processes related to the overall industry environment. The availability, integrity, and confidentiality of systems must be maintained. The goal challenge is the best possible protection from attacks and threats which create immediate financial damage and other risks in the industry (reputation, etc.). Based on the Defense-in-Depth strategy, a holistic, multilayered, and in-depth protection of industrial systems is developed in this paper. Specifically, a Semi-Self-Supervised Intrusion Detection System (S3IDS) is proposed, which combines advanced machine learning techniques for industrial data noise reduction to automate the discovery and separation of classes, which are essentially equivalent to cyber-related anomalies. As demonstrated by a mathematical simulation based on computational number theory and specifically on the concept of the single object, the proposed S3IDS learns to accurately reconstruct samples to predict the nature of an anomaly created directly by the industrial ecosystem.

1. Introduction

Historically, industrial companies worldwide have approached cybersecurity in their Information Technology (IT) and Operational Technology (OT) networks very differently [1]. Most companies have already implemented technological infrastructures for detecting and dealing with network threats, but, for their industrial (OT) systems, coping with cyber threats is usually limited to isolating the relevant procedures from the rest of the network. Industries are constantly being "digitized" by investing more and more in intelligent technologies, new automation systems, and other applications that promote productivity growth or improve many other indicators of interest to the organization [2]. This rapidly equates IT systems with OT systems, making the latter more vulnerable to attacks that formerly solely affected the former [3].

Cyberattacks on industrial organizations are considered a perilous threat [4], as they have the potential to cause significant material losses and lead to disruption of the production cycle of the entire system [5]. They target, among others, industrial control and data collection systems (ICSs, SCADA) [6, 7]. In addition, due to the sensitive information available to industrial organizations, they are usually an attractive target for attackers [3].

The situation to date focuses on the human aspect, experience, and expert opinion, using assistive technology to analyze and reduce risks and dangers to industrial infrastructure [8, 9]. For optimal results with this methodology, there should be up-to-date threat intelligence, incident reports, and vulnerability warnings, which will feed indefinitely the power grid monitoring tools and in-depth human oversight and intervention from cybersecurity staff [10].

The above passive function, in combination with the new class of requirements in cybersecurity, leads to the logic of adopting solutions that include fully automated security methods based on advanced techniques of artificial intelligence [11], with the parallel minimization of human intervention [12]. The idea of getting rid of the constant surveillance and direct presence of people is related to advanced attacks like Stuxnet and BlackEnergy, where it turned out that it just needed an infected USB stick or open a phishing e-mail to allow the attacker to access an isolated industrial network [5]. In addition, throughout the last several months, we have witnessed, in many cases, highly specialized attacks on systems and infrastructures that use industrial protocols [4, 13].

A great example is the largest colonial gas pipeline in the USA, which was shut down for several days after a malicious cyberattack and attributed to the shadow criminal group DarkSide [14–16]. Also, in the first quarter of 2021, in the city of Oldsmar in Florida, there was an attack on the government infrastructure responsible for the city's water supply. In essence, a remote attempt was made to change the mix of related chemicals with water disinfection, resulting in the mass poisoning of consumers [7, 15, 17, 18].

Just months ago, another cyberattack occurred in a public hospital in Israel. The specific attack created significant problems in the smooth provision of services of the organization, while it required the payment of a certain amount of money (ransomware). Hospital services reacted by using alternate resources and support systems. Fortunately, there was no loss of life in this case, as unfortunately happened at a similar point in a hospital in Germany a few months ago [4, 6, 15].

To deal with these offensive techniques, the research community has proposed various solutions in which machine learning systems operate with self-adaptation procedures and rearrange their mode of operation, depending on the algorithms' hyperparameters that most often specify their mode of operation.

2. Literature Review

Several researches have presented adaptive cyberattack detection algorithms to fulfill the requirement for continuing learning paradigm changes [19–21]. Still, they have failed to establish a more comprehensive system of knowledge for detection performance and their evaluation practice [22–24].

In their review of earlier work for threat detection techniques in industrial control systems, Kaouk et al. [3] underlined the difficulties and advantages of putting such solutions into practice. Such information, in our opinion, will be helpful for future studies in manufacturing security. ICS intrusion detection technology is evolving swiftly, but there is still room for improvement. The integration of IDS with ICS will face a variety of risks. Most methods used in the literature are anomaly based, meaning that they look for any notable departure from the norm. To enhance how IDS can react to alarms, techniques that can tell the difference between a flaw and a threat are desperately needed. Another difficulty is that the vast majority of existing IDS are network based and cannot access encrypted data because of this. For instance, encryption use is hampered by hardware limitations. However, new parts of the ICS have begun to offer encryption because of advancements in hardware computation capabilities. IDS must therefore rely on data sources other than Internet activity. The operation of IDS should also be taken into account as ICS grow in size and complexity and comprise geographically dispersed systems. Alternative technologies decentralized and collaborative IDS must therefore be created. Such information, in our opinion, will help advance future studies on the integrity of ICS.

Hu et al. [25] went into more detail on ICS's attributes and security needs in 2018. They proposed a taxonomy of IDS for industrial control systems based on three techniques: protocol analysis, traffic mining, and control process analysis. They also examined the benefits and drawbacks of various IDS categories. They concluded that, despite the rapid advancement of ICS technology, there is still much opportunity for ICS IDS development. It was crucial to construct dispersed and collaborative IDS due to the scattered structure of ICS subsystems. Evaluating associations between distributed IDS, fusing a group of dispersed and potentially contradictory detection findings, and obtaining accurate and real-time complete detection results are a novel and intriguing subject. How to react to warnings is a major problem for ICS IDS. In specific control systems, simply notifying administrators of the alarm may be considered sufficient; nevertheless, automatic reaction mechanisms must be taken into account to ensure the protection and reliability of ICS. How to automatically improve intrusion detection algorithms while they are being used is a crucial topic. To maintain a satisfactory detection accuracy, intrusion detection algorithms must automatically optimize their judgments of changing contexts. ICSs typically need to operate continuously, and the system parameters (such as durable components, access controls, and system constraints) of an objective ICS may change over time. These days, ICSs are internet-accessible, and ICS security concerns are increasingly becoming more critical. Traditional IDS created for IT platforms cannot function well on ICS because of its uniqueness. It can assist ICS in identifying various intrusions and lowering the frequency of industrial mishaps caused by malicious attacks.

Adversarial Machine Learning, often known as cyberattacks over neural network models on Engineering IDS, was examined by Anthi et al. [4]. By constructing adversarial samples and evaluating classification patterns, they studied how adversarial learning may be used to target supervised models. As adversaries could be able to get beyond the defenses, such attacks could have dire effects on ICS systems. This can result in delayed assault detection, which might harm the infrastructure, cause financial loss, or even result in fatalities. An actual electric grid data set was utilized for training and evaluating commonly used unsupervised feature learning classifiers in support of the studies described here. The investigation also studies how adversarial training on such sets can enhance the resilience of supervised models. Using the testing data, adversarial samples were created with

various combinations that changed the model's interference and complexity.

According to Ayodeji et al. [4], in 2020, the failure to recognize and distinguish between the intrinsically identical signatures that define normal transients typical of complex systems contributes significantly to false alarms in ICS systems. The majority of machine learning-based detection techniques created for Scada Systems (ICSs) are taught on network packet logs and solely rely on network layer traffic monitoring to identify intrusions. They looked at the most current developments in malware detection algorithms, their shortcomings, difficulties, and the state of their use in crucial infrastructures. Additionally, they started a conversation about the parallels and differences between the growth of computational skills and equipment for classification and hacking in defense of complex systems and the requirement to distinguish between them clearly. They used nuclear energy controllers as a case study to demonstrate the challenges to a smooth changeover of security algorithms. To significantly reduce the number of annoyance warnings generated, they suggested a method that considers the subtleties in the data utilized in creating machine learning algorithms. The current findings and recommended course of action lay the groundwork for creating robust intrusion detection systems that significantly reduce the problem of false alarms that plague existing intrusion detection systems.

The transition of the ICS from isolated systems to virtualized platforms was closely examined by Bhamare et al. [1], who also noted the considerable efforts made by both business and technology to construct secure ICSs and the relevance of machine learning approaches for ICS cybersecurity. ICS security remains a concern despite the recent popularity of big data insights and cloud computing. Cloud platforms will eventually help ICSs and industries. Still, inadequate security in cutting-edge multicloud platforms could result in expensive security breaches in real-time industry platforms. It is incredibly challenging to prevent and identify assaults at the ICS component level due to the sophistication of emerging viruses attacking control systems, including rootkits and zero-day attacks. New intrusion detection strategies for ICS devices at the production control level are thus required. Additionally, they said that a testbed might help with the difficulties of safeguarding an industrial process by offering more information about how the method is managed with the aid of sensors and control laws and comprehension of the security needs, mainly to handle control using cloud-based services.

An examination of the development and usefulness of security mechanisms that have been put out in both industry and academia was presented by Rubio et al. in [2]. In the past several years, there has been a tremendous advancement in the design of security methods for industrial environments [1]. Advanced solutions like honeypot systems and data correlation systems are integrated into commercially accessible products, but innovative detection techniques and architectures are also created in academia [19]. Research is still needed in several areas, including the viability and incorporation of proactive defenses, the deployment of defensive mechanisms in the IIoT and cloud computing, and the emergence of Industry 4.0[26]. Furthermore, to validate defense mechanisms against Advanced Persistent Threats (APTs) and make them more integrable and usable so they can be readily integrated into more crucial infrastructures, it is vital to take into account existing APTs and APT phases [15, 16, 21].

In this spirit, an approach is needed that with minimal configuration and the necessary training samples each time will be able to create a generalized framework for detecting known and unknown attacks on a network. Based on the above challenge and the Defense-in-Depth strategy, in general, S3IDS is proposed that should be applied in the industry. Using advanced machine learning methods automates recognizing anomalies related to cyberattacks [27]. To prove the applicability, we used mathematical simulation based on computational number theory. Mathematical simulation is a process to identify and predict the behavior, performance, and optimization of some physical or abstract systems corresponding to various scientific and engineering applications.

3. Proposed S3IDS

Given the general issues of machine learning systems to deal against serious cyberattacks effectively and with minimal human intervention, this work proposes the creation of an innovative computer intelligence system [28, 29], with minimal human intervention [30–32], significantly strengthening the security mechanisms of network infrastructure [12, 23, 33]. In particular, S3IDS is proposed, an advanced cyber threat detection system, which is a highly innovative tool for operational security. Specifically, we implement a semi-self-adapted machine learning methodology [9–11] based on Semi-Self-Supervised Learning, which may determine the sort of attack based on generic reshaping characteristics generated directly from the unknown online environment and web data [22, 34, 35].

The proposed system's major innovation is based on computational number theory, notably the idea of a monoid object in a category. Monoids are semigroups that have an identity. A monoid is a set containing an associative binary operation and an identity member in abstract algebra, a field of mathematics. For example, nonnegative integers with addition form a monoid, with 0 as the identity member. Such algebraic structures may be found in many disciplines of mathematics. In terms of function composition, the functions from a set create a monoid. In general, the morphisms of an item form a monoid in category theory; conversely, a monoid may be considered a category containing a single entity.

Many abstract data types in computer science may have a monoid structure. A succession of monoid components is "folded" or "stacked" to generate a final value in a recognizable pattern. Many iterative algorithms, for example, must update some "current set" at each iteration. A monoid function may be used to represent this pattern cleanly. In particular, the proposed methodology ensures that the correlation of monoid operations can be predicted using a correlation algorithm, effectively using multiple cores [36, 37].

In particular, if A is a nonempty set, the operation on A for any representation of the form $f: A \times A - \longrightarrow A$; e.g., addition and multiplication are operations on Z. The value of f in the pair (a, b) will be denoted by afb. A pair $(G, *)$, where G is a set and $*$ is one operation on G, is called a monoid if the following properties are valid [36, 38, 39]:

$$x * (y * z) = (x * y) * z, \tag{1}$$

such as

$$x * e = x = e * x. \tag{2}$$

If there is another element $k \in G$with the above property, then for every $x \in G$ we have

$$k * x = x = x * k. \tag{3}$$

Thus, we get $k = e * k$ and $e * k = e$, from where $e = k$. Therefore, the element e is unique and is called the neutral element of G. If $x * y = y * x$ is also valid for every $x, y \in G$, then the monoid $(G, *)$is permutable. So, the pairs $(N, +)$, $(Z, +)$, $(Q, +)$ are substitutively monosyllabic with neutral element 0 and the pairs $(N, \cdot)$, $(Z, \cdot)$, $(Q, \cdot)$ are substitutive monoid.

Respectively, if $(G_i, *_i)$ is a monoid with neutral element $e_i\,(i = 1, \ldots, k)$, the set $G_1 \times \cdots \times G_k$ is a monoid with the operation [40, 41]:

$$(x_1, \ldots, x_k) * (y_1, \ldots, y_k) = (x_1 *_1 y_1, \ldots, x_k *_k y_k). \tag{4}$$

Its neutral component is

$$(e_1, \ldots, e_k). \tag{5}$$

If we have a function with the field of definition, the set of positive integers, and a field of values, the set of complex numbers (numerical function), then we denote by A the set of numerical functions, while the numerical function calculates the exponential product of f and g [36, 37, 39]:

$$f * g\colon N \setminus \{0\} \longrightarrow C, n \longmapsto (f * g)(n) = \sum_{ab=n} f(a) g(b), \tag{6}$$

where the pairs (a, b) run through all the natural whose product is equal to n. The correspondence $(f, g) - \longrightarrow f * g$ defines an operation on A, which is called associative multiplication since the pair $(A, *)$ is a permutable monoid. If $g, h \in A$, then for every natural $n > 0$ we have [36, 37, 39]

$$\begin{aligned} [f * (g * h)](n) &= \sum_{ab=n} f(a)(g * h)(b) \\ &= \sum_{ab=n} f(a) \sum_{c\,d=b} g(c) h(d) \\ &= \sum_{ac\,d=n} f(a) g(c) h(d). \end{aligned} \tag{7}$$

Similarly, we get

$$[(f * g) * h](n) = \sum_{ac\,d=n} f(a) g(c) h(d). \tag{8}$$

Therefore, for every natural $n > 0$ it holds

$$[f * (g * h)](n) = [(f * g) * h](n). \tag{9}$$

And so

$$f * g = g * f. \tag{10}$$

Next, consider the numerical function ϵ defined by the relations:

$$\begin{aligned} \epsilon(1) &= 1, \\ \epsilon(n) &= 0. \end{aligned} \tag{11}$$

For every $f \in A$ and natural $n > 1$, we have

$$(f * \epsilon)(n) = \sum_{ab=n} f(a) \epsilon(b) = f(n), \tag{12}$$

where $f * \epsilon = f$. As the operation $*$ is transitive, the relation $\epsilon * f = f$ is also valid. So, the function ϵ is the neutral element for associative multiplication. Therefore, the pair $(A, *)$ is a permutable monoid. $(G, *)$ is a monoid and ϵ its neutral element. A subset H of G is called a submonoid of G if $\epsilon \in H$ and for every $x, y \in H$ it holds $x * y \in H$; that is, the pair $(H, *)$ is also a monoid with a neutral element ϵ.

Based on the above view, $(A, *)$, $(B, \diamond)$, an d $(C, \triangleright)$ are monoids with neutral elements eA, eB, and eC, respectively, and $f: A \longrightarrow B$, $f: B \longrightarrow C$ are monoid morphisms. We will show that the expression $g \circ f$ is a monoid morphism since, for every $x, y \in A$, the composition of two morphisms of monoids is a monoid morphism, which is proved by the following relation [36, 38, 41]:

$$\begin{aligned} (g \circ f)(x * y) &= g(f(x * y)) \\ &= g(f(x) \diamond f(y)) \\ &= g(f(x)) \triangleright g(f(y)) \\ &= (g \circ f)(x) \triangleright (g \circ f)(y). \end{aligned} \tag{13}$$

Also, it holds

$$\begin{aligned} (g \circ f)(e_A) &= g(f(e_A)) \\ &= g(e_B) \\ &= e_C. \end{aligned} \tag{14}$$

This hypothesis creates a process where the data in a machine learning system is predicted with high accuracy (any anomalies are recognized) even when they come slightly modified [22, 33]. The output of the intelligent mechanism can now be considered as a recognition of the input data's shifted prediction, based on the isomorphism of monoids that may appear in the unknown data set (assuming a uniform distribution which, although unknown, includes properties of monoid theory). That is, the output of the intelligent mechanism approaches the displaced version of the input as the intelligent system is trained. The machine learning system learns to distinguish displaced samples using this approach, resulting in highly generalized algorithmic frameworks for detecting abnormalities [19, 20].

Given that the synthesis of two monoid morphisms is a monoid morphism, proving that the inverse representation of a monoid isomorphism is likewise a monoid isomorphism suffices for implementing this mechanism [36, 37, 39].

So, considering $(M, *), (N, \diamond)$ monoids and $f : M \longrightarrow N$ isomorphism of monoids, if $y_1, y_2 \in N$, then there exist $x_1, x_2 \in M$ with $y_1 = f(x_1)$ and $y_2 = f(x_2)$. The above formulation is related to the hypothesis of a supervised learning problem, where a set of training with N samples, $\{X,Y\} = \{x_i, y_i\}_{i=1}^{N}$, where xi $\in R^{n_i}$, y_i is a no-dimensional binary vector with only one input (corresponds to the class x_i) equal to a multidimensional categorization process, where ni and no are the input and output dimensions, respectively. Unlabeled data helps study the data structure of the accessible data set, but classified data aids in learning. With this in mind, we have [36, 41]

$$\begin{aligned} f^{-1}(y_1 \diamond y_2) &= f^{-1}(f(x_1) \diamond f(x_2)) \\ &= f^{-1}(f(x_1 * x_2)) \\ &= \left(f^{-1} \circ f\right)(x_1 * x_2) \\ &= I_G(x_1 * x_2) \\ &= x_1 * x_2 \\ &= f^{-1}(y_1) * f^{-1}(y_2). \end{aligned} \tag{15}$$

If eM and eN are the neutral elements of M and N, respectively, then $f(e_M) = e_N$ and therefore $f^{-1}(e_N) = e_M$ and f^{-1} is a monoid morphism.

$(G, *)$ is a monoid with neutral elements e and $x \in G$. Assume that $y \in G$ exists such that

$$x * y = e = y * x. \tag{16}$$

In this case, the element y is unique because if y' is another element with this property, then

$$\begin{aligned} y &= y * e \\ &= y * (x * y') \\ &= (y * x) * y' \\ &= e * y' \\ &= y'. \end{aligned} \tag{17}$$

So, the element y is symmetric to x. Also, the symmetric of y is x.

But since in a monoid each element does not always have a symmetric, then f must be calculated which has a symmetric element g (associative inverse of f). If and only if $g * f = e$, which is equivalent to $(1) f(1) = 1$, then [38, 40]

$$\sum_{ab=n} g(a) f(b) = 0, \tag{18}$$

for every natural $n > 1$. In general, for every natural $n > 1$, it applies

$$f^*(n) = -\frac{1}{f(1)} \sum_{st=n, t<n} f(s) f^*(t). \tag{19}$$

Therefore, f has an associative inverse if and only if $f(1) \neq 0$. The derivative of the function is

$$f_\Delta(t) = \frac{\mathrm{d}u_\Delta(t)}{\mathrm{d}t} = \{0, ttn < q0h_{1x}/\Delta C, ; 0 \le t \le \Delta\, 0, t \ge \Delta. \tag{20}$$

So, the data set is obtained as a subscale of the signal processing process for analyzing and manipulating the physical quantities that define the given problem of information systems security [42]. Thus, when $\Delta \longrightarrow 0$, the duration of the pulse decreases and its height increases, but the area remains constant and equal to the unit. So, we study the function $f(t)$ as an operator that acts on other functions that are smooth at points 0. Thus, we can express the function $f(t)$ as [43–45]

$$\int_{-\infty}^{+\infty} f(t)\varphi(t)\mathrm{d}t = \varphi(0), \tag{21}$$

where $\varphi(t)$ is a test function, for $f(t) = 0$ and $t \neq 0$. So, the above process can be generalized to describe the time-shifted data expressed by the functionf (t-t_0) ADDIN ZOTERO_ITEM CSL_CITATION {"citationID":"l8CicIUS","properties":{"formattedCitation":" [46]\\uc0\\u8211{┤} [47]","plainCitation":" [46–48]","noteIndex":0},"citationItems":[{"id":47,"uris":["http://zotero.org/users/local/knpFELzr/items/RISE.2017.8378144","event":"2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)", "page":"153–156","source":"IEEE Xplore","title": "Moving object detection using self adaptive Gaussian Mixture Model for real-time-applications","author":[{"family":"Ali","given": "Syed Tariq"},{"family":"Goyal","given": "Kalpana"},{"family":"Singhai","given":"Jyoti"}],"issued": {"date-parts":[["2017",7]]}}},{"id":277,"uris":["http://zotero.org/users/local/knpFELzr/items/ICOMSSC45026.2018.8941982","event": "2018 International Computers, Signals and Systems Conference (ICOMSSC)","page":"378–381","source": "IEEE Xplore","title": "A Method of Fast Extract Signal Subspace Based on the Householder Transformation","author": [{"family":"Chang","given":"Yu"},{"family": "Wan","given":"Qun"},{"family":"Xia","given": "Changxiong"},{"family":"Wan","given": "Yihe"}],"issued":{"date-parts":[["2018",9]]}}},{"id": 279,"uris":["http://zotero.org/users/local/knpFELzr/items × 32K complex points) image, achieving real-time-performance.","container-title": "2016 IEEE 13th International Conference on Signal Processing (ICSP)","DOI":"10.1109/ICSP.2016.7877887","event":"2016 IEEE 13th International Conference on Signal Processing (ICSP)","note":"ISSN: 2164–5221", "page":"513–517","source":"IEEE Xplore","title": "Design of a flexible high-performance real-time SAR signal processing system","author":[{"family":

"Jin","given":"Ting"},{"family":"Wang","given":"Hongxian"},{"family":"Liu","given":"Hongwei"}],"issued":{"date-parts":[["2016",8]]}}}],"schema":"https://github.com/citation-style-language/schema/raw/master/csl-citation.json"}
[46–48]:

$$\begin{aligned}\int_{-\infty}^{+\infty} f(t-t_0)\varphi(t)\mathrm{d}t &= \int_{-\infty}^{+\infty} f(t-t_0)\varphi(t_0)\mathrm{d}t \\ &= \varphi(t_0)\int_{-\infty}^{+\infty} f(t-t_0)\mathrm{d}t \\ &= \varphi(t_0).\end{aligned} \tag{22}$$

The above relation describes the mathematical model of the sampling process applied during the application of the semisupervised learning technique of the proposed machine learning model [49–51].

For $\varphi(t) = 1$, we have [52–55]

$$\int_{-\infty}^{+\infty} f(t)\mathrm{d}t = \int_{0^-}^{0^+} f(t)\mathrm{d}t = 1. \tag{23}$$

And so

$$\int_{t_1}^{t_2} f(t-t_0)\varphi(t)\mathrm{d}t = \{\varphi(t_0), t_1 < t_0 < t_2\, 0, t_0 < t_1, t_0 > t_2. \tag{24}$$

However,

$$\int_{t_1}^{t_2} f(\tau-t)f(t-t_0)\mathrm{d}t = f(t-t_0), t_1 < t_0 < t_2. \tag{25}$$

According to the logic presented by the system under consideration, the error function is defined as the integral [56]:

$$erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\mathrm{d}t, x > 0. \tag{26}$$

Also, the complementary error function is defined as the integral [57]:

$$erfc(x) = \frac{2}{\sqrt{\pi}}\int_x^{\infty} e^{-t^2}\mathrm{d}t, x > 0. \tag{27}$$

So, the error function and the complementary error function satisfy the following equation:

$$er\ f(x) + er\ fc(x) = 1. \tag{28}$$

The above hypotheses are proved based on the observation that

$$erf(x) + erfc(x) = \frac{2}{\sqrt{\pi}}\int_0^{\infty} e^{-t^2}\mathrm{d}t. \tag{29}$$

For the calculation of the integral [58],

$$\int_0^{\infty} e^{-t^2}\mathrm{d}t. \tag{30}$$

We consider

$$t^2 = u \Rightarrow 2t\,dt = du. \tag{31}$$

Since the integration ends are the same, we have

$$\begin{aligned}\int_0^{\infty} e^{-t^2}\mathrm{d}t &= \int_0^{\infty} e^{-u}u^{-1/2}\frac{du}{2} \\ &= \frac{1}{2}\int_0^{\infty} e^{-u}u^{1/2-1}\mathrm{d}u \\ &= \frac{1}{2}\Gamma\left(\frac{1}{2}\right) \\ &=^{(2.9)} \frac{\sqrt{\pi}}{2}.\end{aligned} \tag{32}$$

This fact proves the above hypotheses about the relationships of the error functions.

Finally, a self-supervised learning methodology [17, 59, 60] is an unsupervised learning method where supervised learning work is created from unlabeled input data. Simple supervised learning usually requires a lot of labeled data. Obtaining good quality labeled data is a costly and time-consuming task, especially for a complex task such as detecting anomalies. On the other hand, unlabeled information is readily available in abundance. So, the motivation behind the self-supervised learning methodology is to learn useful representations of industrial data from an unlabeled data pool using the semisupervised process and then refine the few-tagged representations for the supervised work.

The implementation of the self-supervised learning methodology will require the reconstruction loss function, which is responsible for capturing the essential features of the context of the complete categorization process. The loss function used to train an undercomplete autoencoder is called reconstruction loss, as it is a check of how well the image has been reconstructed from the input [54, 61, 62]:

$$L_{rec}(x) = \| \widehat{M} \odot (x - F((1-\widehat{M})\odot x))\|_2^2, \tag{33}$$

and the adversarial loss which models the latent data entry space of the monoid morphisms in which the following is trained:

$$\begin{aligned}L_{adv} = \max_D\ E_{x\in X}[\log\ (D(x)) \\ +\log\ (1 - D(F((1-\widehat{M})\odot x)))].\end{aligned} \tag{34}$$

Joint loss was used to implement the proposed template utilizing the combination of the above functions as follows:

$$L = \lambda_{rec}L_{rec} + \lambda_{adv}L_{adv}. \tag{35}$$

To develop representations encapsulating the underlying standard information across various regions of the data while rejecting low-level information and noise that is a local phenomenon, we use the Contrastive Predictive Coding technique [63–66]:

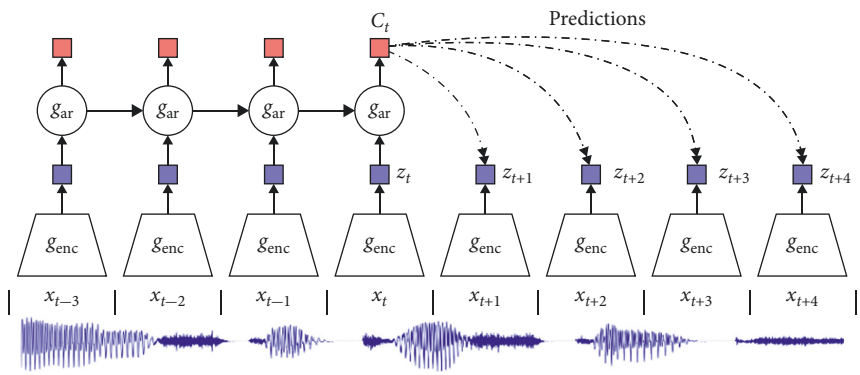


Figure 1: Contrastive Predictive Coding example (https://anilkeshwani.github.io/CPC/).

$$L_{\theta^a,\theta^+,\{\theta^-\}} = -\log \frac{\exp \exp\left(\theta^a \cdot \left(\theta^+/k\right)\right)}{\exp\left(\theta^a \cdot \left(\theta^+/k\right)\right) + \sum_{\theta^-} \exp\left(\theta^a \cdot \left(\theta^+/k\right)\right)}. \tag{36}$$

For example, given a lack of information, Figure 1 depicts the Contrastive Predictive Coding network, where "x" is a time series signal, data for which is available until time "t" and the model must predict the signal by the time "$t+4$." Here, "g_{enc}" is an integration network that extracts "z_t" attributes from the "x_t" signal and "g_{ar}" is a self-regression model that summarizes all the $z \leq t$ in the integration space to produce a latent representation of the environment $c_t = g_{ar}$ $(z \leq t)$ [67]. This composite representation is used to model a density ratio that maintains the mutual information between the predicted signal and the aggregate environment "c_t" [68–71].

Thus, in the proposed system, we combine future observation predictions with a likely loss linked to whether each monoid element is always symmetric [72].

So, for a 2-class (binary) classification problem, we have [69, 71] the following.

Activation function:

$$y = \sigma(a) \equiv \frac{1}{1 + \exp(-a)}. \tag{37}$$

Probability:

$$p(t \mid x, w) = \prod_{n=1}^{N} y(x_n, w)^{t_n} \{1 - y(x_n, w)\}^{1-t_n}. \tag{38}$$

Error function:

$$E(w) = -\sum_{n=1}^{N} \{t_n \ln y_n + (1 - t_n) \ln(1 - y_n)\}. \tag{39}$$

Moreover,

$$\frac{\partial E}{\partial a_k} = y_k - t_k. \tag{40}$$

However, for classification with k-classes (multiclass), we have the following.

Activation function:

$$y_k(x, w) = \frac{\exp \quad \exp\left(a_k(x, w)\right)}{\sum_j \exp\left(a_j(x, w)\right)}. \tag{41}$$

Probability:

$$p(T \mid W) = \prod_{n=1}^{N} \prod_{k=1}^{K} y_{nk}^{t_{nk}}. \tag{42}$$

Error function:

$$E(w) = -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{kn} \ln y_k(x_n, w). \tag{43}$$

Furthermore,

$$yy_k = a_k \Rightarrow \frac{\partial E}{\partial a_k} = y_k - t_k. \tag{44}$$

When an anomaly is run through the model, it will not recreate it since it is taught only to reproduce standard data, resulting in a considerable Mean Absolute Percentage Error (MAPE) [73, 74]:

$$\frac{100}{n} \sum_{i}^{n} \frac{y_i - \widehat{y}_i}{y_i}. \tag{45}$$

The comparison and the final categorization are achieved by defining a threshold value for MAPE, which is not sensitive to extreme values. At the same time, its values are normalized based on the actual observation, so it predicts the sample's class with high precision and recall [75].

This procedure may be repeated multiple times if it makes sense; that is, the reconstruction at each phase is adequate, implying that the new objectives are not too challenging. When a sample goes to a displaced region, it is always conceivable that it may end up in a zone with more opponents than previously. Furthermore, even if his aim puts him in a better position than before, there is a potential that the sample will be rebuilt in a worse situation. In these circumstances, repeating the operation for the troublesome pieces each time seems reasonable. That is, rediscover the problematic samples as they emerge from the categorizer's reconstruction of the details, to use the procedure again to locate the inverse of the function and to begin the process

from where the previous phase of categorizing had ended. So, if the reconstruction is good enough, the procedure can detach the network from local minimums and may be done numerous times. The suggested method's most significant novelty is the simple confirmation of the results of assigning classes to an unknown collection of values using quantifiable criteria [64, 76].

Finally, we have a reduction in data dimension, clear separation of classes, and self-adaptation with this method, as the proposed system learns to reconstruct the wrong samples in the supersphere defined by computational number theory and precisely the concept of the single object to perceive the nature of an unknown state based on generalized reshaped characteristics that come directly from the unknown environment.

4. Conclusions

Attempting to comment on the proposed system, it is a sophisticated practice that solves an essential problem of information systems security with great accuracy and reliability. With the proposed methodology currently presented and simulated by mathematical modeling, the artificial intelligence algorithm leads to a high learning rate, which is determined by how fast the industrial system converges. In general, self-adaptation and self-learning functionality enable identifying and maintaining fundamental characteristics of complex patterns that grow and contribute to the timely and accurate forecast of circumstances completely relevant to the industrial environment.

The proposed technique significantly strengthens the methodology because, in this problem of high complexity under consideration, the results of the prediction eliminate the variability, which is attributed to the sensitivity of industrial data. This complicated connection identifies and captures the minute distinctions that set them out amid the chaotic din. The suggested technique assures that the correlation of monoid operations can be anticipated using an intelligent correlation algorithm, efficiently using multiple learning cores and matching machine learning algorithmic structures with the single-mode process.

Furthermore, an additional benefit derived from the suggested function is that it provides better prediction and a more stable categorization rate since the general behavior of the model minimizes the overall probability of an awful decision that may be associated with occurrences such as this notion. This is because modern industrial data generators generate data in huge quantities and at high speed. The result is an increase in flow data. Extracting useful information from flow data is a challenge because its nature imposes constraints that cannot be satisfied by classical learning algorithms. Stream data is infinitely large, so it is not stored in memory, and each snapshot is usually and only accessible once. So, the snapshots are not available from the beginning as they arrive at a fast pace. Also, every snapshot is processed within a short time, and access to the actual price is limited. Most important, however, is the possibility of a change in the essential data production function, which is predicted with high reliability by the proposed system.

It is critical to stress that the quality of model adaption is interpreted as a percentage of "prediction self-improvement" owing to the higher rate of categorization accuracy fluctuation using this approach. The high percentages of accuracy reached after the general convergence of the reconstructed samples represent the temporal bias induced towards the dynamics of a model at a particular moment.

Another critical interpretation that emerges from the proposed algorithm's methodology is the characteristics of the relatively low rate of "mutation" in the changes that characterize the data shift, which allows the discovery of local extremes that may be included in a learning context, given the exploration of new areas of the multidimensional solution space. On the other hand, if the rate of "mutation" was too great, it might restrict the utilization of regions of high appropriateness in the solution space and imprison the system in nongeneralizable solutions.

The basic model is high speed, owing to the limits on the connections between the hidden and visible units that make it up, as mentioned above. Because of the algorithm hidden layer's function, where the teams of one level rely solely on branches of the other level, it also efficiently and precisely detects high-level correlations in data sets. Another significant feature of the proposed method is its ability of separating and rejecting random noise in the training set. The addition of automation to reclassifying complex data as a future extension of the proposed system is essential. This is the most realistic way of operating and using intelligent systems in the operational security of modern industrial infrastructures and systems.

Suggestions for future development and enhancements to this system should also concentrate on further improving the settings of the heuristic approach of redefining and rearranging the issue samples utilized to obtain an even more efficient, accurate, and quicker classification process. Finally, it is critical to investigate the expansion of this algorithm for the analysis and classification of real-time data presented in streams so that it can completely automate identifying even stealth zero-day attack types.[77]

Data Availability

The data can be obtained from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan, and N. Meskin, "Cybersecurity for industrial control systems: a survey," *Computers & Security*, vol. 89, Article ID 101677, 2020.

[2] J. E. Rubio, C. Alcaraz, R. Roman, and J. Lopez, "Current cyber-defense trends in industrial control systems," *Computers & Security*, vol. 87, Article ID 101561, 2019.

[3] M. Kaouk, J.-M. Flaus, M.-L. Potet, and R. Groz, "A review of intrusion detection systems for industrial control systems," in *Proceedings of the 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 1699–1704, Paris, France, April 2019.

[4] E. Anthi, L. Williams, M. Rhode, P. Burnap, and A. Wedgbury, "Adversarial attacks on machine learning cybersecurity defences in Industrial Control Systems," *Journal of Information Security and Applications*, vol. 58, Article ID 102717, May 2021.

[5] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, "Cyber threats to industrial IoT: a survey on attacks and countermeasures," *IoT*, vol. 2, no. 1, p. 1, 2021.

[6] I. Zerdazi and M. Fezari, "SCADA attack modeling using bond graph," in *Proceedings of the 2019 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*, pp. 1-2, Paris, France, September 2019.

[7] C. Kaura, N. Sindhwani, and A. Chaudhary, "Analysing the impact of cyber-threat to ICS and SCADA systems," in *Proceedings of the 2022 International Mobile and Embedded Technology Conference (MECON)*, pp. 466–470, March 2022.

[8] W. Li, L. Xie, D. Liu, and Z. Wang, "False logic attacks on SCADA control system," in *Proceedings of the 2014 Asia-Pacific Services Computing Conference*, pp. 136–140, Fuzhou, China, September 2014.

[9] A. Babay, T. Tantillo, T. Aron, M. Platania, and Y. Amir, "Network-attack-resilient intrusion-tolerant SCADA for the power grid," in *Proceedings of the 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 255–266, Luxembourg, Luxembourg, June 2018.

[10] R. Atat, L. Liu, H. Chen, J. Wu, H. Li, and Y. Yi, "Enabling cyber-physical communication in 5G cellular networks: challenges, spatial spectrum sensing, and cyber-security," *IET Cyber-Physical Systems: Theory & Applications*, vol. 2, no. 1, pp. 49–54, 2017.

[11] A. M. Kanca and Ş. SAĞIROĞLU, "Sharing cyber threat intelligence and collaboration," in *Proceedings of the 2021 International Conference on Information Security and Cryptology (ISCTURKEY)*, pp. 167–172, Ankara, Turkey, September 2021.

[12] M. A. Umer, K. N. Junejo, M. T. Jilani, and A. P. Mathur, "Machine learning for intrusion detection in industrial control systems: applications, challenges, and recommendations," *International Journal of Critical Infrastructure Protection*, Article ID 100516, 2022.

[13] H. Yang, L. Cheng, and M. C. Chuah, "Deep-learning-based network intrusion detection for SCADA systems," in *Proceedings of the 2019 IEEE Conference on Communications and Network Security (CNS)*, pp. 1–7, Washington, DC, USA, June 2019.

[14] F. A. Alhaidari and E. M. Al-Dahasi, "New approach to determine DDoS attack patterns on SCADA system using machine learning," in *Proceedings of the 2019 International Conference on Computer and Information Sciences (ICCIS)*, pp. 1–6, Sakaka, Saudi Arabia, Apil 2019.

[15] S.-P. Hong, C.-H. Lim, and H. J. Lee, "APT attack response system through AM-HIDS," in *Proceedings of the 2022 24th International Conference on Advanced Communication Technology (ICACT)*, pp. 271–274, PyeongChang Kwangwoon_Do, Republic of Korea, 2022.

[16] P. V. S. Charan, P. Mohan Anand, S. K. Shukla, N. Selvan, and H. Chunduri, "DOTMUG: a threat model for target specific APT attacks–misusing google teachable machine," in *Proceedings of the 2022 10th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1–8, Istanbul, Turkey, June 2022.

[17] Y. Xue, Q. Zhang, and F. Neri, "Self-Self-Adaptive Particle Swarm Optimization-Based Echo State Network for Time Series Prediction.daptive particle swarm optimization-based echo state network for time series prediction," *International Journal of Neural Systems*, vol. 31, no. 12, Article ID 2150057, 2021.

[18] S.-X. Lin, Z.-J. Li, T.-Y. Chen, and D.-J. Wu, "Attack tactic labeling for cyber threat hunting," in *Proceedings of the 2022 24th International Conference on Advanced Communication Technology (ICACT)*, pp. 34–39, PyeongChang Kwangwoon_Do, Republic of Korea, October 2022.

[19] L. Xing, K. Demertzis, and J. Yang, "Identifying data streams anomalies by evolving spiking restricted Boltzmann machines," *Neural Computing and Applications*, vol. 32, no. 11, pp. 6699–6713, 2020.

[20] K. Demertzis, L. Iliadis, E. Pimenidis, and P. Kikiras, "Variational restricted Boltzmann machines to automated anomaly detection," *Neural Computing and Applications*, vol. 34, 2022.

[21] K. Demertzis, D. Taketzis, V. Demertzi, and C. Skianis, "An Ensemble Transfer Learning Spiking Immune System for Adaptive Smart Grid Protectionnsemble transfer learning spiking immune system for adaptive smart grid protection," *Energies*, vol. 15, no. 12, p. 4398, 2022.

[22] J. Guo and Y. Shen, "Online Online Anomaly Detection of Industrial IoT Based on Hybrid Machine Learning Architecturenomaly detection of industrial IoT based on hybrid machine learning architecture," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 8568917, 10 pages, 2022.

[23] X. Zheng and X. Yin, "A A Privacy-Preserved Variational-Autoencoder for DGA Identification in the Education Industry and Distance Learningrivacy-preserved variational-autoencoder for DGA identification in the education industry and distance learning," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 7384803, 8 pages, 2022.

[24] W. Jiang, "Machine Learning Methods to Detect Voltage Glitch Attacks on IoT/IIoT Infrastructuresearning methods to detect voltage glitch attacks on IoT/IIoT infrastructures," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6044071, 7 pages, 2022.

[25] Y. Hu, A. Yang, H. Li, Y. Sun, and L. Sun, "A survey of intrusion detection on industrial control systems," *International Journal of Distributed Sensor Networks*, vol. 14, no. 8, Article ID 155014771879461, 2018.

[26] A. Facchini, "Semilocal categories and modules with semilocal endomorphism rings," 2022, https://link.springer.com/book/10.1007/978-3-030-23284-9.

[27] F. Zhao, H. Zhang, J. Peng, X. Zhuang, and S.-G. Na, "A semi-self-taught network intrusion detection system," *Neural Computing and Applications*, vol. 32, no. 23, pp. 17169–17179, 2020.

[28] S. Algarni, F Eassa, K Almarhabi et al., "Blockchain-Blockchain-Based Secured Access Control in an IoT Systemased secured access control in an IoT system," *ApplApplied Sciences*, vol. 11, no. 4, p. 1772, 2021.

[29] X. Li, "A A Blockchain-Based Verifiable User Data Access Control Policy for Secured Cloud Data Storagelockchain-based verifiable user data access control policy for secured

cloud data storage," *Computational Intelligence and Neuroscience*, vol. 2022, pp. Apr–12, Article ID 2254411, 2022.

[30] M. R. Naeem, R. Amin, S. S. Alshamrani, and A. Alshehri, "Digital Digital Forensics for Malware Classification: An Approach for Binary Code to Pixel Vector Transitionorensics for malware classification: an approach for binary code to pixel vector transition," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6294058, 12 pages, 2022.

[31] Z. Ma, J. Li, Y. Song, X. Wu, and C. Chen, "Network Network Intrusion Detection Method Based on FCWGAN and BiLSTMntrusion detection method based on FCWGAN and BiLSTM," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6591140, 17 pages, 2022.

[32] W. Zhang, Y Zhang, C Guo et al., "Certificateless Certificateless Hybrid Signcryption by a Novel Protocol Applied to Internet of Thingsybrid signcryption by a novel protocol applied to internet of things," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 3687332, 7 pages, 2022.

[33] W. Jiang, "A A Machine Vision Anomaly Detection System to Industry 4.0 Based on Variational Fuzzy Autoencoderachine vision anomaly detection system to industry 4.0 based on variational fuzzy autoencoder," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 1945507, 10 pages, 2022.

[34] D. Li, J. Wang, Z. Tan, X. Li, and Y. Hu, "Differential privacy preservation in interpretable feedforward-designed convolutional neural networks," in *Proceedings of the 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 631–638, Guangzhou, China, September 2020.

[35] M. A. Albahar, M. S. ElSayed, and A. Jurcut, "A A Modified ResNeXt for Android Malware Identification and Classificationodified ResNeXt for android malware identification and classification," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 8634784, 20 pages, May 2022.

[36] Y. Dandan, V. Gould, M. Hartmann, N. Ruškuc, and R.-E. Zenab, "Coherency and Coherency and Constructions for Monoidsonstructions for monoids," *The Quarterly Journal of Mathematics*, vol. 71, no. 4, pp. 1461–1488, 2020.

[37] H. Machida, "Centralizing monoids, majority operations and the słupecki clone," in *Proceedings of the 2022 IEEE 52nd International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 62–67, Dallas, TX, USA, February 2022.

[38] M. Behrisch, "Centralising monoids with conservative majority operations as witnesses," in *Proceedings of the 2021 IEEE 51st International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 56–61, Nur-sultan, Kazakhstan, February 2021.

[39] T. Ishida and S. Inokuchi, "Commutativity of composition of some elementary cellular automata on monoids," in *Proceedings of the 2020 Eighth International Symposium on Computing and Networking (CANDAR)*, pp. 128–133, Naha, Japan, August 2020.

[40] H. Machida and I. G. Rosenberg, "Centralizing monoids on a three-element set related to binary idempotent functions," in *Proceedings of the 2016 IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 84–89, Sapporo, Japan, February 2016.

[41] H. Machida and I. G. Rosenberg, "Centralizing monoids and the arity of witnesses," in *Proceedings of the 2017 IEEE 47th International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 236–241, Novi Sad, Serbia, February 2017.

[42] M. Behrisch and V. García, "Centralising Monoids with Low-Arity Witnesses on a Four-Element Set," vol. 13, no. 8, p. 1471, 2022, https://www.mdpi.com/2073-8994/13/8/1471.

[43] A. Hassanpour, M. Moradikia, H. Adeli, S. R. Khayami, and P. Shamsinejadbabaki, "A novel end-to-end deep learning scheme for classifying multi-class motor imagery electroencephalography signals," *Expert Systems*, vol. 36, no. 6, Article ID e12494, 2019.

[44] W. Ruan, J. Zhao, and X. Bai, "Block backtracking-based matching pursuit for arbitrary block sparse signal recovery," in *Proceedings of the 2018 International Conference on Signals and Systems (ICSigSys)*, pp. 209–212, Bali, Indonesia, February 2018.

[45] K. Yan, H.-C. Wu, H. Xiao, and X. Zhang, "Novel robust band-limited signal detection approach using graphs," *IEEE Communications Letters*, vol. 21, no. 1, pp. 20–23, 2017.

[46] S. T. Ali, K. Goyal, and J. Singhai, "Moving object detection using self adaptive Gaussian Mixture Model for real time applications," in *Proceedings of the 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, pp. 153–156, Bhopal, India, July 2017.

[47] T. Jin, H. Wang, and H. Liu, "Design of a flexible high-performance real-time SAR signal processing system," in *Proceedings of the 2016 IEEE 13th International Conference on Signal Processing (ICSP)*, pp. 513–517, Chengdu, China, August 2016.

[48] Y. Chang, Q. Wan, C. Xia, and Y. Wan, "A method of fast extract signal subspace based on the householder transformation," in *Proceedings of the 2018 International Computers, Signals and Systems Conference (ICOMSSC)*, pp. 378–381, Dalian, China, September 2018.

[49] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.

[50] Y. Ouali, C. Hudelot, and M. Tami, *An Overview of Deep Semi-Supervised Learning*, 2020, https://arxiv.org/abs/2103.00550.

[51] Y.-F. Li and D.-M. Liang, "Safe semi-supervised learning: a brief introduction," *Frontiers of Computer Science*, vol. 13, no. 4, pp. 669–676, 2019.

[52] Q. Liu, X. Liao, and L. Carin, "Semi-supervised life-long learning with application to sensing," in *Proceedings of the 2007 2nd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pp. 1–4, St. Thomas, VI, USA, September 2007.

[53] P. K. Mallapragada, R. Jin, A. K. Jain, and Y. Liu, "SemiBoost: SemiBoost: Boosting for Semi-Supervised Learningoosting for semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, 2009.

[54] M. Soliman, C. Lehman, and G. AlRegib, "S6: semi-supervised self-supervised semantic segmentation," in *Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP)*, pp. 1861–1865, Abu Dhabi, United Arab Emirates, July 2020.

[55] C. Wei, C. Guo, and W. Yan, "Forest fire risk forecast method with pseudo label based on semi-supervised learning," in *Proceedings of the 2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, pp. 36–39, Taiyuan, China, September 2021.

[56] J. G. Altonji and R. L. Matzkin, "Cross section and panel data estimators for nonseparable models with endogenous regressors," *Econometrica*, vol. 73, no. 4, pp. 1053–1102, 2005.

[57] "Co-simulation study on vibration control of multistage gear transmission system based on multiple control algorithms | IEEE Conference Publication | IEEE Xplore," https://ieeexplore.ieee.org/abstract/document/8316474.

[58] K. Demertzis, L. S. Iliadis, and V.-D. Anezakis, "Extreme deep learning in biosecurity: the case of machine hearing for

marine species identification," *J. Inf. Telecommun.*vol. 2, no. 4, pp. 492–510, 2018.

[59] A. J. Kulkarni, I. P. Durugkar, and M. Kumar, "Cohort Cohort Intelligence: A Self Supervised Learning Behaviorntelligence: a self supervised learning behavior," in *Proceedings of the 2013 2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1396–1400, Manchester, UK, July 2013.

[60] K. M. O. Vale, A. C. Gorgônio, F. D. L. E. Gorgônio, and A. M. D. P. Canuto, "An An Efficient Approach to Select Instances in Self-Training and Co-Training Semi-Supervised Methodsfficient approach to select instances in self-training and Co-training semi-supervised methods," *IEEE Access*, vol. 10, pp. 7254–7276, 2022.

[61] L. Song and W. Luo, "Self-supervised learning of visual odometry," in *Proceedings of the 2020 International Conference on Information Science, Parallel and Distributed Systems (ISPDS)*, pp. 5–9, Xi'an, China, December 2020.

[62] L. Zhou, X. Ling, S. Zhu, Z. Sun, and J. Yang, "An self-supervised learning & self-attention based method for defects classification on PCB surface images," in *Proceedings of the 2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT)*, pp. 229–234, Sanya, China, September 2021.

[63] L. Chen, X. Liang, Y. Feng, L. Zhang, J. Yang, and Z. Liu, "Online Online Intention Recognition With Incomplete Information Based on a Weighted Contrastive Predictive Coding Model in Wargame.ntention recognition with incomplete information based on a weighted contrastive predictive coding model in wargame," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2022.

[64] J. Ebbers, M. Kuhlmann, T. Cord-Landwehr, and R. Haeb-Umbach, "Contrastive predictive coding supported factorized variational autoencoder for unsupervised learning of disentangled speech representations," in *Proceedings of the ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3860–3864, Toronto, Canada, June 2021.

[65] X. Zhu, H. Dong, P. S. Rossi, and M. Landrø, "Time-Time-Frequency Fused Underwater Acoustic Source Localization Based on Contrastive Predictive Codingrequency fused underwater acoustic source localization based on contrastive predictive coding," *IEEE Sensors Journal*, vol. 22, no. 13, pp. 13299–13308, 2022.

[66] R. Qiu, Z. Huang, and H. Yin, "Memory Memory Augmented Multi-Instance Contrastive Predictive Coding for Sequential Recommendationugmented multi-instance contrastive predictive coding for sequential recommendation," in *Proceedings of the 2021 2021 IEEE International Conference on Data Mining (ICDM)*, pp. 519–528, Auckland, New Zealand, September 2021.

[67] X. Zhu, H. Dong, P. S. Rossi, and M. Landrø, "Self-supervised Self-supervised Underwater Source Localization based on Contrastive Predictive Codingnderwater source localization based on contrastive predictive coding," in *Proceedings of the 2021 IEEE Sensors*, pp. 1–4, Sydney, Australia, July 2021.

[68] Y. Chen, J. Zhao, W. Wang et al., "SEQ-CPC: sequential contrastive predictive coding for automatic speech recognition," in *Proceedings of the ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3880–3884, Toronto, Canada, June 2021.

[69] C. Wan, T. Zhang, Z. Xiong, and H. Ye, "Representation learning for fault diagnosis with contrastive predictive coding," in *Proceedings of the 2021 CAA Symposium on Fault Detection, Supervision, and Safety for Technical Processes (SAFEPROCESS)*, pp. 1–5, Chengdu, China, September 2021.

[70] J. Nistal, C. Aouameur, S. Lattner, and G. Richard, "VQCPC-GAN: variable-length Adversarial audio synthesis using vector-quantized contrastive predictive coding," in *Proceedings of the 2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 116–120, Paltz, NY, USA, July 2021.

[71] A. van den Oord, Y. Li, and O. Vinyals, *Representation Learning with Contrastive Predictive Coding*, 2019, https://arxiv.org/abs/1807.03748.

[72] E. Anthi, L. Williams, A. Javed, and P. Burnap, "Hardening Machine Learning Denial of Service (DoS) Defences against Adversarial Attacks in IoT Smart home Networks - ScienceDirect,"vol. 108 https://www.sciencedirect.com/science/article/pii/S0167404821001760002520.

[73] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A review on generative adversarial networks: algorithms, theory, and applications," 2020, http://arxiv.org/abs/2001.06937.

[74] Handbook of Statistics, *Bayesian Thinking, Modeling and Computation - PDF Free Download*, vol. 25 https://epdf.tips/handbook-of-statistics-volume-25-bayesian-thinking-modeling-and-computation.html.

[75] K. Demertzis, L. S. Iliadis, and V. Anezakis, *A Dynamic Ensemble Learning Framework for Data Stream Analysis and Real-Time Threat Detection*, https://www.springerprofessional.de/en/a-dynamic-ensemble-learning-framework-for-data-stream-analysis-a/16154694.

[76] S. V. Boštjančič Rakas and M. D. Stojanović, "A Review of Research Work on Network-Based SCADA Intrusion Detection Systems," *IEEE Journals & Magazine | IEEE Xplore*, vol. 8 https://ieeexplore.ieee.org/author/37086663933https://doi.org/10.1109/ACCESS.2020.2994961https://ieeexplore.ieee.org/document/9094250.

[77] A. Ayodeji, Y. k. Liu, N. Chao, and L. q. Yang, "A new perspective towards the development of robust data-driven intrusion detection for industrial control systems," *Nuclear Engineering and Technology*, vol. 52, no. 12, pp. 2687–2698, 2020.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9856503, 1 page
https://doi.org/10.1155/2023/9856503

Retraction

Retracted: A Deep Spiking Neural Network Anomaly Detection Method

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] L. Hu, Y. Liu, and W. Qiu, "A Deep Spiking Neural Network Anomaly Detection Method," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6391750, 13 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6391750, 13 pages
https://doi.org/10.1155/2022/6391750

Research Article

A Deep Spiking Neural Network Anomaly Detection Method

Lixia Hu,[1] **Ya Liu**,[2] **and Wei Qiu**[1]

[1]*Department of Computer Science and Engineering, Langfang Polytechnic Institute, Langfang 065000, China*
[2]*Department of Electrical Automation, Hebei University of Water Resources and Electric Engineering, Cangzhou 061001, China*

Correspondence should be addressed to Ya Liu; liu_yaxcd@yeah.net

Received 26 July 2022; Revised 11 August 2022; Accepted 26 August 2022; Published 21 September 2022

Academic Editor: Konstantinos Demertzis

Cyber-attacks on specialized industrial control systems are increasing in frequency and sophistication, which means stronger countermeasures need to be implemented, requiring the designers of the equipment in question to re-evaluate and redefine their methods for actively protecting against advanced mass cyber-attacks. The attacks in question have huge motivations, ranging from corporate espionage to political targets, but in any case, they have a substantial financial impact and severe real-world implications. It should also be said that it is challenging to defend against cyber threats because a single point of entry can be enough to destroy an entire organization or put it out of business. This paper examines threats to the digital security of vibration monitoring systems used in petroleum infrastructure protection services, such as pipelines, pumps, and tank farms, where malicious interventions can cause explosions, fires, or toxic releases, with incalculable economic and environmental consequences. Specifically, a deep spiking neural network anomaly detection method is presented, which models the spike sequences and the internal presentation mechanisms of the information to discover with very high accuracy anomalies in vibration analysis systems used in oil infrastructure protection services. This is achieved by simulating the complex structures of the human brain and the way neural information is processed and transmitted. This work uses a particularly innovative form of the Galves–Löcherbach Spiking Model (GLSM) [1], which is a spiking neural network model with intrinsic stochasticity, ideal for modeling complex spatiotemporal situations, which is enhanced with possibilities of exploiting confidence intervals by modeling optimally stochastic variable-length memory chains that have a finite state space.

1. Introduction

The advent of the 4th industrial revolution and the Internet of Things, communication between humans and machines is becoming more evident and functional [1]. The placement of sensors and smart applications and data extraction from the devices during their operation offer more and more accurate data to manage the process and control of the machines adequately. A big problem for industries is the shutdown of their infrastructure, which will cause significant economic damage [2]. The worst-case scenario is lured into unmanageable damaging scenarios, such as leaks and explosions, that can endanger thousands of people's lives or cause enormous ecological disasters. To avoid similar problems, industries add advanced techniques for monitoring the state of their infrastructures, vibration analysis, for example, is a procedure that monitors the operational status of active industrial equipment using data analytic methods [3, 4].

Specifically, vibration analysis is a method that analyzes the amount and frequency of machine vibrations and then utilizes this information to assess the machine's condition. Although the inner workings and formulae required to compute various vibrations may be sophisticated, all functions begin with high-speed laser sensors to capture and quantify them appropriately. Every time the machine operates, vibrations are generated. Laser sensors connected to the device produce a signal corresponding to the amount and frequency of the vibrations generated. All acquired data are sent immediately to the data collector, which records specific characteristics that can be used to determine infrastructure capacity [5].

Specifically, by attempting a technical evaluation of the characteristics of vibration analysis in industrial equipment, working conditions, and the features of vibrating elements, three main fields of interest are identified that can be exploited to extract useful information, namely:

(1) Time Domain: the physical amount of vibration received by the laser sensors is converted into an electrical signal in the time domain and appears as a waveform.

(2) Frequency Domain: by performing analysis on the waveforms, using frequencies versus amplitude, that is, spectrum analysis, we get the most thorough analysis of machine vibration.

(3) Joint Domain: when the vibration signals change over time, it is helpful to compute more than one spectrum simultaneously. This is done using the standard time approach, Gabor-Wigner-Wavelet, which calcifies variants of the fast Fourier transform, including the short-time Fourier transform.

This procedure is a classical signal processing method that can condense measurements to extract information about some distant state of nature. From this point of view, signal processing can be described from different perspectives. To an acoustician, it is a tool to turn measured signals into useful information. To a sonar designer, it is one part of a sonar system. To an electrical engineer, it is often restricted to digitization, sampling, filtering, and spectral estimation. Mechanics and vibration process analyze these data to determine the machine's operating condition and identify potentially dangerous problems such as looseness, imbalance, misalignment, and lubrication issues. Unbalance, misfires, mechanical looseness, misalignment, tuning, motor faults, bent shafts, bearing failures, voids or bubbles in pumps, and critical speeds or environmental conditions, in particular, may be detected via vibration analysis [6].

One of the critical digital security tools of petroleum infrastructure, which extends and enhances vibration analysis systems, is anomaly detection systems. These systems are called upon to solve the complicated problem of identifying vibrations from abnormal events. As can be seen, these systems should be able to understand the underlying distribution of the data and single out outliers, which may be very few compared to the whole, but are of great importance.

So, the ability to formally express the dependencies between given multivariate events and to reason about the different states of the system over time is of great importance in the anomaly systems in question. This lies in the fact that infrequent events can provide precise expressions of patterns that can inform the system's future behavior and facilitate its more general supervision. As a result, it is critical to implement an event correlation system, which will also provide a common framework for representing the internal dynamics of a time series of events, especially the events associated with failure time data [7].

Standard probabilistic logics and corresponding tools provide reasoning over uncertain data, allowing the annotation of crucial facts with a probability value and using rules. However, in most cases, this is insufficient to express temporal correlations between observed patterns. To represent uncertain data and time dependencies, probabilistic temporal logic programming paradigms have been proposed in the literature, which is simple enough to model special situations. In contrast, improved precision data analysis technologies, such as neural network techniques, can use complex data such as vibration analysis data to extract a pattern or predict a future trend. These analyses are difficult to perform due to human observation and experience complexity [8]. The most important, perhaps, a disadvantage of simple artificial neural networks is the weakness they present in understanding their operation and the fact that the use of the mathematical relationships they implement does not necessarily guarantee that the neural network works efficiently, particularly in complicated circumstances where the time domain plays a vital role. This is because simple neural networks learn and train through a series of examples that are input as templates, automatically making the process of adequately selecting input data significantly train the neural network. For example, a severe simplifying assumption of simple neural networks concerns the view that the neural code used to exchange information between neurons is based on the average value of emitted spikes, a fact modeled as the propagation of continuous variables from one computing unit to another. But, it has been shown experimentally that not only is there not a constant propagation of spikes from which it is tentative to obtain their average value but also that the spikes appear periodically after the application of action and that the exact time of the spikes plays a significant role, if not the more critical role in neural information processing [9].

Spiking neural networks, on the other hand, allow for the thorough analysis and modeling of temporally determined information, utilizing important information such as neuronal firing rate, the relationship between a stimulus and individual or aggregate neural responses, the association between neuronal electrical activity as a whole them, and also the processes of polarization or depolarization of neuronal activity. These characteristics carry all the necessary elements for transmitting, analyzing, and utilizing information to the maximum extent. Such approaches extend the syntax and semantics of probabilistic logic programs, allowing reasoning about probabilities of points over time intervals using probabilistic time rules [10].

A complex spiking neural architecture is employed in this study to cover and avoid the simplifying assumptions and extensions of current essential neural network technologies. Specifically, a deep spiking neural network anomaly detection method is presented, which models the spike sequences and internal presentation mechanisms of the information to detect anomalies in vibration analysis systems used in oil infrastructure protection services with very high accuracy by simulating most realistically the complex structures of the human brain and the way neural information is processed and transmitted. In particular, a specialized GLSM is presented, which is ideal for modeling complex spatiotemporal situations, enhanced with confidence interval capabilities.

2. State-of-the-art

Recently, many studies have used spiking neural networks [11] in practical applications [12], the results of which show promise in the solution of real complex problems. Significant progress with their use has been made in areas such as speech recognition, machine vision, computer system security, complex learning systems from heterogeneous agents, and mechanisms that utilize associative memory and robotics.

In their study, Bariah et al. [1] utilized the characteristics of the Spiking Neural Network to construct an appropriate detector, address the problem of acquiring complicated features, collecting actionable data, and discriminating normal from differential expression. Throughout their training, they developed possible neurons, which spiked once they identified an abnormal pattern in the data. Their method was composed of three steps: implementing the weight values with the rank order method, expressing the real input data as peak values with Gaussian Receptive Fields, and finding the firing nodes that indicated anomalous data. They extended their method to anomalous data extracted from time series datasets. The experimental findings demonstrated that the proposed method could detect anomalies in information with an acceptable Classification Error Rate.

Demertzis et al. [11] published in 2017 an enhanced Spiking One-Class Anomaly Detection Framework relying on the developing Spiking Neural Network technique, which enabled a novel implementation of the one-class supervised classification. Because it is trained solely with data describing the usual function of an Industrial Control System, it is able to identify deviating trends and anomalies related to Advanced Persistent Threat campaigns. They logically structured information in a spatiotemporal fashion while simulating the activity of organic brain cells in the most practical style. Length and intensity of time bursts between neurons are essential parameters in the transmission of generated signals. In addition, they incorporated AI technology at the level of real-time evaluation of industrial machinery, which significantly strengthens the protective mechanisms of vital infrastructures, managing the interrelations of ICS at all times and making it much simpler to detect APT attempts, as they discovered.

Stratton et al. [13] examined Spiking Neural Networks by employing text stream anomaly detection. They demonstrated that SNNs are well-suited for recognizing unusual strings, that they might learn quickly, and that numerous SNN architectural and learning modifications can increase anomaly detection efficiency. Anomaly detection must be automated in order to manage big quantities of information and meet real-time processing restrictions. Spiking Neural Networks offer the ability to perform well with AD, particularly for edge applications where it must be limited, easily adaptive, independent, and dependable. Upcoming research will employ more comprehensive and difficult training datasets and directly compare the performance of SNNs and Deep Neural Networks educated on a same dataset using systems of comparable size.

Utilizing spiking neural networks, Dennler et al. [10] suggested a neuromorphic method for dynamic analysis that may be used to a variety of circumstances. Vibration patterns provide vital data on health condition of operating equipment, which is typically utilized in preventative maintenance duties for big manufacturing systems. Nonetheless, the scale, sophistication, and power budget needed by conventional ways to exploit this information are often too expensive for various applications such as selfdriving automobiles, uncrewed aerial vehicles, and automation. They created a spike-based end-to-end system that works unsupervised online to identify system abnormalities using vibration data, leveraging building blocks compatible with analog-digital artificial neural circuits. They proved that the suggested approach met or surpassed state-of-the-art performance on two publicly available datasets. In addition, they implemented a proof-of-concept on an asynchronous artificial neural processor device, advancing the development and operation of independent reduced end devices for continuous vibration analysis.

Maciag et al. [14] found anomalies in stream data without supervision as the primary study subject in their research work. Specifically, they proposed an Online Growing Spiking Neural Network for Unsupervised Anomaly Detection method. Unlike the Online Growing Spiking Neural Network, it worked unsupervised and did not partition output neurons into discrete judgment classes. However, collecting adequate instruction data with labeled irregularities for labeled data of an automated tracking that may subsequently be implemented to spot actual abnormalities in data in real-time is challenging in many cases. As a result, it is critical to building anomaly detectors that can identify abnormalities even in the absence of labeled training data. To identify abnormalities, they used a two-step technique. The proposed detector outperforms existing methods published in the literature for data streams in experimental comparisons with state-of-the-art unassisted and semisupervised sensors of deviations in datasets from known dataset repositories.

Using an adjusted evolving Spiking Neural Network, Dennis et al. [9] presented a model for detecting anomalies in flowing multivariate time series. They contributed a substitute rank-order-based autoencoder that used the precise times of incoming spikes for adjusting network parameters, an adapted, real-time-capable, and reliable learning algorithm for multivariate data based on multidimensional Gaussian Receptive Fields, and a constant outlier scoring function for enhanced interpretability of the classifications. The potential applications for this type of algorithm are diverse. It extends from tracking digital machinery and preventative maintenance to big data healthcare datalogger analysis applications. Spiking algorithms are particularly effective in time-dependent information processing. They showed the prototype's effectiveness on a synthetic dataset based on a reference point containing various types of anomalies, comparing it to other streaming anomaly detection methodologies and demonstrating that their algorithm performed better at detecting anomalies while requiring fewer computational resources for processing high-dimensional data.

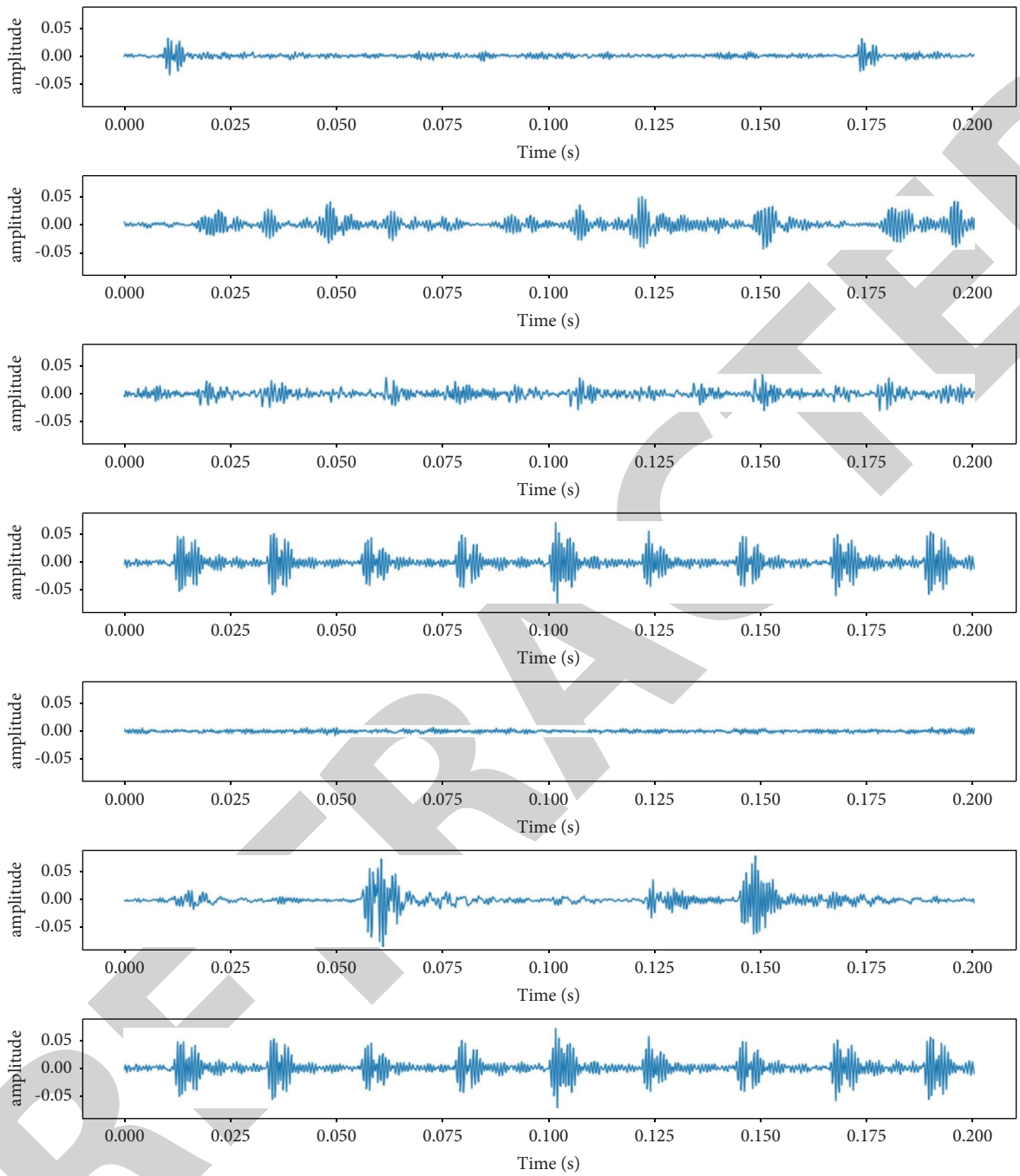


Figure 1: Vibration data.

3. Vibration Analysis Data

Failures related to bearings and lubrication systems are one of the leading causes of forced outages in turbine systems [15]. In some cases, machinery can explode, causing extensive damage to other equipment and even human casualties. Transverse vibrations in high-speed shafts, mainly in turbine engines, can capture the coexistence of transverse cracking and wear in bearings under certain conditions. A shaft is considered a high-speed shaft if it rotates at a speed greater than the critical speed, that is, a speed at which transverse vibrations of significant magnitude occur. This analysis focuses on motion oscillation problems in the critical and postcritical speed ranges, determining the frequency ranges for natural vibrations and critical speeds, and evaluating stability, specifically, stability within the case-specific critical range [16].

In particular, the analysis considers the case of the rotor bearing system using boundary conditions that combine the rotor shear force and the fluid frame (flange) forces at the points where the bearings contact. The behavior of the bearing is assumed to be nonlinear since its dynamic

properties are captured by the actual function of its periodic position and linear velocity at a given time. It should be noted that material damping is introduced into the rotor model to achieve time-domain solutions, even at critical speeds, independent of the length of time the system is in resonance. In such cases, the analysis gives response results too close to or even higher than the radial distance and is considered insufficient. The model has an initial condition with time-dependent boundary values. This is obtained by expressing flange forces as a function of rotor force shear [17].

The initial depiction of the operating conditions at random times is shown diagrammatically in Figure 1.

The main objective of the current work is to investigate the system's dynamics with characteristics in both the frequency and time domains under the assumption of a continuous rotor in nonlinear conditions of bearing operations [18].

The approach looks for elements such as rotor motion, rotational inertia, punching deformation, and torque due to power transmission, considering the gyroscopic effect. Specifically, the function that is considered for the vertical movement is [19]:

$$E^* I_j \frac{\partial^4 Y_j}{\partial x^4} - \frac{E^* I_j \rho}{kG^*} \frac{\partial^4 Y_j}{\partial x^2 \partial t^2} + T \frac{\partial^3 Z_j}{\partial x^3} - \frac{T\rho}{kG^*} \frac{\partial^3 Z_j}{\partial x \partial t^2} + \rho A_j \frac{\partial^2 Y_j}{\partial t^2} - \rho A_j r_0^2 \left[\left(\frac{\partial^4 Y_j}{\partial x^2 \partial t^2} - \frac{\rho}{kG^*} \frac{\partial^4 Y_j}{\partial t^4} \right) + 2\Omega \left(\frac{\partial^3 Z_j}{\partial x^2 \partial t} - \frac{\rho}{kG^*} \frac{\partial^3 Z_j}{\partial t^3} \right) \right] = 0. \tag{1}$$

While for the horizontal movement, it is the:

$$E^* I_j \frac{\partial^4 Z_j}{\partial x^4} - \frac{E^* I_j \rho}{kG^*} \frac{\partial^4 Z_j}{\partial x^2 \partial t^2} - T \frac{\partial^3 Y_j}{\partial x^3} + \frac{T\rho}{kG^*} \frac{\partial^3 Y_j}{\partial x \partial t^2} + \rho A_j \frac{\partial^2 Z_j}{\partial t^2} - \rho A_j r_{0_j}^2 \left[\left(\frac{\partial^4 Z_j}{\partial x^2 \partial t^2} - \frac{\rho}{kG^*} \frac{\partial^4 Z_j}{\partial t^4} \right) - 2\Omega \left(\frac{\partial^3 Y_j}{\partial x^2 \partial t} - \frac{\rho}{kG^*} \frac{\partial^3 Y_j}{\partial t^3} \right) \right] = 0. \tag{2}$$

Given the:

Complex Young modulus:

$$E^* = E(1 + i \cdot \eta). \tag{3}$$

Complex shear modulus:

$$G^* = G \cdot (1 + i \cdot \eta). \tag{4}$$

The radius of gyration of each step:

$$r_{0_j} = \sqrt{\frac{I_j}{A_j}}. \tag{5}$$

Boundary conditions are constraints necessary to solve a boundary value problem. A boundary value problem is a differential equation (or system of differential equations) to be solved in a domain on whose boundary a set of conditions is known. Aside from the boundary condition, boundary value problems are also classified according to the type of differential operator involved. These categories are further subdivided into linear and various nonlinear types. For an elliptic operator, one discusses elliptic boundary value problems. For a hyperbolic operator, one discusses hyperbolic boundary value problems.

Many significant problems, such as flow driven by moving objects, free-surface flow, flow involving air bubbles, flow accompanying phase transition, and fluid-structure interaction, are moving boundary problems. In dealing with such boundaries with movement or deformation, the traditional mesh methods such as finite difference, finite element, and finite volume method generally encounter difficulties in accurately calculating the geometric form of the border.

Numerous numerical analyses have been performed to study problems with large interfacial deformation, such as free-surface and multiphase flows. However, besides the aforementioned valuable characteristics, the vibration calculation algorithm of the particle method also has a severe negative aspect: difficulty in the treatment of fixed boundaries (e.g., solid walls). In the mesh-free framework of calculation, spatial derivatives of the physical quantities (e.g., velocity and pressure) and the value of the particle number density are calculated by referencing the relative positions of surrounding particles that are present inside the practical domain. Still, the affective domain is truncated by the actual boundary for particles near the real boundary.

In our approach, boundary conditions express the continuity and discontinuity of shear force, bending moment, pitch, and displacement of the rotor depending on where the boundary exists. Specifically, the following boundary conditions are expressed in equations:

Slope:

$$S_{Y_{j,R}}(x,t) = \frac{\partial}{\partial x}\left(Y_{j,R}(x,t)\right), S_{Y_{J,I}}(x,t) = \frac{\partial}{\partial x}\left(Y_{j,I}(x,t)\right),$$
$$S_{Z_{J,R}}(x,t) = \frac{\partial}{\partial x}\left(Z_{j,R}(x,t)\right), S_{Z_{J,I}}(x,t) = \frac{\partial}{\partial x}\left(Z_{j,I}(x,t)\right). \tag{6}$$

Bending moment:

$$M_{Y_{J,R}}(x,t) = EI_j \frac{\partial^2}{\partial x^2}\left(Y_{j,R}(x,t)\right) - EI_j \eta \frac{\partial^2}{\partial x^2}\left(Y_{j,I}(x,t)\right),$$
$$M_{Y_{J,I}}(x,t) = EI_j \frac{\partial^2}{\partial x^2}\left(Y_{j,I}(x,t)\right) + EI_j \eta \frac{\partial^2}{\partial x^2}\left(Y_{j,R}(x,t)\right),$$
$$M_{Z_{J,R}}(x,t) = EI_j \frac{\partial^2}{\partial x^2}\left(Z_{j,R}(x,t)\right) - EI_j \eta \frac{\partial^2}{\partial x^2}\left(Z_{j,I}(x,t)\right),$$
$$M_{Z_{J,I}}(x,t) = EI_j \frac{\partial^2}{\partial x^2}\left(Z_{j,I}(x,t)\right) + EI_j \eta \frac{\partial^2}{\partial x^2}\left(Z_{j,R}(x,t)\right). \tag{7}$$

Shearing force:

$$V_{Y_{J,R}}(x,t) = EI_j\left(\frac{\partial^3 Y_{j,R}(x,t)}{\partial x^3} - \eta\frac{\partial^3 Y_{j,I}(x,t)}{\partial x^3}\right) - \rho A_j r_{0,j}^2\left(G_{1,R}(x,t) + 2\Omega G_{2,R}(x,t)\right),$$
$$V_{Y_{J,t}}(x,t) = EI_j\left(\eta\frac{\partial^3 Y_{j,R}(x,t)}{\partial x^3} + \frac{\partial^3 Y_{j,I}(x,t)}{\partial x^3}\right) - \rho A_j r_{0j}^2\left(G_{1_{j,t}}(x,t) + 2\Omega G_{2,t}(x,t)\right),$$
$$V_{Z_{J,R}}(x,t) = EI_j\left(\frac{\partial^3 Z_{j,R}(x,t)}{\partial x^3} - \eta\frac{\partial^3 Z_{j,I}(x,t)}{\partial x^3}\right) - \rho A_j r_{0j}^2\left(G_{3_{j,R}}(x,t) + 2\Omega G_{4_{4,R}}(x,t)\right),$$
$$V_{Z_{J,I}}(x,t) = EI_j\left(\eta\frac{\partial^3 Z_{j,R}(x,t)}{\partial x^3} + \frac{\partial^3 Z_{j,I}(x,t)}{\partial x^3}\right) - \rho A_j r_{0j}^2\left(G_{3_{j,I}}(x,t) + 2\Omega G_{4_{,n}}(x,t)\right). \quad (8)$$

Using finite frontal differences of the first order in the time domain, the terms are expressed as shown in the following equations:

Real vertical slope:

$$G_{1_{j,R}}(x,t) = \frac{S_{Y_{j,R}}(x,t) - 2S_{Y_{j,R}}(x,t-\Delta t) + S_{Y_{j,R}}(x,t-2\Delta t)}{\Delta t^2}. \quad (9)$$

Imaginary vertical slope:

$$G_{1_{j,t}}(x,t) = \frac{S_{Y_{j,t}}(x,t) - 2S_{Y_{j,t}}(x,t-\Delta t) + S_{Y_{j,t}}(x,t-2\Delta t)}{\Delta t^2}. \quad (10)$$

Real vertical slope velocity:

$$G_{2_{j,R}}(x,t) = \frac{S_{Y_{j,R}}(x,t) - S_{Y_{j,R}}(x,t-\Delta t)}{\Delta t}. \quad (11)$$

Imaginary vertical slope velocity:

$$G_{2_{j,I}}(x,t) = \frac{S_{Y_{j,I}}(x,t) - S_{Y_{j,I}}(x,t-\Delta t)}{\Delta t}. \quad (12)$$

Real horizontal slope:

$$G_{3_{j,R}}(x,t) = \frac{S_{Z_{j,R}}(x,t) - 2S_{Z_{j,R}}(x,t-\Delta t) + S_{Z_{j,R}}(x,t-2\Delta t)}{\Delta t^2}. \quad (13)$$

Imaginary horizontal slope:

$$G_{3_{j,t}}(x,t) = \frac{S_{Z_{j,t}}(x,t) - 2S_{Z_{j,t}}(x,t-\Delta t) + S_{Z_{j,t}}(x,t-2\Delta t)}{\Delta t^2}. \quad (14)$$

Real horizontal slope velocity:

$$G_{4_{j,R}}(x,t) = \frac{S_{Z_{j,R}}(x,t) - S_{Z_{j,R}}(x,t-\Delta t)}{\Delta t}. \quad (15)$$

Imaginary horizontal slope velocity:

$$G_{4_{j,I}}(x,t) = \frac{S_{Z_{j,I}}(x,t) - S_{Z_{j,I}}(x,t-\Delta t)}{\Delta t}. \quad (16)$$

Real vertical acceleration:

$$G_{5_{j,R}}(x,t) = \frac{Y_{j,R}(x,t) - 2Y_{j,R}(x,t-\Delta t) + Y_{j,R}(x,t-2\Delta t)}{\Delta t^2}. \quad (17)$$

Imaginary vertical acceleration:

$$G_{5_{j,I}}(x,t) = \frac{Y_{j,I}(x,t) - 2Y_{j,I}(x,t-\Delta t) + Y_{j,I}(x,t-2\Delta t)}{\Delta t^2}. \quad (18)$$

Real horizontal acceleration:

$$G_{6_{j,R}}(x,t) = \frac{Z_{j,R}(x,t) - 2Z_{j,R}(x,t-\Delta t) + Z_{j,R}(x,t-2\Delta t)}{\Delta t^2}. \quad (19)$$

Imaginary horizontal acceleration:

$$G_{6_{j,I}}(x,t) = \frac{Z_{j,I}(x,t) - 2Z_{j,I}(x,t-\Delta t) + Z_{j,I}(x,t-2\Delta t)}{\Delta t^2}. \quad (20)$$

Real boundary conditions.

Vertical plane bearings:

$$\sigma_1 = M_{Y_{1,R}}(0,t),$$
$$\sigma_2 = M_{Y_{2,R}}(L,t),$$
$$\sigma_3 = F_{Y_{1,R}} - EI_1\frac{\partial^3 Y_{ST_1}}{\partial x^3}\bigg|_{x=0} - V_{Y_{1,R}}(0,t),$$
$$\sigma_4 = F_{Y_{2,R}} + EI_2\frac{\partial^3 Y_{ST_2}}{\partial x^3}\bigg|_{x=L} + M_{Y_{2,R}}(L,t). \quad (21)$$

Location of floor:

$$\sigma_5 = M_{Y_{2,R}}(L_1,t) - M_{Y_{1,R}}(L_1,t),$$
$$\sigma_6 = M_{Y_{2,R}}(L_1,t) - M_{Y_{1,R}}(L_1,t) + I_P\Omega G_{2_{2,R}}(L_1,t) + I_T G_{3_{2,R}}(L_1,t),$$
$$\sigma_7 = V_{Y_{2,R}}(L_1,t) - V_{Y_{1,R}}(L_1,t) - F_{u_Y} + \mathrm{E_F} + m_d G_{5_{2,R}}(L_1,t),$$
$$\sigma_8 = Y_{2,R}(L_1,t) - Y_{1,R}(L_1,t). \quad (22)$$

Horizontal plane bearings:

$$\sigma_9 = M_{Z_{1,R}}(0,t),$$
$$\sigma_{10} = M_{Z_{2,R}}(L,t),$$
$$\sigma_{11} = F_{Z_{1,R}} - V_{Z_{1,R}}(0,t),$$
$$\sigma_{12} = F_{Y_{2,R}} + V_{Z_{2,R}}(L,t). \quad (23)$$

Location of floor:

$$\begin{aligned}
\sigma_{13} &= S_{Z_{2,R}}(L_1,t) - S_{Z_{1,R}}(L_1,t),\\
\sigma_{14} &= M_{Z_{2,R}}(L_1,t) - M_{Z_{1,R}}(L_1,t)\\
&\quad + I_P\Omega G_{4_{2,R}}(L_1,t) - I_T G_{1_{2,R}}(L_1,t),\\
\sigma_{15} &= V_{Z_{2,R}}(L_1,t) - V_{Z_{1,R}}(L_1,t)\\
&\quad - F_{u_Z} + m_d G_{6_{2,R}}(L_1,t),\\
\sigma_{16} &= Z_{2,R}(L_1,t) - Z_{1,R}(L_1,t)\cdot
\end{aligned} \tag{24}$$

Imaginary boundary conditions.
Vertical plane bearings:

$$\begin{aligned}
\sigma_{17} &= M_{Y_{1,t}}(0,t),\\
\sigma_{18} &= M_{Y_{2,t}}(L,t),\\
\sigma_{19} &= F_{Y_{,,}} - EI_1 \frac{\partial^3 Y_{ST_1}}{\partial x^3}\bigg|_{x=0} - V_{Y_{1,t}}(0,t),\\
\sigma_{20} &= F_{Y_{2,I}} + EI_2 \frac{\partial^3 Y_{ST_2}}{\partial x^3}\bigg|_{x=L} + V_{Y_{2,I}}(L,t)\cdot
\end{aligned} \tag{25}$$

Location of floor:

$$\begin{aligned}
\sigma_{21} &= S_{Y_{2,t}}(L_1,t) - S_{Y_{1,t}}(L_1,t),\\
\sigma_{22} &= M_{Y_{2,i}}(L_1,t) - M_{Y_{1,t}}(L_1,t)\\
&\quad + I_P\Omega G_{2_{2,t}}(L_1,t) + I_T G_{3_{2,t}}(L_1,t),\\
\sigma_{23} &= V_{Y_{2,i}}(L_1,t) - V_{Y_{1,t}}(L_1,t)\\
&\quad - F_{u_Y} + E_F + m_d G_{S_{2,i}}(L_1,t),\\
\sigma_{24} &= Y_{2,I}(L_1,t) - Y_{1,I}(L_1,t)\cdot
\end{aligned} \tag{26}$$

Horizontal plane bearings:

$$\begin{aligned}
\sigma_{25} &= M_{Z_{1,I}}(0,t),\\
\sigma_{26} &= M_{Z_{2,I}}(L,t),\\
\sigma_{27} &= F_{Z_{1,I}} - V_{Z_{1,I}}(0,t),\\
\sigma_{28} &= F_{Y_{2,I}} + V_{Z_{2,I}}(L,t)\cdot
\end{aligned} \tag{27}$$

Location of floor:

$$\begin{aligned}
\sigma_{29} &= S_{Z_{2,I}}(L_1,t) - S_{Z_{1,I}}(L_1,t),\\
\sigma_{30} &= M_{Z_{2,I}}(L_1,t) - M_{Z_{1,I}}(L_1,t)\\
&\quad + I_P\Omega G_{4_{2,I}}(L_1,t) - I_T G_{1_{2,I}}(L_1,t),\\
\sigma_{31} &= V_{Z_{2,I}}(L_1,t) - V_{Z_{1,I}}(L_1,t)\\
&\quad - F_{u_Z} + m_d G_{6_{2,I}}(L_1,t),\\
\sigma_{32} &= Z_{2,I}(L_1,t) - Z_{1,I}(L_1,t)\cdot
\end{aligned} \tag{28}$$

Before extracting features, the recorded signals received appropriate preprocessing, intending to suppress unwanted distortions and enhance the most critical features for further processing [20]. Specifically, first, the signals were filtered using a digital bandpass filter of order 5 with a range of 0.5-30 Hz. We use low frequencies because the parts of the signals that contain essential information may be in lower frequency bands. Also, since the effect of filtering on the signals is weighted, that is, greater in the center of the filter values, not all the interference of the 50 Hz band is neutralized. As a result, bandpass filtering must be applied to the signals immediately after bandpass filtering. A band-stop filter with a narrow rejection band and a high-quality factor $Q = 30$ is used to attenuate frequencies near 50 Hz. Accordingly, after filtering, we proceed to normalize the signals. For each trial and electrode, the average of the signal is subtracted from each time sample. The result of the previous operation is divided by the standard deviation, as shown by the following equation:

$$x_i^*(t) = \frac{x_i(t) - \overline{x}_i}{SD(x)_i}. \tag{29}$$

To calculate the band power characteristics for each of the three channels containing useful information C3, C4, and CZ, the parts of the signal corresponding to the quiescent state are trimmed based on the quiescent times. On the resulting signals after clipping, we apply bandpass filtering to 72 frequency bands using different overlapping narrow bands between 8 Hz and 30 Hz. The characteristics are calculated by subtracting the power values of the resting parts from the power values of the parts corresponding to vibrations. Then, we apply bandpass filtering to the resting window using a digital filter of order 5 at 8-12 Hz and 13-25 Hz, since the power during the resting state serves as a reference point in both frequency bands. The segments are also filtered with the same filter at the frequencies 8-12 Hz and 13-30 Hz, respectively. We get the power samples from the parts of the original signal that have been cut and filtered by squaring the available amplitude samples. To calculate the final features, we subtract the average power of a trial during the windows from the filtered resting time intervals. Similarly, the mean of single-trial power samples during rest windows is subtracted from the band-filtered rest intervals [21].

Finally, for pattern estimation and feature distribution, it is approximated by the logarithm of the variance of a bandpass filtered signal at specific time intervals so that [22]:

$$u_c(i) = \frac{2}{\tau - 1} \sum_{j=t_n}^{k_n+\tau-1} \left(x_y - \overline{x}_i\right)^2\cdot \tag{30}$$

For each channel, the logarithmic band power characteristic is defined by the following formula:

$$BP_c(i) = \log\ \left(u_e(i)\right)\cdot \tag{31}$$

Each class and each channel are calculated by taking the median of the average data variances for all trials. We use the median as it is more robust to extreme values. The total band power is written as follows:

$$\widetilde{BP}_c^X = \log\ \left(\widetilde{u}_e\right)\cdot \tag{32}$$

Thus, we can now define the Pattern Difference PDc for channels C3 and C4 between the two problem classes N (normal) and A (abnormal) [23]:

$$\begin{aligned} PD_{C3} &= \widetilde{BP}^{N}_{C3} - \widetilde{BP}^{A}_{C3}, \\ PD_{C4} &= \widetilde{BP}^{N}_{C4} - \widetilde{BP}^{A}_{C4}. \end{aligned} \tag{33}$$

When PD pattern difference is calculated on small subbands of a dataset, the results show in which frequency bands the reduction in band power due to vibrations is most prominent. Since the specific band varies between subjects and recording sessions, results will vary accordingly along the band.

4. Proposed Deep Spiking Neural Architecture

The suggested application's primary goal is anomaly identification, which finds unusual occurrences, features, or observations that are unusual because they deviate considerably from typical patterns or behaviors. Anomalies in the data should be separated by an algorithmic system that can identify and appropriately classify standard deviations associated with marginal uses of the equipment relative to outliers, noise, novel uses, and exceptions. The modeling in question requires a neural architecture where the exchange of information between random neurons occurring at random synapses can create a specialized functional structure connecting neurons through random discrete events. Presynaptic and postsynaptic neurons can alternate their functions based on probabilistic functions. A neuron can simultaneously be presynaptic to some synapses and postsynaptic to others, depending on how it participates in them.

Based on the above requirements, a particularly innovative form of the Galves–Löcherbach Spiking Model (GLSM) [1] is used in this work, which is a spiking neural network model with inherent stochasticity, ideal for modeling complex spatiotemporal situations. A countable number of elements (idealized neurons) interact with sporadic near-instantaneous discrete events in the proposed GLSM (spikes or firings). Each neuron N fires independently at each moment, with a probability determined by the firing history of all neurons since the last time N fired. As a result, whenever a neuron fires, it forgets all initial spikes, including its own. This attribute represents the model's distinguishing trait. Specifically, we consider a stochastic chain $(X_t)_{t\in\mathbb{Z}}$ that takes values in $\{0, 1^I\}$ for some measurable set of neurons I, defined in a suitable probability space (Ω, A, P). For each neuron i at each time $t \in \mathbb{Z}$, $X_t(i) = 1$, if neuron i is spiking at time t and $X_t(i) = 0$ otherwise. The global configuration of neurons at time t is denoted $X_t = (X_t(i), i \in I)$. Filtering is defined as follows:

$$\mathscr{F}_t = \sigma\left(X_s, s \in \mathbb{Z}, s \le t\right), t \in \mathbb{Z}\cdot \tag{34}$$

For each neuron $i \in I$, for each instant $t \in Z$ holds:

$$L^i_t = \sup\{s < t\colon X_s(i) = 1\}\cdot \tag{35}$$

Which also translates as the last spike time of neuron i strictly before time t.

Then, we introduce a family of synaptic weights $W_{j\longrightarrow i} \in \mathbb{R}$, for $j \ne i, W_{j\longrightarrow j} = 0$ for all j. $W_{j\longrightarrow i}$ is the synaptic weight of neuron j to neuron i. We assume that the synaptic weights have the following uniform summation property [24]:

$$\sup_{i\in I} \sum_j \left|W_{j\longrightarrow i}\right| < \infty\cdot \tag{36}$$

At each time t (which is conditionally valid for the entire past), the domains are updated independently, which means that for any finite subset $J \subset I$, $a_i \in \{0, 1\}, i \in J$, we have:

$$P\left(X_t(i) = a_i, i \in J \mid \mathscr{F}_{t-1}\right) = \prod_{i\in J} P\left(X_t(i) = a_i \mid \mathscr{F}_{t-1}\right)\cdot \tag{37}$$

Additionally, the probability of having a spike in neuron i at time t is given by the relation:

$$P\left(X_t(i) = 1 \mid \mathscr{F}_{t-1}\right) = \phi_i\left(\sum_j W_{j\longrightarrow i} \sum_{s=L^i_t}^{t-1} g_j(t-s) X_s(j), t - L^i_t\right), \tag{38}$$

where $\phi_i\colon \mathbb{R}\times\mathbb{N} \longrightarrow [0, 1]$ and $g_j\colon \mathbb{N} \longrightarrow \mathbb{R}_+$ are measurable functions for all $i \in I, j \in I$. We assume that ϕ_i is uniformly continuous, that is, there exists a positive constant γ such that for all $s, s' \in \mathbb{R}, n \in \mathbb{N}, i \in I$ to be valid [25]:

$$\left|\phi_i(s, n) - \phi_i(s', n)\right| \le \gamma\left|s - s'\right|\cdot \tag{39}$$

The probability of a spike in the next time unit depends on the system's overall time evolution after the component's last spike time. We consider that the functions ϕ_i and g_j additionally satisfy the following assumptions and specifically there exists $\delta > 0$ such that for all $i \in I, s \in R, n \in N$ [26]:

$$\phi_i(s, n) \ge \delta\cdot \tag{40}$$

So that:

$$G(1) + \sum_{n=2}^{\infty} (1-\delta)^{n-2} n^2 G(n) < \infty, \tag{41}$$

where $G(n) = \sup_i \sum_{m=1}^{n} g_i(m)$.

So, in this case we have a fast decomposition of the synaptic weights, that is, [27]:

$$\sup_i \sum_{k\ge 1} \left|V_i(k)\right| \left(\sum_{j\notin V_i(k-1)} \left|W_{j\longrightarrow i}\right|\right) < \infty\cdot \tag{42}$$

As can be reasonably expected, the results of applying categorization to a data stream are the set of tuples that would result from the union of the individual results that the application would have on the current contents of the stream at any time. Specifically, let Q be a categorization query submitted at a time $\tau_o \in T$ to the data stream S. Then, the results Q^c to be obtained at time $\tau_i \in T$ are the set of tuples $Q(S(\tau))$ that satisfies the query Q from every current content $S(\tau)$ of the stream until then, that is, [28]:

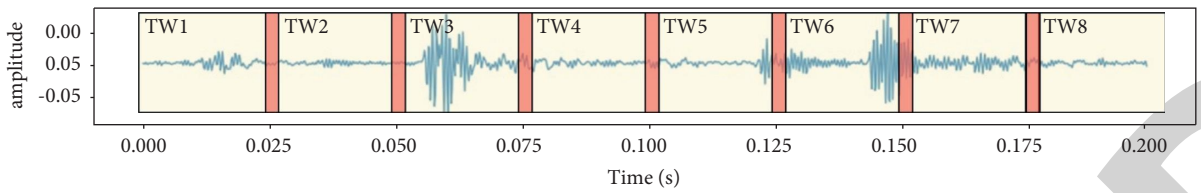


Figure 2: Time windows of vibration analysis.

$$\forall \tau_i \in T, Q^c\left(S\left(\tau_i\right)\right) = \% \underset{\tau_0 \le \tau \le \tau_i}{\cup} Q(S(\tau)). \tag{43}$$

Obviously, the total valuation for all successive time points is practically unprofitable. So, a strategy is needed that allows for periodic checking of the data so that the answers are calculated periodically. Since all tuples are marked with distinct time milestones, it is possible to evaluate Q with a time step Δ_τ. In the k-th iteration, only the intermediate results $Q^c\left(S\left(\tau_O + k\Delta\tau\right)\right) - Q^c\left(S\left(\tau_O + (k-1)\Delta\tau\right)\right)$ should be added to the current answer replacing the immediately preceding one, as follows:

$$\Delta Q = \% \underset{\tau_0 \le \tau \le \tau_0 + k\Delta\tau}{\cup} Q(S(\tau)) - \% \underset{\tau_0 \le \tau \le \tau_0 + (k-1)\Delta\tau}{\cup} Q(S(\tau)). \tag{44}$$

However, the previous formula does not always guarantee correct results. One only must think about what will happen if, over time, newer answers cancel out earlier ones. For example, we consider that all vibrations whose value variation has never exceeded the general threshold index can be requested in the anomaly detection application. But as the data changes dynamically, some tuples included in previous answers may no longer meet the criterion. However, with periodic computation, such results will be retained in all subsequent responses, even though they should typically be discarded.

The following Figure 2 depicts an example of time windows of vibration analysis.

An important innovation in implementing the proposed algorithm is the extension of the GLSM probabilistic algorithm, with the introduction of time intervals and confidence points in the cases where we have samples of independent and identically distributed variables and correspondingly when we have a Markov chain [29]. This extension leads to a system that can extract probabilistic recognition of complex events over time from a stream of low-level events. Accordingly, by introducing time windows and working memory, we modified the existing GLSM approach, where we consider a new class of non-Markovian processes with a countable number of interacting elements, in which and for each time unit, each element can take two values (normal or abnormal), indicating the anomaly state at the given time.

So, the system expands in a nontrivial way for interacting systems, mainly Markovian, and for stochastic chains with variable-length memory that have a finite state space. These features make the proposed GLSM suitable for describing and modeling the temporal evolution of systems, such as tracking anomalies originating from vibration or vibration analysis systems and their intrinsic mapping from biological patterns of neural systems, using a probabilistic tool in a random environment or spacetime.

The application above is founded on the premise that a snapshot of activity, for example, determined by a snapshot of vibrations, might lead to incorrect identification owing to sensor unreliability or inaccuracy, as well as a variety of extraneous events that can generate noise in the data. In terms of the monitoring and control of the recognition process, such occurrences of misidentification of activities might result in unwarranted delays and slow operations. Therefore, there is a need for a more robust identification that identifies the time intervals within which a high-level activity takes place. As a result, we provide a probabilistic technique for computing events based on time intervals.

Specifically, the probability of an interval $I_{LTA} = [i, j]$ of an LTA activity with length $(I_{LTA}) = j - i + 1$ time instants is defined as [30]:

$$P\left(I_{LTA}\right) = \frac{\sum_{k=i}^{j} P\left(\text{holdsAt}\left(LTA, k\right)\right)}{\text{length}\left(I_{LTA}\right)}. \tag{45}$$

In other words, the probability of an interval equals the average of the probabilities of each instant in time it contains. More generally, a maximum likelihood interval $I_{LTA} = [i, j]$ of an LTA activity is an interval such that, given a probability threshold $\mathcal{T} \in [0, 1]$, $P\left(I_{LTA}\right) \ge \mathcal{T}$ and there is no other interval I'_{LTA} such so that $P\left(I'_{LTA}\right) \ge \mathcal{T}$ and I_{LTA} is a subinterval of I'_{LTA}.

A consequence of the maximum likelihood interval criterion above is that such intervals can overlap. So, we keep only one from each set of overlapping intervals, using interval reliability as a selection criterion. The reliability of an interval is defined as the product of its length times its probability [31]:

$$\begin{aligned} \text{Cred}\left(I_{LTA}\right) &= \text{length}\left(I_{LTA}\right) \cdot P\left(I_{LTA}\right) \\ &= \sum_{k} P\left(\text{holdsAt}\left(LTA, k\right)\right), \end{aligned} \tag{46}$$

where k are the time instants of the interval I_{LTA}. Therefore, for each set of overlapping maximum likelihood intervals $S = \left\{I_1, I_2, \ldots, I_k\right\}$, we choose the one with the largest confidence value, or briefly the interval I_{LTA} with confidence $\text{Cre d}\ \left(I_{LTA}\right) = \max_i \left(\text{Cre d}\left(I_i\right)\right)$ for each $i = 1, \ldots, k$.

It should be noted that when the interactions between the systems are represented by a critically directed random graph with a large but finite number of components, the proposed framework yields an explicit upper bound for the correlation between successive intervals between spikes

that is consistent with previous empirical findings of the process.

Applying the Monte Carlo algorithm [32], we obtain the bootstrap estimator of the standard deviation:

$$\widehat{\sigma}_B = \left(\frac{\sum_{b=1}^{B} \left\{ \widehat{\theta}^*(b) - \widehat{\theta}^*(\cdot) \right\}^2}{B-1} \right)^{1/2}, \tag{47}$$

where:

$$\widehat{\theta}^*(\cdot) = \frac{\sum_{b=1}^{B} \widehat{\theta}^*(b)}{B}. \tag{48}$$

After calculating the estimators using bootstrap, we may construct confidence intervals as follows [33]:

$$\theta \in \left[\widehat{G}^{-1}\left(\frac{\alpha}{2}\right), \widehat{G}^{-1}\left(1-\frac{\alpha}{2}\right) \right], \tag{49}$$

as an approximate $1-\alpha$ central interval for θ. This is the percentile method. Let $\widehat{G}(s)$ be the parametric bootstrap cumulative distribution function of $\widehat{\theta}^*$:

$$\widehat{G}(s) = \mathrm{Prob}_* \left\{ \widehat{\theta}^* < s \right\}, \tag{50}$$

where Prob_* denotes the probability calculated according to the bootstrap distribution of $\widehat{\theta}^*$.

We assume that for every integer n⩾1, the weights W_n are interchangeable. Based on the vector of weights W_n, the generalized bootstrap mean corresponding to this vector will be the following:

$$\overline{X}_{\mathscr{W},n} = \frac{1}{n} \sum_{i=1}^{n} W_{n,i} X_i. \tag{51}$$

So, the weights W_n will satisfy the following conditions:

$$\begin{aligned} &(\mathscr{W}_I) W_{n,i} \geqslant 0, i = 1, 2, \ldots, n, n \geqslant 1, \\ &(\mathscr{W}_{II}) \quad \sum_{i=1}^{n} W_{n,i} = n, \\ &(\mathscr{W}_{III}) \quad \frac{1}{n} \sum_{i=1}^{n} \left(W_{n,i} - 1 \right)^2 \longrightarrow_P c^2 \text{ as } n \longrightarrow \infty. \end{aligned} \tag{52}$$

It should be said that the bootstrap method with a weighted Poisson distribution was used in the proposed application carried out in the context of this work.

The proposed GLSM is implemented in a deep neural architecture with fully connected layers that unfold in time. The sending of information is carried out based on the generation of an action potential in the body of the presynaptic cell. Whenever a spike propagates through the axon, the firing of the neurons causes a series of actions in the postsynaptic cell, while the membrane rapidly equalizes the postsynaptic potential, at which point the membrane is depolarized. Changes in synaptic plasticity are associated with various forms of memory, as well as short or long memory, flash memory, etc.

Training is performed based on the general backpropagation through time (BPTT) approach so that the neural network stabilizes in time by stacking identical copies of trained neurons. In particular, the matrices of the system weights $w_{ij}^{\text{in}}, w_{ij}, w_{ij}^{\text{out}}, w_{ij}^{\text{back}}$ remain the same in all copies of the layers. The training data consists of a time series of input-output samples that take the following form:

$$\begin{aligned} \mathrm{u}(n) &= \left(u_1(n), \ldots, u_K(n)\right)', \mathrm{d}(n) \\ &= \left(d_1(n), \ldots, d_L(n)\right)' n = 1, \ldots, T. \end{aligned} \tag{53}$$

The training process starts from the first level and continues gradually to the next levels of the stack that have been acquired by the unfolding of the levels over time. In each copy of the layers at time n, the input $u(n)$ is introduced, the $x(n)$ of the intermediate layers is calculated based on $u(n), x(n-1)$ and $y(n-1)$ and finally, the output $y(n)$ is calculated. The error function that is being reduced is as follows:

$$E = \sum_{n=1,\ldots,T} \|\mathrm{d}(n) - y(n)\|^2 = \sum_{n=1,\ldots,T} E(n), \tag{54}$$

but the meaning of t has changed from the training instance to time. For each activation of nodes $x(n), y(n)$, the error propagation is given by the following formulas. For the input nodes in the time plane T:

$$\delta_j(T) = \left(d_j(T) - y_j(T)\right) \frac{\partial f(u)}{\partial u}\bigg|_{u=z_j} (T). \tag{55}$$

For the output nodes in time plane T:

$$\delta_i(T) = \left[\sum_{j=1}^{L} \delta_j(T) w_{ji}^{\text{out}} \right] \frac{\partial f(u)}{\partial u}\bigg|_{u=\lambda_i(n)}, \tag{56}$$

For the internal nodes in the time plane T:

$$\delta_j(n) = \left[\left(d_j(n) - y_j(n)\right) + \sum_{i=1}^{N} \delta_i(n+1) w_{ij}^{back} \right] \frac{\partial f(u)}{\partial u}\bigg|_{u=z_j(n)}, \tag{57}$$

for the output nodes of previous time levels:

The maximum value that the current nodes can reach can be realized based on the following settings:

$$\begin{aligned} w_{ij} &= w_{ij} + \gamma \sum_{n=1}^{T} \delta_i(n) x_j(n-1), \\ w_{ij}^{in} &= w_{ij}^{in} + \gamma \sum_{n=1}^{T} \delta_i(n) u_j(n), \\ w_{ij}^{\text{out}} &= w_{ij}^{\text{out}} + \gamma \times \begin{cases} \sum_{n=1}^{T} \delta_i(n) u_j(n), \\ \sum_{n=1}^{T} \delta_i(n) x_j(n-1), \end{cases} \\ w_{ij}^{back} &= w_{ij}^{back} + \gamma \sum_{n=1}^{T} \delta_i(n) y_j(n-1). \end{aligned} \tag{58}$$

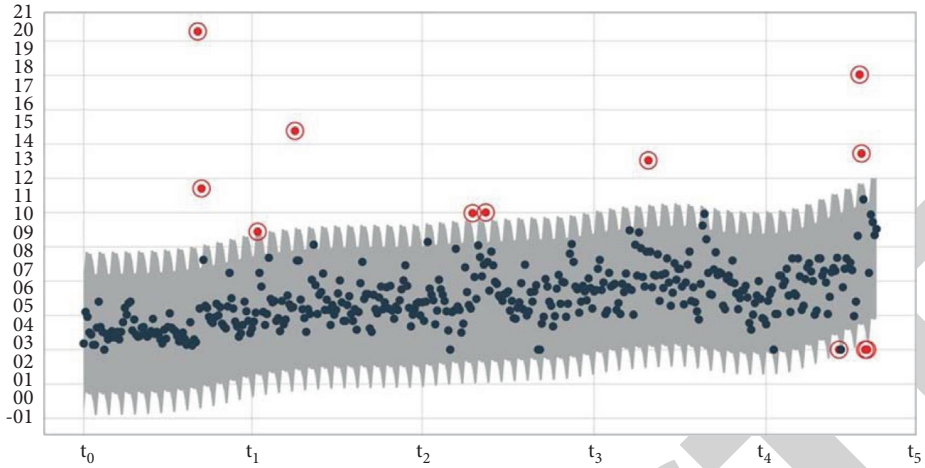


Figure 3: Anomalies plot.

Table 1: Anomalies detection performance metrics.

Classifier	Accuracy (%)	RMSE	Precision	Recall	F-score	AUC
GLSM	97.88	0.0710	0.987	0.987	0.987	0.9883
Autoencoder	95.92	0.0847	0.960	0.960	0.960	0.9798
LSTM	95.08	0.0891	0.951	0.951	0.952	0.9732
CNN	94.59	0.0928	0.946	0.946	0.947	0.9781
One class SVM	93.26	0.0933	0.933	0.934	0.933	0.9590
Isolation forest	92.51	0.0925	0.925	0.925	0.925	0.9519

Figure 3 shows the result of the process after using the proposed algorithm in modeling the problem of discovering anomalies in vibration data.

To compare the method with competitors and to confirm its superiority, the comparison Table 1 is shown below.

In conclusion, and as can be seen from the results and analyses presented above, the proposed model, considering the objective difficulties raised in this research, is a significant and powerful anomaly recognition model capable of coping with complex situations. It is essential to state that the repeated execution of these training epochs results in a complex nonlinear dynamic system that can often deviate from the desired behavior. Therefore, bifurcations are likely to form when the initialization values of the network weights are quite different from the dynamics of the system we aim to model. Near such bifurcations, gradient information can become essentially useless, dramatically reducing convergence. The fault may develop suddenly near such critical points because of the crossing boundaries of the branches. Nevertheless, modeling with spiking neural networks where idealized neurons interact with sporadic near-instantaneous discrete events in confidence intervals is guaranteed to converge to a minimum local error. This observation cannot occur with feedforward networks because they model only simple functions and not dynamical systems.

5. Conclusion

In this paper, we examine threats to the digital security of petroleum infrastructure protection services. Specifically, we presented a deep spiking neural network anomaly detection method by simulating the human brain and the way neural information is processed and transmitted. We utilized a particularly innovative form of the Galves–Löcherbach Spiking Model (GLSM), which is a spiking neural network model with intrinsic stochasticity, ideal for modeling complex spatiotemporal situations.

The proposed model is a significant and powerful anomaly recognition model capable of coping with complex situations, despite the objective difficulties raised in this research. The repeated execution of these training epochs leads in a complicated nonlinear dynamic system that frequently deviates from the planned behavior. Therefore, bifurcations are likely to occur when the initialization values of the network weights are significantly dissimilar to the dynamics of the system we are attempting to mimic. Near to such bifurcations, gradient information may become essentially meaningless, hence drastically lowering convergence. Due to the crossing borders of the branches, the fault may form suddenly close to such crucial spots. However, modeling with spiking neural networks in which idealized neurons interact with random near-instantaneous discrete events in confidence intervals is guaranteed to converge to a minimum local error.

Conventional deep learning relies on stochastic gradient descent and error backpropagation, which requires differentiable activation functions. Consequently, modifications are required to reduce activations to binary values. Integrating the timing of asynchronous operations into the training process is only performed by asynchronous spiking

neural networks. Such networks share the discontinuous nature of data but not the asynchronous operation mode of spiking neural networks. In contrast to deterministic models for spiking neural networks, a probabilistic model defines the outputs of all spiking neurons as jointly distributed binary random processes. The joint distribution is differentiable in the synaptic weights, and, as a result, so are principled learning criteria from statistics and information theory, such as likelihood function and mutual information. The change in weight distribution during the learning process is based on the weight distribution of each time interval. The maximization of such measures can apply to arbitrary topologies and does not require the implementation of backpropagation mechanisms. Hence, a stochastic viewpoint has significant analytic advantages, which translate into deriving flexible learning rules from first principles. These rules recover as exceptional cases in the theoretical neuroscience literature of the proposed model.

On the other hand, a learning rule is a local binary random process whose operation can be decomposed into atomic steps carried out in parallel at distributed processors based only on locally available information and limited communication on the connectivity graph. Local knowledge at a neuron includes the membrane potential, the feedforward filtered traces for the incoming synapses, the local feedback filtered trace, and the local model parameters. Besides local signals, learning rules may also require global feedback signals. Finally, the proposed model can use an iteration rule for a rate-coded error signal on a more extended "macro" time scale and combines this with an update on a shorter "micro" time scale which captures individual spike effects.

As an extension of the envisioned system in the future, future study should include further modification of the GLSM parameters to develop a categorization process that is even more effective and rapid [34]. For the suggested framework to fully automate the process of locating APT attacks, it must also be expanded based on methodologies for selfimprovement and parameter redefinition. In the direction of future expansion, the creation of an additional cross-sectional anomaly analysis system is a further feature that may be examined. This could act diametrically opposed to the GLSM classifier's ideology and increase the system's effectiveness.

Data Availability

The data used in the proposed approach are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] A. Galves and E. Löcherbach, "Modeling networks of spiking neurons as interacting processes with memory of variable length," 2015, http://arxiv.org/abs/1502.06446.

[2] M. Abomhara and G. M. Køien, "Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks," *Journal of Cyber Security and Mobility*, vol. 4, no. 1, pp. 65–88, 2015.

[3] A. Turnbull, J. Carroll, and A. McDonald, "Combining SCADA and vibration data into a single anomaly detection model to predict wind turbine component failure," *Wind Energy*, vol. 24, pp. 197–211, 2021.

[4] X. E. A. Novelo and H.-Y. Chu, "Application of vibration analysis using time-frequency analysis to detect and predict mechanical failure during the nut manufacturing process," *Advances in Mechanical Engineering*, vol. 14, Article ID 168781322210827, 2022.

[5] A. Galves and E. Löcherbach, "Infinite systems of interacting chains with memory of variable length - a stochastic model for biological neural nets," *Journal of Statistical Physics*, vol. 151, pp. 896–921, 2013.

[6] L. Xing, K. Demertzis, and J. Yang, "Identifying data streams anomalies by evolving spiking restricted Boltzmann machines," *Neural Computing & Applications*, vol. 32, pp. 6699–6713, 2020.

[7] Z. Lijun, H. Guiqiu, L. Qingsheng, and D. Guanhua, "An intuitionistic calculus to complex abnormal event recognition on data streams," *Security and Communication Networks*, vol. 2021, Article ID e3573753, 14 pages, 2021.

[8] K. Demertzis, N. Tziritas, P. Kikiras, S. L. Sanchez, and L. Iliadis, "The next generation cognitive security operations center: adaptive analytic lambda architecture for efficient defense against adversarial attacks," *Big Data and Cognitive Computing*, vol. 3, no. 1, p. 6, 2019.

[9] D. Bäßler, T. Kortus, and G. Gühring, "Unsupervised anomaly detection in multivariate time series with online evolving spiking neural networks," *Machine Learning*, vol. 111, no. 4, pp. 1377–1408, Apr. 2022.

[10] N. Dennler, G. Haessig, M. Cartiglia, and G. Indiveri, "Online detection of vibration anomalies using balanced spiking neural networks," 2021, http://arxiv.org/abs/2106.00687.

[11] K. Demertzis, L. Iliadis, and S. Spartalis, "A spiking one-class Anomaly detection framework for cyber-security on industrial control systems," in *in engineering Applications of neural networks*, G. Boracchi, L. Iliadis, C. Jayne, and A. Likas, Eds., vol. 744, pp. 122–134, springer international publishing, Cham, 2017.

[12] K. Demertzis and L. Iliadis, "Detecting inva sive species with a bio-inspired semi-supervised neurocomputing approach: the case of Lagocephalus sceleratus," *Neural Computing & Applications*, vol. 28, pp. 1225–1234, 2017.

[13] P. Stratton, A. Wabnitz, and T. J. Hamilton, "A Spiking Neural Network Based Auto-Encoder for Anomaly Detection in Streaming Data," in *Proceedings of the in 2020 IEEE Symposium Series On Computational Intelligence (SSCI)*, pp. 1981–1988, Canberra, Australia, December 2020.

[14] P. S. Maciąg, M. Kryszkiewicz, R. Bembenik, J. Lobo, and J. Del Ser, "Unsupervised anomaly detection in stream data with online evolving spiking neural networks," *Neural Networks*, vol. 139, pp. 118–139, 2021.

[15] A. Khadersab and S. Shivakumar, "Vibration analysis techniques for rotating machinery and its effect on bearing faults," *Procedia Manufacturing*, vol. 20, pp. 247–252, 2018.

[16] M. Conti, D. Donadel, and F. Turrin, "A survey on industrial control system testbeds and datasets for security research," *IEEE Communications Surveys & Tutorials*, vol. 23, pp. 2248–2294, Article ID 3094360, 2021.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4611331, 20 pages
https://doi.org/10.1155/2022/4611331

Research Article

GLD-Net: Deep Learning to Detect DDoS Attack via Topological and Traffic Feature Fusion

Wei Guo, Han Qiu, Zimian Liu, Junhu Zhu, and Qingxian Wang

State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450002, China

Correspondence should be addressed to Han Qiu; qiuhan_loach@sina.com

Received 28 May 2022; Revised 8 July 2022; Accepted 27 July 2022; Published 16 August 2022

Academic Editor: Konstantinos Demertzis

Distributed denial of service (DDoS) attacks are the most common means of cyberattacks against infrastructure, and detection is the first step in combating them. The current DDoS detection mainly uses the improvement or fusion of machine learning and deep learning methods to improve classification performance. However, most classifiers are trained with statistical flow features as input, ignoring topological connection changes. This one-sidedness affects the detection accuracy and cannot provide a basis for the distribution of attack sources for defense deployment. In this study, we propose a topological and flow feature-based deep learning method (GLD-Net), which simultaneously extracts flow and topological features from time-series flow data and exploits graph attention network (GAT) to mine correlations between non-Euclidean features to fuse flow and topological features. The long short-term memory (LSTM) network connected behind GAT obtains the node neighborhood relationship, and the fully connected layer is utilized to achieve feature dimension reduction and traffic type mapping. Experiments on the NSL-KDD2009 and CIC-IDS2017 datasets show that the detection accuracy of the GLD-Net method for two classifications (normal and DDoS flow) and three classifications (normal, fast DDoS flow, and slow DDoS flow) reaches 0.993 and 0.942, respectively. Compared with the existing DDoS attack detection methods, its average improvement is 0.11 and 0.081, respectively. In addition, the correlation coefficient between the detection accuracy of attack flow and the four source distribution indicators ranges from 0.7 to 0.83, which lays a foundation for the inference of attack source distribution. Notably, we are the first to fuse topology and flow features and achieve high-performance DDoS attack intrusion detection through graph-style neural networks. This study has important implications for related research and development of network security systems in other fields.

1. Introduction

Popular industries such as shopping, education, finance, government affairs disclosure, and communications connect core services, such as payments, instant messaging, and big data analysis, to the Internet in real time for user access. Due to these services' vulnerability and high value, attacks on infrastructures that provide these services are favored by hackers. One of the most common attacks to block these services is the DDoS attack [1]. How to deal with DDoS attacks to ensure network smoothness has become a research hotspot [2].

The traditional defense strategy assumes that the attack topology is a static point-to-point model whose topology remains unchanged during the attack [3, 4]. Under this premise, DDoS detection is mainly realized through changes in traffic size, and the corresponding single-point defense is relatively simple [5]. In 2022, Israel's network providers were hit by a large DDoS attack from abroad, paralyzing the website of the Interior Ministry for hours. The same year, the Ukrainian government suffered repeated DDoS attacks from Russia and Belarus, forcing multiple portals to shut down to avoid losses. The network situation has changed as DDoS attacks shift from individual behaviors to confrontations between countries. Except for the increase in the attack traffic, the range of attack sources continues to expand; the flow topology evolves during the attack. Conventional single-point defense cannot cope with these changes, and multisource protection requires attack source location. However, previous detections cannot identify the attack

distribution, thus impossible to precise defense. Therefore, we need a detection method that can determine the attack distribution through topology changes to support further attack tracing and defense deployment.

DDoS detection aims to distinguish attack traffic from legitimate traffic. According to the different fields of mathematics, the current mainstream DDoS detection methods can be divided into three categories: statistics, machine learning, and deep learning. The statistical method uses measures such as entropy to evaluate the traffic distribution's change. It is simple and requires no additional hardware support. However, its detection effect depends on thresholds, which researchers usually give directly [6]. This subjective assignment lacks an objective basis, affecting the reliability of results. Machine learning classifies network traffic through classifiers designed based on selected features. Due to modeling using features, it exhibits excellent flexibility. However, reliance on feature engineering makes it less adaptable in the face of complex real network traffic. In addition, traditional machine learning belongs to shallow learning, making it difficult to learn deep relationships. Thus, its accuracy is usually less than 90%. Deep learning utilizes multilayer neural networks to learn the inherent laws of network traffic. The feature extraction is contained in the neural network structure without additional processing. Besides, multilayer neural networks can mine deep information, making up for the defects of shallow learning. DDoS detection based on deep learning has high accuracy and efficiency. For different requirements and problems, many related research studies are emerging [7]. These studies usually improve performance by improving or fusing network architectures. For example, convolutional neural network (CNN) and recurrent neural network (RNN) are used to process the relationship between features in time and space [8], RNN and automatic codec are combined to improve the detection sensitivity on SDN [9], and adaptive transfer learning is introduced to achieve small sample detection [10]. These methods achieve good performance by exploiting the efficient information in the input as much as possible through elaborate devised architecture and parameters.

DDoS attacks have two notable characteristics: (1) from a spatial perspective, heavy traffic in the short term changes the distribution of adjacent upstream nodes of the victim host [11]. (2) From a time perspective, the prolonged blocking makes the limited attack nodes have multiple attack behaviors on the target [12]. These two intrinsic peculiarities of DDoS attacks make the network topology before and after the attack significantly different. Therefore, in addition to traffic characteristics, DDoS attacks can also be detected based on the topology changes [13]. This difference is implicit in the evolution of the topology structure, which traffic statistics cannot depict. Introducing topology changes can improve detection accuracy and help analyze the distribution of attack sources. The data (such as graph) considering topological connection are non-Euclidean data. Sample points (nodes) have different numbers of neighbor points, and edges depict their interdependence. However, traditional deep learning requires Euclidean data as input to extract features. For example, CNN needs the sample to be regular and independent. RNN demands the data to be a one-dimensional real vector. The linear input cannot deal with topological relationships. Graph attention network (GAT) is a powerful analysis tool for graph data [14]. It incorporates the attention mechanism into the graph neural network and captures associations through neighborhoods. Further, the attention mechanism assigns different weights to adjacent nodes, improving feature sensitivity. In this study, we treat topology as graph data. In particular, edge attributes denote traffic features, and node attributes indicate topological features. Therefore, GAT can simultaneously analyze traffic and topological features with the graph as input. To our knowledge, we are the first to achieve DDoS attack detection using graph-style deep learning. The main contributions of this study are as follows:

(1) The proposed dynamic topology construction algorithm integrates topology and flow features into node or edge attributes.

(2) GAT is used to mine topology change patterns and train classifiers.

(3) Compared with other methods, the deep learning method integrating topology and traffic features achieves higher accuracy in both two classifications and three classifications of DDoS traffic.

(4) The proposed detection method supports estimating the distribution of attack traffic sources.

The rest of this study is organized as follows. In Section 2, we discuss research related to DDoS detection. Section 3 describes the details of the proposed method in terms of feature extraction and deep learning architecture. Section 4 designs the experiments and analyzes the results. We summarize this research in Section 5. Finally, the shortcomings and future research are pointed out in Section 6.

2. Related Work

In recent years, DDoS detection research mainly acquires traffic features containing attack-specific information through feature acquisition [15]. Then, characteristics are analyzed based on different theories or tools to discover traffic classification patterns [16]. Section 2 describes current DDoS feature acquisition methods and summarizes three mainstream DDoS detection methods: statistics, machine learning, and deep learning.

2.1. Feature Acquisition. Valid feature input is critical to traffic classification performance since it determines the valuable information contained in samples. There are two main methods for feature acquisition: output features using generator tools (such as CICFlowMeter) [17, 18] and custom features based on subjective experience [19, 20]. The former applies public datasets or traffic extraction tools to obtain features, while the latter designs corresponding features according to application requirements. In 2017, Yuan et al. [17] extracted 20 network traffic fields from the ISCX2012 dataset for *DeepDefense* detection model training. This

method is simple and avoids complex statistical feature calculations. In 2018, Idhammad et al. [18] reduced the feature dimension of datasets such as UNSW-NB15 based on collaborative clustering. Then, simplified datasets were used to test machine learning methods' classification performance. The results show that this method effectively reduces the false-positive rate. In 2018, Doshi et al. [19] extracted three stateless and two stateful features through network packet behavior, which showed high accuracy in IoT traffic detection. In 2019, De Lima Filho et al. [20] utilized 25 IPv4 variables to design 33 signature features suitable for IP, UDP, and TCP, which improved the sensitivity of online DDoS detection. In 2022, Chouhan et al. [21] defined the seven most relevant features for real-time traffic detection. They extracted these features from switch statistics based on the Ryu controller module, reducing the identification delay of the classifier.

The above methods propose efficient feature acquisition strategies. Nevertheless, these flow feature extraction ways lack the characterization of the topology. Therefore, it is needed to define topological features and give corresponding acquisition methods.

2.2. Statistical Method. Statistical methods use numerical distribution to differentiate traffic. In 2017, Hoque et al. [22] proposed a new correlation indicator NaHiD based on standard deviation and mean. Experimental results show that this measure is more robust and sensitive to state changes than traditional metrics. In 2022, Tsobdjou et al. [23] raised a dynamic entropy threshold algorithm based on Chebyshev inequality. Comparative experiments indicate that this method can better adapt to varied online environments than static thresholds. The same year, Ahalawat et al. [24] proposed a Renyi entropy DDoS attack detection technique based on the packet drop strategy. It can evaluate the probability distribution of flow fluctuations and achieve better results than the Shannon entropy.

These methods analyze the numerical fluctuation of flow from a macro-view. However, their application scope is narrow due to the lack of fine-grained characterization. Thus, statistical methods are usually not used alone for comprehensive evaluation.

2.3. Machine Learning. Machine learning can automatically learn feature patterns and create classifiers. In 2019, Gu et al. [25] proposed the DDoS detection algorithm SKM-HFS. Weighted K-means analysis balances the number of samples and accuracy, and the density clustering center algorithm optimizes the extreme values. The results show that this method performs best when choosing TOPSIS as the evaluation factor. In 2020, Pande et al. [26] utilized the random forest algorithm to distinguish between normal and attack samples and used the WEKA tool to detect DDoS attack ping of death. Experiments on NSL-KDD indicate that random forest achieves the highest accuracy of 99.76% on specific attacks. In 2021, Cvitic et al. [27] understand DDoS detection as a multi-device classification problem and distinguish traffic generated by different IoT devices through a logical model tree. A comparison of four typical devices shows that the logical model tree can better identify DDoS traffic from IoT devices. In 2022, Kumar et al. [28] designed the recursive feature elimination method RFE. It is also combined with the random forest algorithm to train the classifier. Experiments show that this method can cope with fast detection under large network traffic.

The above methods extract relevant information from the traffic details. However, they rely heavily on feature engineering and have low performance in the face of large samples. Hence, we need to find a more efficient detection model.

2.4. Deep Learning. Deep learning applies a multilayer neural network to obtain the correlation between input and output. In 2019, Liang and Znati [29] employed LSTM in a DDoS detection framework. LSTM captures the implicit sequence representation in the input vector through three gating units. This method can learn flow-level modes, avoiding expensive and error-prone feature engineering. In 2020, Doriguzzi-Corin et al. [30] proposed LUCID, a lightweight DDoS detection system that utilizes one-dimensional CNN to reduce computational load. Experiments on ISCX2021, CIC-IDS2017, and CSE-CIC2018 datasets show that LUCID has a 40x reduction in processing time compared with other deep learning methods, so it is suitable for detection under limited resources. In 2021, Cil et al. [31] built a traffic classification model based on the deep neural network. Its structure contains feature extraction, and training can be completed with only three fully connected layers. Experiments on CIC-DDoS2019 show that the model has an accuracy of 95%. In 2022, Boonchai et al. [32] implemented two DDoS detection models using the DNN architecture and autoencoder, respectively, and verified the attack recognition ability of the models through the CIC-DDoS2019 dataset with an accuracy rate of 87% and 91.9%, respectively.

A single method is challenging to meet diverse DDoS detection needs. Therefore, many scholars extend the applicability through method mixing. In 2019, Pektaş and Acarman [8] extracted five statistical features: duration, bytes, packets, periodicity, and states through network traffic summary and mined semantic information in the feature sequence through CNN and RNN. The accuracy of this method reaches 99.1%, significantly higher than a single network. In 2020, Wang and Liu [33] employed information entropy and deep learning to detect DDoS attacks in SDN. First, IP entropy identifies malicious traffic routers, and then, CNN classifies packet-level traffic. This method achieves 98.98% accuracy and also reduces training time. In 2020, Elsayed et al. [9] proposed DDoSNet, an intrusion detection system for SDN. This system combines RNN and autoencoder. RNNs capture sequence relationships, and autoencoders detect small perturbations. Compared with baseline methods such as decision tree, random forest, and support vector machine, DDoSNet is more stable and achieves an accuracy of 99%. In 2021, Shieh et al. [34] built a DDoS unknown traffic discovery model BI-LSTM-GMM. It consists of bidirectional LSTM (BI-LSTM) and Gaussian mixture model (GMM). GMM labels the unknown traffic

and adds it to the new input of BI-LSTM. Experiments show that this method can identify unknown attacks through reinforcement learning. In 2022, Almaraz-Rivera et al. [35] designed a new intrusion detection system based on machine learning and deep learning models to solve the unbalanced detection of DDoS attack categories. It combines decision trees and multilayer perceptrons to test binary classification performance on different datasets, avoiding data and fragmentation effects.

Besides binary classification, multi-classification studies that ease defense deployment are also emerging. In 2019, Toupas et al. [36] utilized stacked fully connected layers for intrusion detection. Experiments show that this method can better learn the difference between fast and slow DDoS flows with an accuracy of 95.62%. In 2020, Alzahrani et al. [37] proposed FastGRNN, a DDoS multi-classification method for IoT, which reduces training complexity by adding residual to hidden states. It achieves 1:5 optimization of detection time and training time to adapt to real-time detection. In 2020, Hussain et al. [38] used ResNet for complex traffic detection. They convert traffic into a three-channel format and analyze it through ResNet. This method achieves an accuracy of 87% in distinguishing normal flow, fast DDoS flow, and slow DDoS flow and an increase of 9% compared with other methods. In 2022, Rusyaidi et al. [39] designed a high-precision DDoS attack detection system based on DNN and LSTM. It achieved an accuracy of 97.37% on the NSL-KDD dataset and excellent performance in identifying 22 traffic types.

With the in-depth development of deep learning, many researchers also apply new architectures to optimize DDoS detection performance. In 2020, He et al. [10] employed transfer learning for small-sample DDoS detection. They also define the transfer ability to evaluate different networks and select the best network structure and parameters. This method improves the detection accuracy on small samples by 20.8%, which can effectively cope with training degradation. In 2021, Novase et al. [40] utilized generative adversarial network (GAN) to detect DDoS adversarial attacks. It improves system robustness through adversarial training and uses IP entropy to analyze continuous traffic for real-time monitoring. This method shows strong adaptability in detecting adversarial attacks. In 2022, Doriguzzi-Corin and Siracusa [41] proposed an adaptive mechanism for DDoS attack detection based on federated learning, FLAD. It updated federated learning to solve the integration problem in dynamic security confrontation, monitoring the status locally without interaction. The experimental results verified the efficiency and performance of the method.

Deep learning has shown advantages in different detection requirements. However, it can only process traffic characteristics and not extract topology information. Thus, we need to find a new way to consider both features to improve detection accuracy and lay a basis for attack source localization.

3. Methodology

This section details the procedure and implementation of deep learning detection based on topological and flow features. As shown in Figure 1, our proposed DDoS detection system has three main parts. The first part is the extraction module. It is responsible for extracting features from public datasets or actual scenes and transforming samples into graph data consisting of nodes and edges. The second part is the training module, which builds a classification model that can mine deep-level information from samples. The input is sample data, the output is label type, and parameters are optimized during training. The third part is the evaluation module, which compares detection effects under different hyperparameters to select the optimal configuration.

When the above stages are completed, the pattern analysis for feature extraction is saved as an extractor, and the trained neural network is preserved as a classifier. Then, real traffic can be quickly classified by running through these processing parts only once without retraining.

3.1. Feature Extraction. Building a topology diagram is the core of topology feature extraction. It maps from raw traffic data to dynamic topology plots that evolve; an example is shown in Figure 2. In particular, f_i denotes the flow's distribution originating from the corresponding ip_n, G_m denotes the subgraph under time slice t_m, and the indicators in the feature table denote extracted samples. Besides, F, Tab, G, and T represent the set of flow distribution $\{f_1, f_2, \cdots, f_s\}$, feature table $[tab_1, tab_2, \cdots, tab_m]$, topology map $G_1 \cup G_2 \cup \cdots \cup G_m$, and time slice $T_1 + T_2 + \cdots + T_m$, respectively. In Figure 2, there are two major stages. The first stage realizes the transformation from traffic data to node or edge features. The traffic records of different source IPs are divided according to the time unit. Then, the time slice proportions of features are formed into [IP-Feature] pairs and saved in the feature table. The second stage builds the connection graph, adds attributes to nodes or edges according to the feature table, and decides whether to add based on the flow that exists or not in the topology. The final topological feature map $G = G_1 \cup G_2 \cup \cdots \cup G_m$ on m sub-time segments is obtained when the addition is finished.

In Figure 2, the feature table determines the attributes of edges and nodes. Besides statistical features, we also add connection state, packet marker, and centrality features. The attack pattern implied in the connection state sequence can distinguish attack phases [8]. For example, the target maintains many half-open connections in a SYN flood attack, making the state list of long LISTENs. The normal communication state is composed of LISTEN, ESTABLISHED, and CLOSED. In this case, the proportion of LISTEN differs from that of SYN attack. Therefore, state sequences reflecting this divergence can be used for traffic classification. Packet flags reveal the attacker's malicious attack intention. For example, regular data packets must be queued in the buffer before parsing, while numerous URG flags set to 1 increase the processing priority, thus enabling fast attacks. Hence, the packet tampering details that macroscopic features cannot describe are hidden in the packet marking sequence, thus detecting the attack. We choose the degree and betweenness centrality based on the understanding that destructive attackers usually control

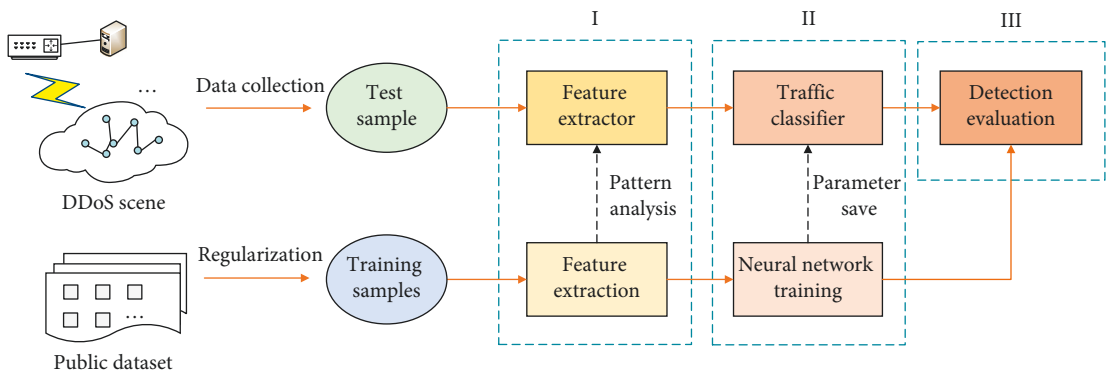


Figure 1: Framework of DDoS attack detection system based on deep learning.

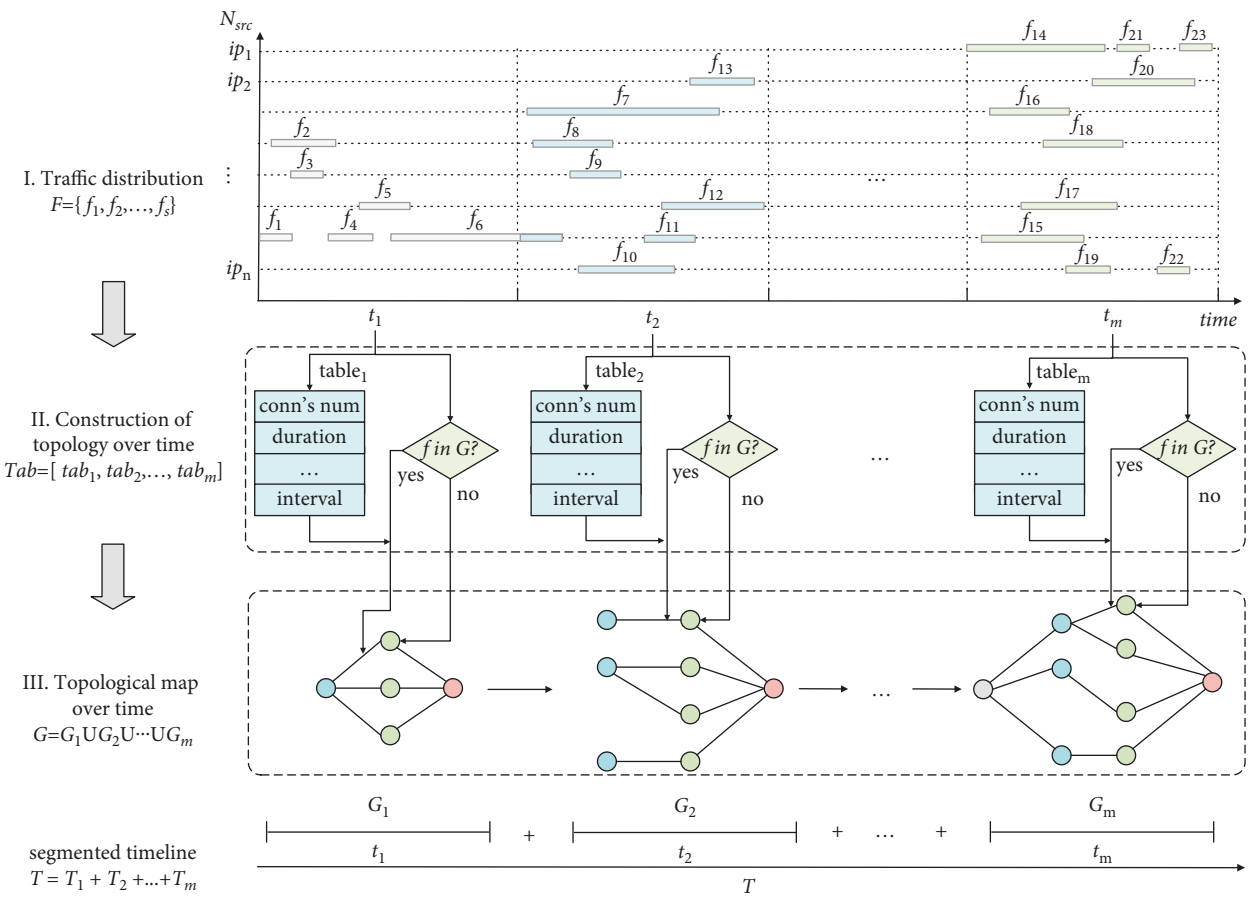


Figure 2: An example of topological feature extraction and topological map construction process based on time series.

more agents to execute attacks, and targets are generally critical nodes [42]. Then, these two centralities can capture the attack preference to realize attack detection. In conclusion, we extract multiple node or edge features for attack detection, addressing the one-sidedness of the training data.

We extract features from ICMP, UDP, and TCP, respectively. The features of protocols except TCP are the same. For clarity, we take TCP features as an example to illustrate the attributes of the extracted features, as shown in Table 1. Among them, t_s represents an arbitrary time field.

Table 1 includes eight edge and two node features. In particular, the edge feature depicts the traffic distribution through the edge in period t_s; the node feature describes the spatial distribution of adjacent nodes within t_s. URG and ECE flags are extended to lists to preserve the time-varying properties. We also introduce degree and betweenness centrality to characterize topology changes. The meaning and acquisition of the features in Table 1 are described below.

Among edge features, except "connection number," "connection states," "URG flag," and "ECE flag," the other four can be calculated by statistical formula. Notably, the

Table 1: Node or edge TCP features collected from time series.

Item Object	Name	Type	Number	Description
Edge	Connection number	Integer	1	Total number of traffic records in t_s
	Connection states	List	States' number	Number of communication states in t_s
	Duration	Float	6	Total, mean, median, standard deviation, maximum, and minimum of flow duration
	Packet interval	Float	6	Total, mean, median, standard deviation, maximum, and minimum of the packets' interval
	Forward packets' number	Float	6	Total, mean, median, standard deviation, maximum, and minimum of the forward packets' number
	Backward packets' number	Float	6	Total, mean, median, standard deviation, maximum, and minimum of the backward packets' number
	URG flag	List	Packets' number	Sequences of URG flags in t_s
	ECE flag	List	Packets' number	Sequences of ECE flags in t_s
Node	Degree centrality	Float	1	Number of neighbor nodes connected to the node within t_s
	Betweenness centrality	Float	1	Number of shortest paths passing through the node within t_s

standard deviation measures the discrete distribution of samples. When the number of samples with the same IP and protocol in t_s is 3, regular flows are much more than attack flows, making the training challenging to converge. So, when the number of samples is not less than 3, the standard deviation has practical significance. In addition, in subsequent experiments, we found that the detection efficiency and accuracy are balanced when the number of samples is not less than 4. Therefore, we only consider four or more identical protocol connections established between the same node pair as actual training data.

Connection number and connection state are two macro-edge features. The former reflects the frequency of establishing connections between nodes; the latter reflects the continuous change in the protocol state. For example, CLOSED means all active links are closed; LISTEN signifies waiting for new requests; and ESTABLISHED means the connection is established. These states can be obtained through traffic analysis tools like TShark or CICFlowMeter [43]. However, the acquired elements are of type string and hard to use for training directly. We use one hot to convert state sequences into real vectors to simplify computation [44]. In particular, zero indicates that the state is not enabled, and one denotes that the state is activated.

"URG flag" and "ECE flag" record the state sequence of consecutive packets. These two flags represent unexpected events during the sending of traffic. In particular, a URG of 1 indicates that the current data packet is prioritized and should be processed without queuing; an ECE of 1 indicates congestion, and the sending window decreases. The abnormal state to one is set, and the normal state is set to zero; then, the state sequence is a list of zeros and ones. This binary list can be used for training directly without encoding.

Two node features, degree and betweenness centrality, characterize the connectivity properties of the neighborhood. Degree centrality uses the number of adjacent nodes to denote the node importance. Let *Len* be a function of solving the number of non-repetitive elements; $N_{neighbor}$ represents the number of neighboring nodes of node N; $\langle N^s_{src}, N^s_{des}\rangle$ indicate flow s. Then, degree centrality is formulated as follows:

$$N_{degree} = \frac{N_{neighbor}}{Len\left(\left(N^1_{src}, \cdots, N^s_{src}\right) \cup \left(N^1_{des}, \cdots, N^s_{des}\right)\right) - 1}. \quad (1)$$

Betweenness centrality measures the node importance by the ratio of shortest paths' number through a node. Let $Path(N)$ denote the number of shortest paths containing node N, and $Path(src_i, dst_j)$ denote the number of shortest paths between src_i and dst_j. Then, the formula of betweenness centrality is as follows:

$$N_{between} = \frac{Path(N)}{\sum_{i,j} Path\left(src_i, dst_j\right)}. \quad (2)$$

We also show the construction process of the DDoS topology map through pseudo-code, as shown in Algorithm 1.

Algorithm 1 can be divided into three main stages, namely, the initial assignment stage (1~2), configuration stage (3~11), and mapping stage (12~21). The graph structure is initialized in the initial assignment phase, and necessary parameters are set. Statistical features are calculated based on item durations' total and average in the configuration stage. In the final phase, nodes and directed edges are added to graph G, and features are attached to them. After the above steps are completed, the topology graph G is constructed.

To sum up, by extracting the features in Table 1 and constructing a dynamic topology graph according to Algorithm 1, the flow or topology attributes are included in the node or edge features. Till now, we have obtained structured data for training.

3.2. Architecture of Deep Learning Model. This part introduces GLD-Net, a deep learning model capable of analyzing and fusing topology and flow features. Its structure is shown in Figure 3. This model has three main parts: the GAT layer, LSTM, and the fully connected layer (also known as the dense layer). Firstly, an L-layer GAT network is used to analyze the topological data. Its output is a spatial sequence over the neighborhood. Secondly, sequence relationships in the output are mined by LSTM. Finally, the dense layer reduces the feature dimension, and the softmax function limits the output size between zero and one. This value corresponds to the traffic label to achieve classification. In the following subsections, we will detail the processing method of each neural network and the information transfer within it. In particular, L and K in Figure 3 represent the number of attention mechanisms and the number of splice heads in multi-head attention, respectively. The detailed parameter functions and setting methods of the GAT, LSTM, and dense layer will be explained in each subsection.

3.2.1. GAT Layer. The two basic units of GAT are attention coefficient calculation and information aggregation, as shown in the dotted box in Figure 3. The structure of the attention coefficient calculation is shown in Figure 4.

In Figure 4, we take four adjacent nodes $N_j = (n_j^1, n_j^2, n_j^3, n_j^4)$ of node n_i as an example to illustrate the information transfer progress in calculating the attention coefficient. Let $U_{ij} = (u_{ij}^1, u_{ij}^2, u_{ij}^3, u_{ij}^4)$, ur/ij be the intermediate variables obtained by splicing the initial node n_i and the adjacent node n_j^r after feature enhancement φ, and Concat denote concatenation operation; that is,

$$u_{ij}^r = \mathrm{Concat}\left(\varphi\left(n_i\right), \varphi\left(n_j^r\right)\right), \quad r = [1, 2, 3, 4]. \tag{3}$$

Let w be a trainable shared weight, and $\varphi(n_i)$ and $\varphi(n_j^r)$ can be obtained by linear transformations, which are, respectively, expressed as follows:

$$\varphi\left(n_i\right) = w \cdot n_i, \quad \varphi\left(n_j^r\right) = w \cdot n_j^r. \tag{4}$$

A similarity coefficient e_{ij}^r can be obtained by the inner product of the intermediate variable u_{ij}^r and the trainable parameter vector $\overrightarrow{s}$. In addition, the deviation of similarity coefficients is corrected by LeakyReLU. The negative axis slope of LeakyReLU retains negative values so that similarity coefficients do not suffer from the loss of negative information like ReLU. The formula of e_{ij}^r is expressed as follows:

$$e_{ij}^r = \mathrm{LeakyReLU}\left(\langle \overrightarrow{s}^T, u_{ij}^r \rangle\right). \tag{5}$$

e_{ij}^r needs to be normalized on the interval [0, 1] to facilitate information aggregation. According to different transformation modes, normalization can be divided into linear methods, such as min-max [45] and Z-score [46], and nonlinear methods, such as softmax [47]. In particular, min-max only needs extremum and current values. Its calculation is simple but easily affected by individual points. Z-score utilizes comprehensive information and is less affected by outliers. However, the data must meet the normal distribution; otherwise, the output will be seriously distorted. The exponential calculation of softmax is a smooth derivative transformation that retains each value's influence and has no data distribution requirements. We choose the nonlinear function softmax as the normalization method to prevent the loss of complex information in the transformation process. Then, the attention coefficient σ_{ij}^r can be calculated by the following formula:

$$\sigma_{ij}^r = \frac{\exp\left(e_{ij}^r\right)}{\sum_{k=1}^4 \exp\left(e_{ij}^k\right)}. \tag{6}$$

The attention coefficient σ_{ij}^r contains the correlation between node n_i and neighbor node n_j^r. Then, the information aggregation based on neighborhood nodes can be realized with the attention coefficient. Its structure is shown in Figure 5.

In Figure 5, the new node feature n_i' can be calculated by the weighted sum of all neighbor node features $\varphi(n_j^r)$ with σ_{ij}^r. The calculation formula is as follows:

$$n_i' = \tau\left(\sum_{r=1}^4 \sigma_{ij}^r \cdot \varphi\left(n_j^r\right)\right), \tag{7}$$

where τ is the transformation function that maps the original vector space $\mathbb{R}^r$ to a new vector space $\mathbb{R}^{r'}$ centered at n_i'. Due to errors, there may be offsets in a single calculation. Therefore, we use multi-head attention with parameter K to improve the robustness of the results. Commonly used methods of combining multiple attention include concatenation and averaging [14]. We choose arithmetic averaging as the synthesis algorithm for reduced dimensionality and higher efficiency. Then, the following formula is obtained:

$$n_i'(K) = \frac{1}{K}\left(\tau \sum_{h=1}^K n_i'(h)\right). \tag{8}$$

Except for the learnable parameters w and $\overrightarrow{s}$, other parameters, including the number of attention mechanisms L, the number of multiple heads K, and the gradient θ of LeakyReLU, are hyperparameters configured before training. Grassia et al. [48] pointed out that the size of L is related to the ability of information aggregation, and a single attention mechanism can learn node features up to 3 hops away. Thus, the bigger L is, the wider the range of information aggregation is. In datasets such as NSL-KDD2009, the number of IP hops of data packets does not exceed seven jumps [49], so L set to three can meet the requirements. K determines the learning perspective of relevant information. Efficiency and accuracy are balanced when K is set to 20 in subsequent experiments. θ affects the weight update rate, and its value should be adapted to the dataset size and learning depth. Combined with the tradeoff theory [50], θ is set to 0.3.

To sum up, we achieve local information aggregation of node features through GAT's feature transformation and multi-head attention mechanism. This study considers edges and nodes as entities of the same status. Edge features are merged into node features for unified processing to simplify computation.

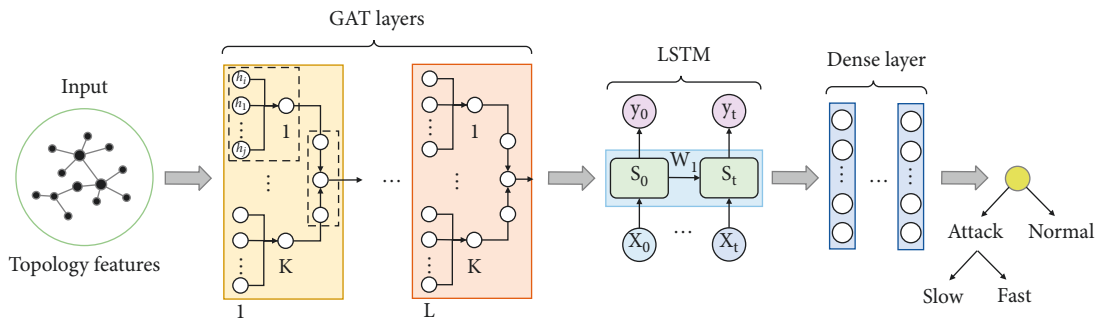


Figure 3: Overall architecture of the deep learning model GLD-Net.

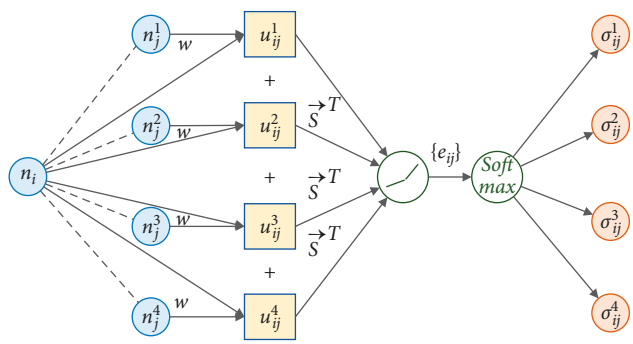


Figure 4: Information transfer process of the attention coefficient calculation in GAT.

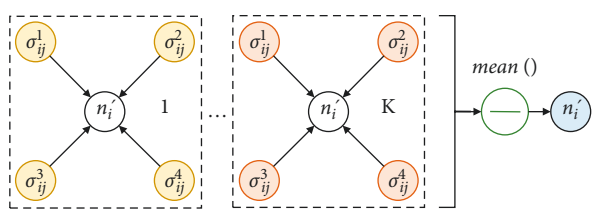


Figure 5: Flow mechanism of the information aggregation based on attention coefficients in GAT.

3.2.2. LSTM. After GAT training, the output vector $n' = \{n_i', i \in no\ de\}$ is obtained. This vector contains the spatial sequence information of nodes in the neighborhood. Suppose it is directly poured into the classifier without processing. In that case, it will cause the loss of semantic information. Common network structures for processing sequence data are RNN and LSTM [51]. RNN uses memory units to retain historical data. Thus, the output is determined by the previous data and current input. However, due to the disappearance of the back propagation gradient, it is easy to cause short-term memory. LSTM enhances memory with gating units to learn relevant information in longer sequences. There are many adjacent attack nodes in DDoS attacks, making the distance between related data in the original data larger. Therefore, we choose LSTM to mine long sequence information. For clarification, we use the input x_t of LSTM at time t as an example to illustrate the information flow. In Figure 3, the memory cell S_t at time t comprises three gating units: input gate $I(t)$, forgetting gate $F(t)$, and output gate $O(t)$. Assuming that the output of the memory cell at time t-1 is c_{t-1}, then the calculation formulas of $I(t)$, $F(t)$, and $O(t)$ at the next moment t are as follows:

$$\begin{aligned} I(t) &= sigmoid\left(W_i^t \cdot concat\left(c_{t-1}, x_t\right) + b_i^t\right) \\ F(t) &= sigmoid\left(W_f^t \cdot concat\left(c_{t-1}, x_t\right) + b_f^t\right), \\ O(t) &= sigmoid\left(W_o^t \cdot concat\left(c_{t-1}, x_t\right) + b_o^t\right) \end{aligned} \tag{9}$$

where W^t and b^t represent the weight and bias of transformation at time t, respectively. The sigmoid function controls the values of $I(t)$, $F(t)$, and $O(t)$ to fall within the interval [0,1]. In particular, one means all the information flow is passed, and zero means the information flow is blocked. Let $\widetilde{B}(t)$ be the data to be processed, and its calculation formula is as follows:

$$\widetilde{B}(t) = \tanh\left(W_e^t \cdot concat\left(c_{t-1}, x_t\right) + b_e^t\right), \tag{10}$$

where W_e^t and b_e^t are the parameters of memory cell state transition. Assume the intermediate state of the memory cell at time t-1 is B_{t-1}. The recording of $\widetilde{B}(t)$ and the forgetting of B_{t-1} are controlled by $I(t)$ and $F(t)$, respectively. Let $\otimes$ denote the defined gating transformation; then, at the next moment, the updated intermediate state B_t can be expressed as follows:

$$B_t = F(t) \otimes B_{t-1} + I(t) \otimes \widetilde{B}(t). \tag{11}$$

The output gate $O(t)$ controls the actual information passing through the intermediate state B_t. Then, the formula to obtain the final output c_t is as follows:

$$c_t = O(t) \otimes \tanh\left(B_t\right). \tag{12}$$

In addition to the learnable weight W and bias b, the calculation of c_t has three key parameters: the input vector dimension, the state dimension of the intermediate layer, and the number of memory cell layers. In particular, the input vector dimension is consistent with the received data n'. The middle layer's state dimension determines memory cells' learning ability. It is set to 32 to cover as many patterns as possible. The internal recursive structure of LSTM makes nonparallel operations more complicated. Noticeably, the excellent extraction ability of LSTM makes it unnecessary to stack too many layers in practical applications. For example, Google Translate only requires no more than eight layers to complete the vast majority of bidirectional translation tasks

[52]. In this study, when the number of memory cell layers is set to 3, the correlation extraction of DDoS data can be satisfied.

3.2.3. Dense Layer. The previous chapter realized the fusion of sequence information. Then, in this part, the final evaluation value will be obtained based on information aggregation. In Figure 3, LSTM is followed by fully connected layers, constituting a classifier with the dropout layer and activation function. GAT and LSTM map the DDoS raw sample to the feature space. Then, the fully connected layer maps the learned feature representation to the DDoS label space. The dense layers are set to 3 to learn nonlinear correlation [53]. The number of neurons in each layer is 128, 64, and 32, considering the running efficiency and learning ability. We also add dropout layers after the first and second layers to avoid overfitting. In testing, the removal probability was set to 0.3 to improve the model's generalization ability. Distinguishing the attack type (slow or fast) is a multi-classification problem. Softmax is selected to assign probabilities between 0 and 1 for different input samples. The formula is as follows:

$$P_i = \text{softmax}(c_i) = \frac{e^{c_i}}{\sum_i e^{c_i}}. \tag{13}$$

This model belongs to supervised learning. Labels allow the model to use the feedback value of the cross-entropy loss function to correct errors during the back propagation. The weights and biases are updated layer by layer to approximate the expectation. Training ends when all iterations are over. Training is done multiple times, and the best performing parameters are saved for fast classification.

4. Experiment

In this section, we elaborate on the implementation and evaluation details of the proposed method. First, running environments are illustrated to enhance the reproducibility of results. Secondly, training datasets are selected, and data preprocessing is given. Then, measures including accuracy, recall, precision, and F1-score are used to evaluate the effectiveness of detection methods. Finally, compared with baselines and state-of-the-art techniques, the performance of the proposed method is verified, and its efficiency is examined. Further, the correlation between the detection value and the source distribution is also analyzed.

4.1. Running Environment. The experiments were run on a Windows 10 workstation with Intel Core i7-12700H 4.7 GHz processor, 32 GB RAM, 512 GB SSD, and NVIDIA RTX 3060 graphics card. The GLD-Net model uses Python 3.5 as the programming language and adopts Keras as the deep learning framework to improve portability. Keras provides structured modules and connects to the GPU for acceleration via the backend engine TensorFlow's cuDNN library. Additionally, libraries such as Pickle, NumPy, and SciPy are loaded to enhance the efficiency of algorithms.

4.2. Datasets. Commonly used cybersecurity datasets include NSL-KDD2009 [54], CIC-IDS2017 [55], CIC-IDS2018 [56], and CIC-DDoS2019 [57]. In particular, CIC-IDS2018 and CIC-DDoS2019 simulate DDoS attacks through point-to-point transmission and lack topology characterization, which cannot meet the needs of this study. In addition to the 76 basic features collected based on CICFlowMeter, CIC-IDS2017 includes the timestamp, source IP and port, destination IP and port, protocol, and attack type. Topology structures in different periods can be obtained through the connection relationship's change between the source and destination IPs. Although IPs are not added in NSL-KDD2009, they can be obtained by parsing the original pcap of DARPA 98 and associated with the traffic record. Therefore, this study chooses two public datasets, NSL-KDD2009 and CIC-IDS2017, as the experimental datasets. The attack on the fifth day of CIC-IDS2017 was a DDoS attack, and its traffic was recorded in "Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv." The topology change is illustrated by taking every 50 traffic records of the fifth day as a unit. Further, four destination IPs, 192.168.10.25/3/50/9, are selected as examples, as shown in Figure 6.

In Figure 6, a typical DDoS topology A-B-C appears in period 51~100. Compared with the other two periods, the topology has changed significantly. From the number, there is a jump change 2-3-2; from the structure, there are different connection relationships: one-to-one, many-to-one, and one-to-many. The connections with the same source or destination address at various stages are also distinct. In brief, the topological changes like Figure 6 in the dataset can support the validation of the findings of this study.

The first dataset, NSL-KDD2009, is an improved version of KDD99 [58]. It optimizes some inherent problems of KDD99, such as the repeated identical records, missing data, and disproportionate training and testing data. This dataset covers 39 conventional attack methods, including six information gathering (probe), ten blocking attacks (DDoS), nine privilege acquisition (U2L), and 14 remote logins (R2L). Its traffic composition is shown in Table 2, where the bold characters indicate the attack types of the training data.

The second dataset, CIC-IDS2017, was developed by Sharafaldin et al. [59] to implement real network traffic collection based on user behavior simulation. It optimizes NSL-KDD2009 by adding the latest attack methods, expanding the feature set, and adding metadata. Fourteen new attack methods are included: two information gathering, six DDoS attacks, three privilege acquisition, and three remote logins. The dataset is not differentiated by training and test data but by acquisition period. Table 3 describes its composition.

4.3. Preprocessing. DDoS-related traffic records are extracted from NSL-KDD2009 and CIC-IDS2017 and classified according to different attack principles, as shown in Table 4. Due to erroneous data, improper formatting, and redundancy, these two datasets require preprocessing before neural network training.

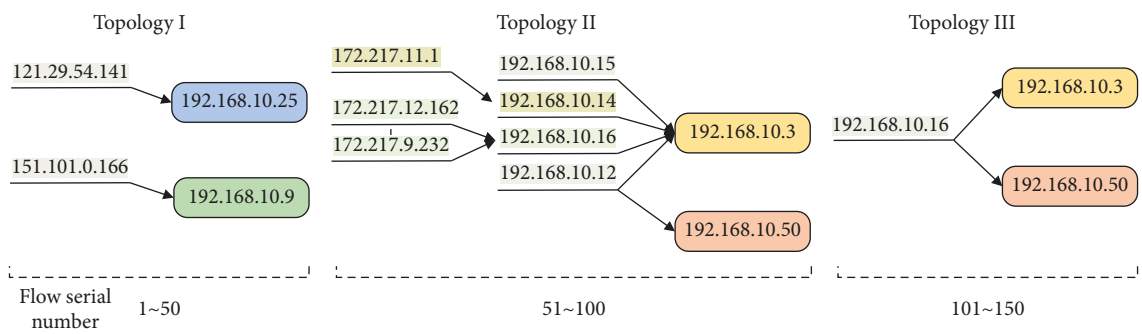


Figure 6: Example of topology structure changing in CIC-IDS2017.

Input: NetFlow items $F = \{f_1, f_2, \cdots, f_s\}$, connection relationship $C = < n_u \longrightarrow n_v >$ between node pair (n_u, n_v) in node set N, time series $T = \{t_1, t_2, \cdots, t_m\}$

Output: network structure graph G of node and edge distribution based on time series

Initialize graph $G = G_1 \cup G_2 \cup \cdots \cup G_m$, normalize traffic F, create mapping table $N \longrightarrow^{C} F$;
set edge list $P =$, time segment serial number i, item serial number j, items' quantity num;
for $(i = 1; i \leq m; i++)$ **do**
  $num = 0$;
    for $(\text{start\_time}(f_j) = \Delta t \cdot (i-1); \text{end\_time}(f_j) \leq \Delta t \cdot i; j++)$. **do**
    if $\sum_{o=1}^{j-1} t_o < \text{Start\_time}(f_j \in F) \| \text{End\_time}(f_j \in F) < \sum_{o=1}^{j} t_o$. **then**
      Calculate the total time $\widetilde{T}$ and average time $\overline{T}$ that fall into period t_i;
      $num = num + 1$;
    P.append(num);
    extract features on t_i, such as duration, interval, and bits;
    save as $edge^i_{num} = table(attr_1, attr_2, \cdots, attr_s)$, $C^i_{num} = < N_{src} \longrightarrow N_{de\ s} >$;
for $(i = 1; i \leq m; i++)$ **do**
  for $(j = 1; j \leq p[i-1]; j++)$**do**
    if graph G_m does not contain node N_{src} or N_{des} of C^i_j **then**
      add edge $\langle N_{src}, N_{des} \rangle$ to graph G_m;
    if graph G_m contains the edge between N_{src} and N_{des} **then**
      add extracted time attributes $edge^i_{num}$ to the edge $\langle N_{src}, N_{des} \rangle$;
    else
      create an edge between N_{src} and N_{des};
      add extracted time attributes $edge^i_{num}$ to the edge $\langle N_{src}, N_{des} \rangle$;
return G

Algorithm 1: Construction of dynamic DDoS topology graph based on time series.

Table 2: Traffic composition of the NSL-KDD2009 dataset.

Type	Attack method
Benign	normal
Probe	**ipsweep**, mscan, **nmap**, **portsweep**, saint, **satan**
DDoS	apache2, **back**, **land**, mailbomb, **neptune**, **pod**, processtable, **smurf**, **teardrop**, udpstrom
U2L	**bufferoverflow**, **loadmodule**, **perl**, ps, **rootkit**, snmpguess, sqlattack, worm, xterm
R2L	**ftp_write**, **guesspasswd**, httptunnel, **imap**, **multihop**, named, **phf**, **spy**, sendmail, snmpgetattack, **warezclient**, **warezmaster**, xlock

First, non-numeric features are standardized. There are two types of non-numeric features in the dataset: irrelevant and categorical strings. The former, such as Flow ID, Source IP/Port, Destination IP/Port, and Timestamp in CIC-IDS2017, have nothing to do with flow characterization and are removed from the dataset. For the latter, such as protocol_type, service, and flag in NSL-KDD2009, its classification includes detection information, which must be converted before use. There are two standard methods of string conversion: one-hot and normalized encoding [60]. One-hot sparse matrix has an enormous dimension and low computational efficiency. This study uses continuous integers [0,1,2, ...] to encode the classification and map it between zero and one through normalization.

Secondly, the classification labels are digitized. Unlike categorical features, the Euclidean distance between labels

Table 3: Traffic composition of the CIC-IDS2017 dataset.

Type	Attack method
Benign	BENIGN
Probe	Heartbleed, PortScan
DDoS	DoS Hulk, DDoS, DoS GoldEye, DoS slowloris, DoS Slowhttptest, bot
U2L	Web Attack-XSS, Infiltration, Web Attack-Sql Injection
R2L	FTP-Patator, SSH-Patator, Web Attack-Brute Force

Table 4: Number and classification of DDoS-related traffic records.

Datasets	Attack method	Sort	Number	Total
NSL-KDD 2009	0. normal	Normal 0	67,343	113,270
	1. neptune	Fast flow (1,2,3)	41,214	
	2. pod		201	
	3. smurf		2,646	
	4. back	Slow flow (4,5,6)	956	
	5. land		18	
	6. teardrop		892	
CIC-IDS 2017	0. BENIGN	Normal 0	2,273,097	2,655,751
	1. bot	Fast flow (1,2,3,4)	1,966	
	2. DoS Hulk		231,073	
	3. DDoS		128,027	
	4. DoS GoldEye		10,293	
	5. DoS slowloris	Slow flow (5,6)	5,796	
	6. DoS Slowhttptest		5,499	

used for error metrics cannot be represented by consecutive integers with uneven differences. After one-hot encoding, the distance between categories is the same and easy to matrix transformation. Therefore, we choose one hot to represent labels for efficient loss function computation. Then, the labels translate to normal traffic (1,0,0), fast traffic (0,1,0), and slow traffic (0,0,1). The columns marked with one here represent the corresponding classifications.

Thirdly, the invalid data are removed. There are two types of useless data: useless row or column data. Useless row data include rows containing ambiguous characters "NaN" and "Infinity." Useless column data include the column where the 43rd feature "success_pred" of NSL-KDD2009 is located. This feature denotes the number of correct predictions, regardless of traffic attributes. Both are deleted from the dataset directly.

Finally, topological features are extracted and normalized. According to Algorithm 1, the topological structure data are obtained. We use min-max to normalize features to cancel the influence of different scales. $x_{\min}$ and $x_{\max}$ are used to represent the minimum and maximum values of feature x, respectively, and its calculation formula is as follows:

$$\tilde{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}. \tag{14}$$

After the above processing, NSL-KDD2009 is transformed into a matrix consisting of 41 features and 45927 moments. CIC-IDS2017 is transformed into a matrix composed of 77 elements, with a total of 2827876 moments. The values in each matrix are between 0 and 1; then, we get normalized input data that are easy for deep learning architectures to process.

4.4. Performance Metrics. The detection accuracy verification of the proposed method includes distinguishing between background traffic and attack traffic and between regular traffic, fast attack traffic, and slow attack traffic. The former is a two-class problem, and the latter is a three-class problem. The two have different fine-grained partitions, so we use the targeted evaluation criteria to measure. For binary classification, the indicators are established through the confusion matrix, which has four essential components: true positive (TP), false positive (FP), true negative (TN), and false negative (FN). TP refers to the correct classification of positive samples as positive classes; FP is the proportion of negative samples misidentified as positive classes; TN refers to the correct classification of negative samples as negative classes; and FN refers to the misclassification of positive samples as negative classes. Based on the combination of these parts, we can get four performance metrics: precision, recall, precision, and F1-score. In particular, accuracy is the proportion of correctly classified samples $x_{correct}$ to the total samples x_{total}. The formula can be expressed as follows:

$$accuracy = \frac{x_{correct}}{x_{total}} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{15}$$

Recall is the ratio of correctly classified positive samples $\overline{x}_{correct}$ to the total positive samples $\overline{x}_{total}$. Its calculation is as follows:

$$recall = \frac{\overline{x}_{correct}}{\overline{x}_{total}} = \frac{TP}{TP + FN}. \tag{16}$$

Precision is the ratio of the correctly classified positive samples $\overline{x}_{correct}$ to the detected positive samples x_{total^+}. The formula is expressed as follows:

$$precision = \frac{\overline{x}_{correct}}{x_{total^+}} = \frac{TP}{TP + FP}. \quad (17)$$

F1-score refers to the weighted harmonic mean of recall and precision. It is used to measure the relative stability of the two. Its formula is as follows:

$$F1 - \text{score} = \left(\frac{\text{recall}^{-1} + \text{precision}^{-1}}{2}\right)^{-1} = \frac{2TP}{2TP + FP + FN}. \quad (18)$$

For triple classification, we adopt comprehensive metrics to measure the overall performance of the detection method, such as macro-average and micro-average. In particular, the macro-average calculates the mean of the metrics under all categories; the micro-average is an extension of the two-category metrics. Considering all class effects, we choose the macro-average as the three-class measure. Let n denote the number of classifications and Xi denote the ith value of indicator X, and the formula is as follows:

$$\text{Macro\_}X = \frac{1}{n}\sum_{i=1}^{n} X_i. \quad (19)$$

Furthermore, this study also studies the relationship between the evaluation result and the distribution of attack source IPs. The Pearson coefficient is used as the correlation measure. Let $\text{cov}(X, Y)$, σ_x, and σ_y denote the formulas for calculating the covariance and variance of variables x and y, respectively, and $E(X)$ and $E(Y)$ denote the expectation calculation. Its formula is as follows:

$$\rho(x, y) = \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E\left(X^2\right) - E^2(X)}\sqrt{E\left(Y^2\right) - E^2(Y)}}. \quad (20)$$

4.5. Results and Analysis. As shown in Table 4, the traffic composition of NSL-KDD2009 and CIC-IDS2017 is quite different. For these imbalanced datasets, 10-fold cross-validation is used for optimization. This method divides the sample into ten equal subsamples, sequentially uses one part for testing and the remaining nine parts for training, and takes the average of 10 times as the final result.

Testing the same method on different datasets may yield different results. Therefore, all DDoS detection methods are validated on the same dataset for comparative effectiveness. In addition, the same features in Table 1 were chosen for training for control variables.

The experiment consists of four parts: the comparison of two-classification methods, the comparison of three-classification methods, the correlation analysis of source IP distribution, and the analysis of method performance. Two-classification and three-classification methods are not always the same. Therefore, different baselines and state-of-the-art methods are selected as the comparison objects for the two comparison experiments.

4.5.1. Two-Classification Experiment. We choose six baseline and state-of-the-art methods as comparison objects in the binary classification experiment. In particular, baselines include the statistical method NaHiD [22], machine learning SKM-HFS [25], and random forest [26]; the latest methods include LUCID [30], DDoSNet [9], and BI-LSTM-GMM [34]. Baselines are reproduced with the Python library. In particular, NaHiD is obtained according to the mean and standard deviation of NumPy. SKM-HFS and random forest are calculated according to scikit-learn. These methods do not support GPU, so all baselines run on CPU. Furthermore, deep learning runs on GPU and compares the efficiency.

First, on NSL-KDD2009, GLD-Net is compared with six other means to verify binary classification performance. The epochs are set to 100, and the results are shown in Figure 7. As shown in Figure 7, these methods have different effects. Random forest achieves the best performance among the baselines with an accuracy of 0.896. The three deep learning methods show better detection performance with scores all above 0.9. In particular, BI-LSTM-GMM achieved the highest accuracy of 0.97 among the three. GLD-Net performs the highest metric on four indicators compared with the above techniques. Its accuracy reaches 0.991, which is 0.205 and 0.021 higher than the baselines and BI-LSTM-GMM, respectively.

Secondly, we also conducted a binary classification comparison experiment on CIC-IDS2017, and the result is shown in Figure 8. As shown in Figure 8, there is a significant gap between different methods. NaHiD still performs poorly, with both precision and recall not exceeding 0.65. The accuracy of deep learning is excellent, all exceeding 0.95. Compared with the other six methods, GLD-Net achieves the best performance, 0.191 and 0.0101 higher than the baseline and BI-LSTM-GMM.

Finally, we compare the accuracy distributions of GLD-Net and the three newest methods on two classifications, as shown in Figure 9. From Figure 9, GLD-Net has the highest accuracy and a concentrated distribution across multiple tests. The upper and lower quartile distances of LUCID and DDoSNet exceed 0.01, and the gap between the maximum and the minimum is close to 0.02. In contrast, the quantile distance of GLD-Net is only 0.003, more than four times lower than the average distance of 0.014 of other newest methods, showing the stability of the attack detection.

In summary, the accuracy of GLD-Net on NSL-KDD2009 and CIC-IDS2017 reaches 0.9914 and 0.9942, respectively. Compared with the latest methods, its average improvements are 0.021 and 0.0101; its stability increases four times, showing the best detection performance and stability.

4.5.2. Three-Classification Experiment. Given the low precision, statistical methods are usually not used to solve multi-classification problems. We choose five baselines and state-of-the-art methods as comparison objects in three-classification experiments. Among them, the baseline methods include decision tree [27] and random forest [26], and the latest methods include Stacked-DNN [36], FastGRNN [37], and ResNet [38]. Baselines are calculated according to scikit-learn.

First, we compare the three-classification performance of GLD-Net and five other methods on NSL-KDD2009. The

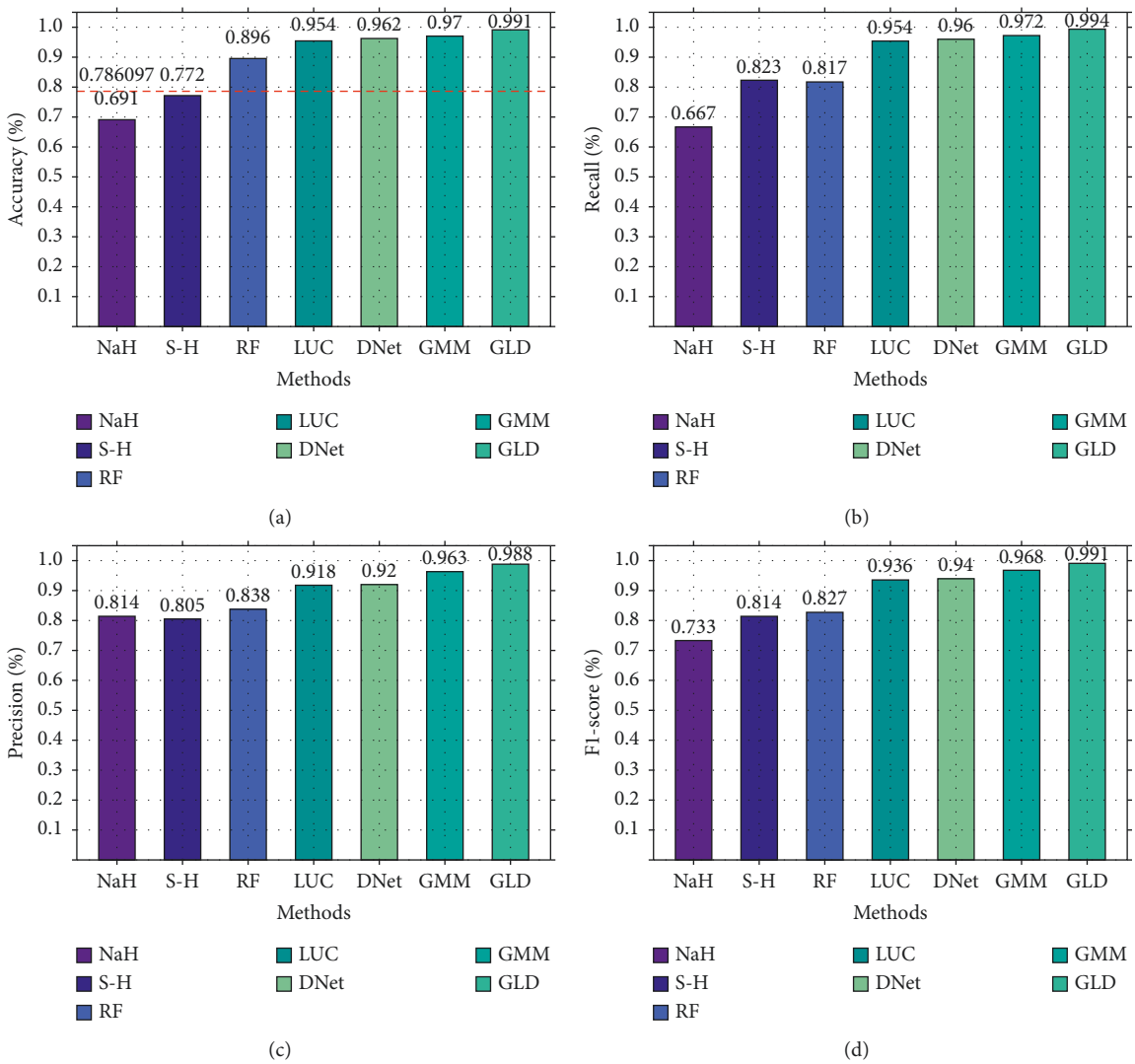


Figure 7: Comparison of binary classification performance between GLD-Net and other six methods on NSL-KDD2009.

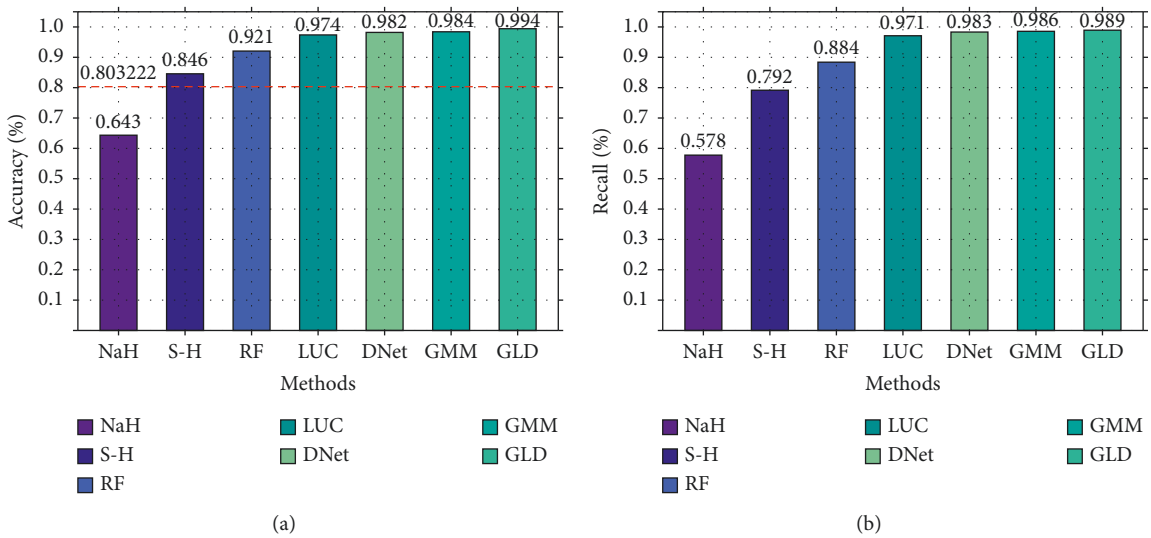


Figure 8: Continued.

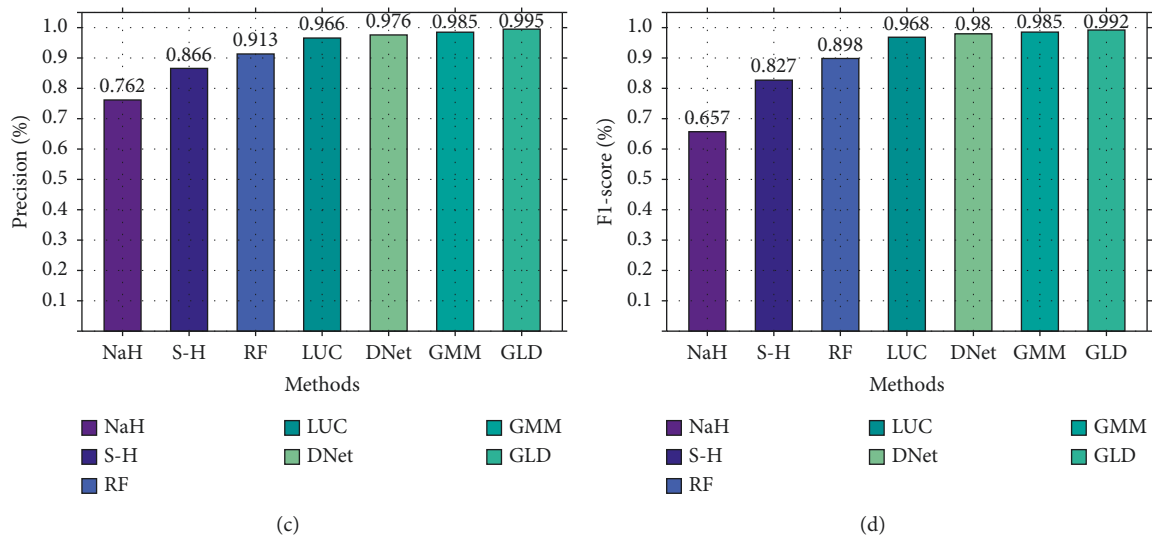


Figure 8: Comparison of binary classification performance between GLD-Net and other six methods on CIC-IDS2017.

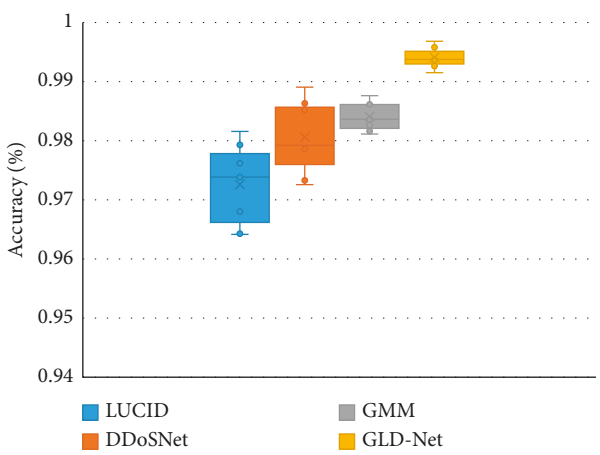


Figure 9: Performance distribution comparison between GLD-Net and three state-of-the-art methods.

results are shown in Figure 10. As shown in Figure 10, GLD-Net achieves the best three-classification performance compared with other methods. Its macro-accuracy reaches 0.958, an average improvement of 0.174 and 0.047 over baselines and the newest techniques.

Secondly, based on CIC-IDS2017, the three-classification experiment was performed, and the results are shown in Figure 11. As shown in Figure 11, GLD-Net achieves the best performance, with the accuracy and F1-score reaching 0.925 and 0.924, respectively. Compared to Figure 10, deep learning performance degrades partly due to varying traffic types in datasets. Compared with baselines and the newest methods' averages, GLD-Net improves the accuracy by 0.131 and 0.019, respectively.

Finally, we compare the difference in the confusion matrix among GLD-Net and three other methods, as shown in Figure 12. The colors in the graph range from white to blue, representing accuracy from 0 to 1.0. The darker the blue, the higher the ratio. As shown in Figure 12, GLD-Net achieves good results in traffic type detection, and TP exceeds 0.9. While the normal flow of decision tree, slow flow of random forest, and fast flow of ResNet have lower TP, which is 0.62, 0.75, and 0.86, respectively. Detection based on GLD-Net has better balance and can meet the needs of fine-grained discrimination.

Compared with the state-of-the-art methods, the three-class accuracy of GLD-Net is improved by 0.047 and 0.02, respectively. Its availability is also increased by 0.023, showing better performance and broad applicability.

4.5.3. Distribution Correlation Analysis. First, the correlation between TP of attack detection and the number of attack source IPs is investigated. The results are shown in Figure 13. It can be seen from Figure 13 that there is a positive correlation between the TP of GLD-Net and the number of attack source IPs. Its Pearson coefficient is 0.789; greater than 0.75 shows a strong correlation.

Secondly, the correlation between TP and the IP hop count (the average hop count of all leaf nodes) is examined. The result is shown in Figure 14. As shown in Figure 14, a positive correlation exists between TP and IP hops' numbers using GLD-Net. The Pearson coefficient is 0.695, close to 0.7, indicating that the two have specific relevance.

Thirdly, at the network level, the number of subnets [61] and closeness centrality [62] are used to investigate the aggregation and distribution of attack sources. The results are shown in Figure 15. It can be seen from Figure 15 that the number of subnetworks and closeness centrality increase with the rise of TP. After 200 cycles of calculation and taking the mean value, the Pearson coefficients obtained are 0.812 and 0.834, respectively, and over 0.8 indicates a strong correlation.

Finally, GLD-Net is compared with other detection methods using non-topological features as input in the correlation between TP and attack source IP number, hop number, subnet number, and closeness centrality. The results

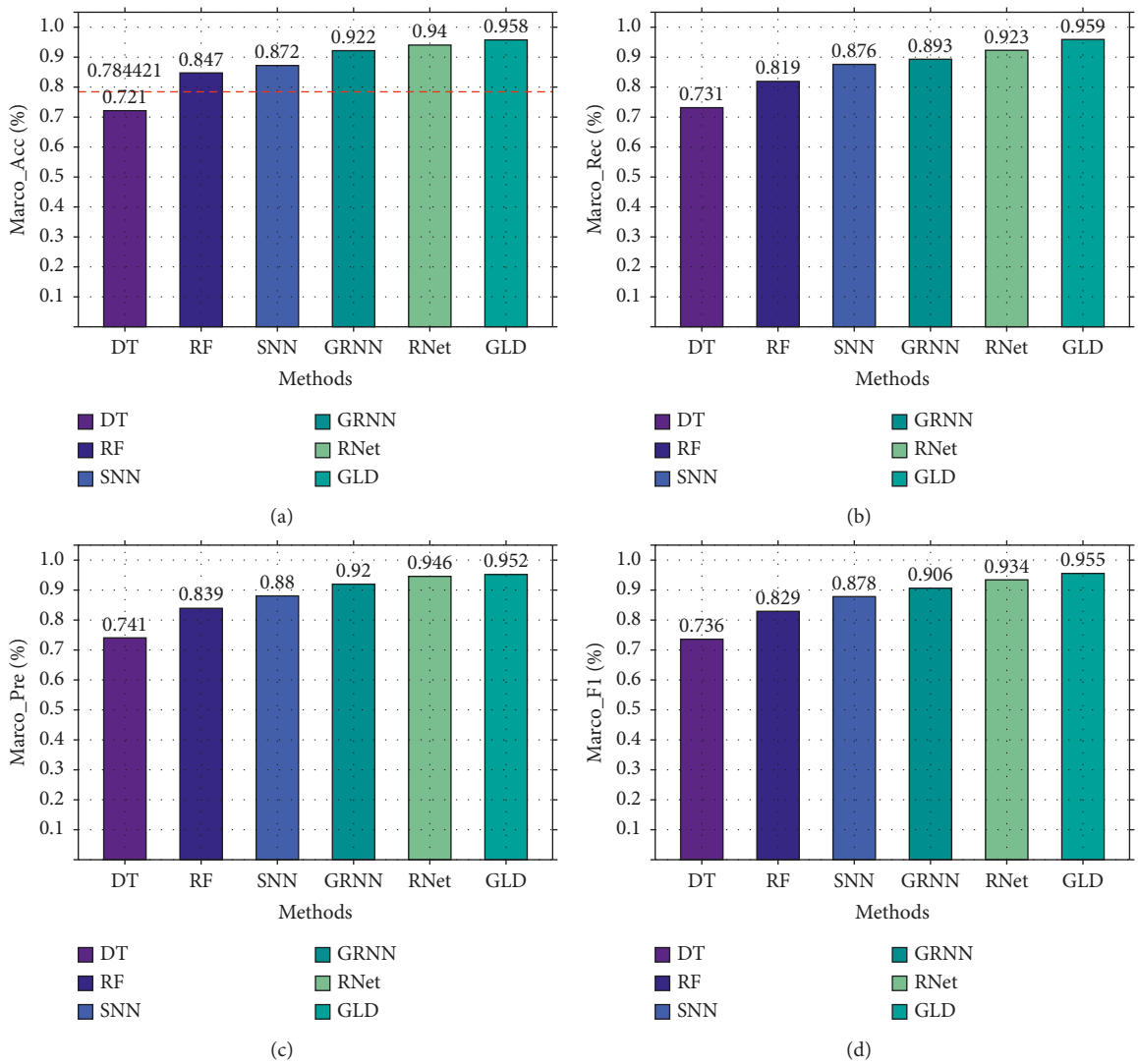


Figure 10: Three-class performance comparison of GLD-Net and other five methods on NSL-KDD2009.

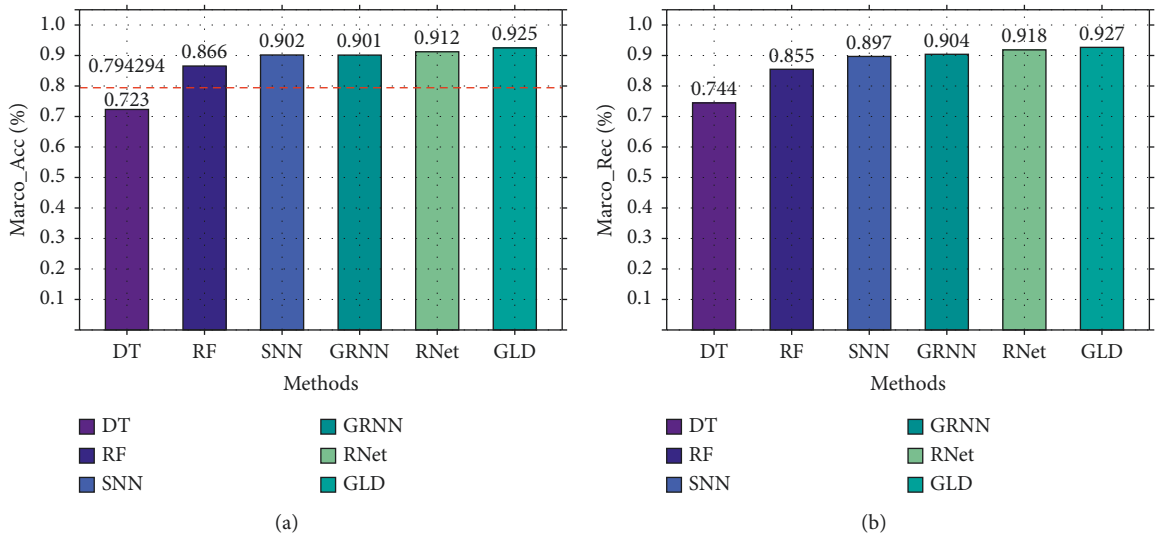


Figure 11: Continued.

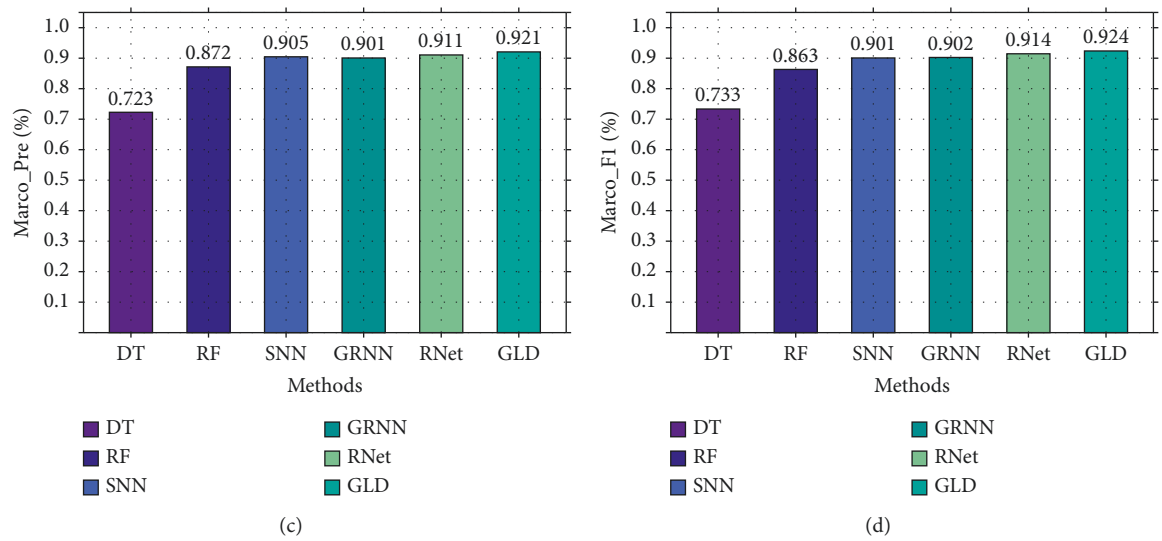


Figure 11: Three-class performance comparison of GLD-Net and other five methods on CIC-IDS2017.

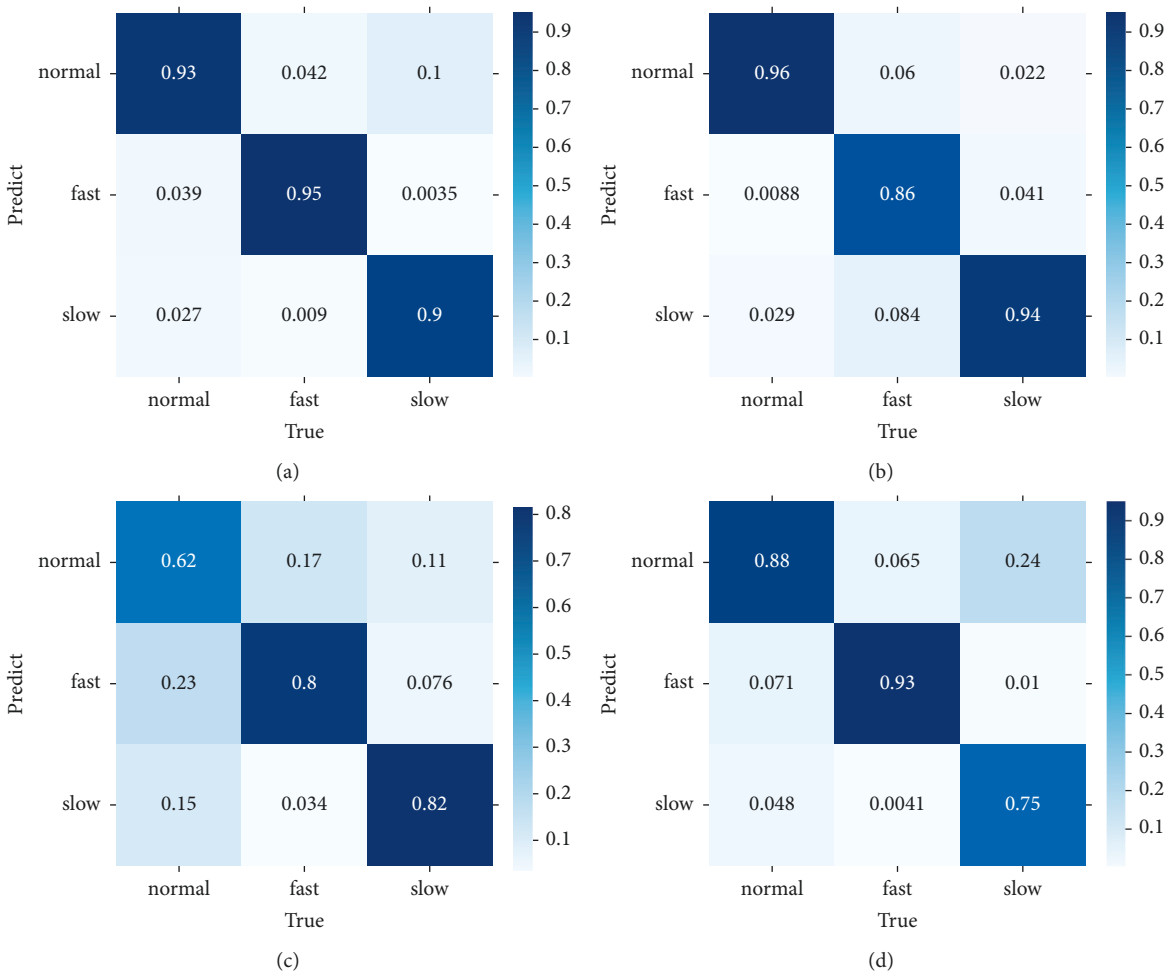


Figure 12: Confusion matrix of GLD-Net and three other methods for three-class detection. (a) GLD-Net. (b) ResNet. (c) Decision tree. (d) Random forest.

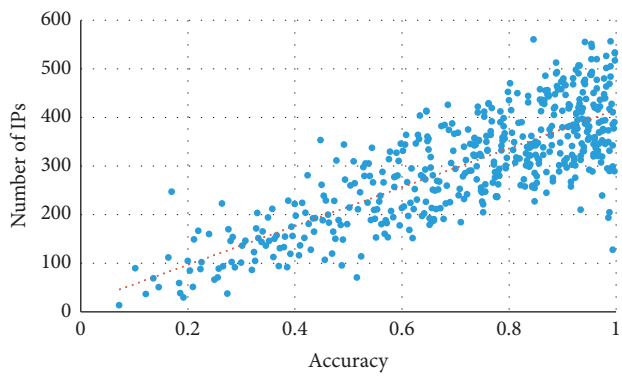


Figure 13: Correlation between TP and IP number under GLD-Net.

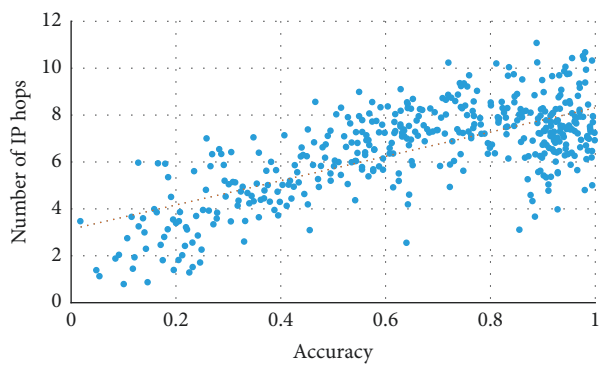


Figure 14: Correlation between TP and IP hop count under GLD-Net.

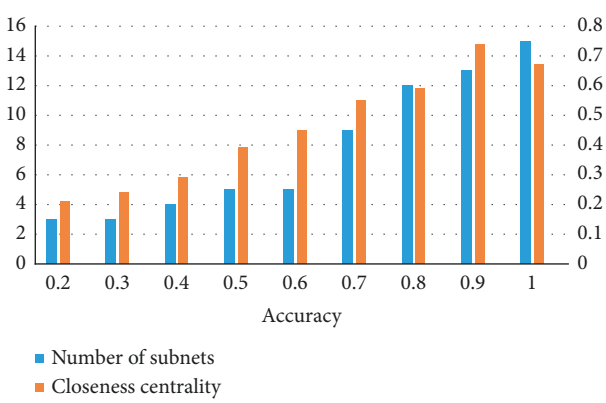


Figure 15: Variation trend of detection TP, number of subnets, and closeness centrality of GLD-Net.

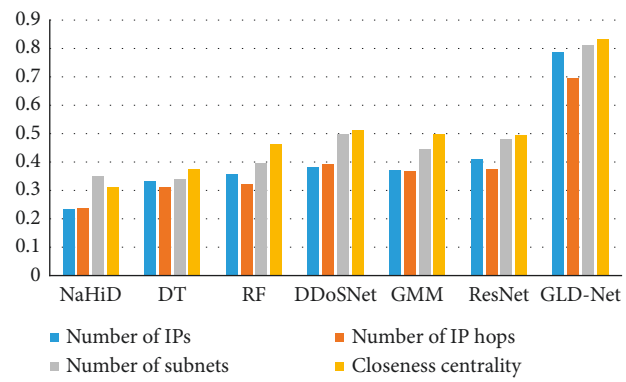


Figure 16: Comparison of correlation coefficients between GLD-Net and other six methods for detecting TP and four distribution metrics.

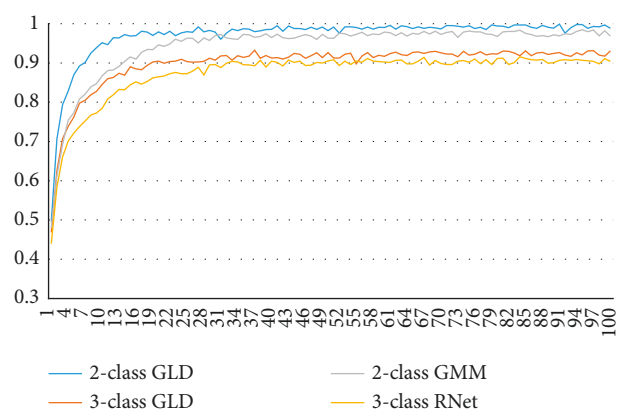


Figure 17: Comparison of accuracy trends of GLD-Net and other methods on two-classification and three-classification detection during 100 epoch training.

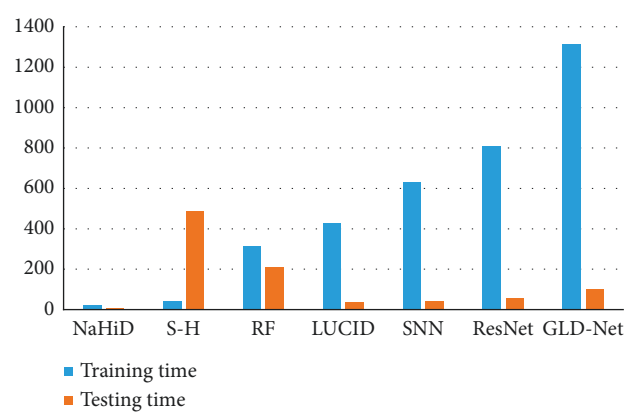


Figure 18: Comparison of training and testing times for GLD-Net and other six standard methods.

are shown in Figure 16. It can be seen from Figure 16 that the correlation coefficients of the comparative methods are primarily in the range 0.3~0.5. The correlation coefficients of GLD-Net fall on the interval 0.7~0.83, and the average increases in the four correlations are 0.441, 0.36, 0.393, and 0.391, respectively. The rise of around 0.4 shows that GLD-Net is more capable of inferring the distribution of attack sources based on the detection results than other methods.

In short, we found that the four attack source distribution indicators have correlation coefficients with TP of GLD-Net reaching 0.789, 0.695, 0.812, and 0.834, respectively. Compared with other methods that take non-topological features as input, the average improvement is 0.441, 0.36, 0.393, and 0.391, respectively. The strong correlation of 0.7~0.83 supports using the evaluation result to infer the distribution of attack sources. For example, when TP is 0.8, combined with Figures 13–15, it can be deduced that the

attack source IPs' number is around 350, hops' number is 7, subnets' number is 12, and closeness centrality is 0.59.

4.5.4. Efficiency Analysis. First, we investigate the accuracy variation of GLD-Net during 100 epochs. We choose BI-LSTM-GMM and ResNet as the two-classification and three-classification comparison objects. The results are shown in Figure 17. It can be seen from Figure 17 that the accuracy of BI-LSTM-GMM and ResNet tends to be stable at the 22nd and 28th epochs, respectively. The accuracy of GLD-Net gradually stabilized at the 11th or 16th epoch. In contrast, our proposed model can converge faster and achieve better performance.

Secondly, we also analyze the training and testing time of GLD-Net. We selected six other methods for comparison, and the result is shown in Figure 18. As shown in Figure 18, the average training time of GLD-Net reaches 1312 s, 4.14 times that of random forest and 1.59 times that of ResNet. The test time of GLD-Net is 107 s, which is 8.16% of its training time and 50.47% of random forest. The results show that although the training time is slightly longer than other deep learning methods, the test time is still within the tolerance range. This overhead is worthwhile compared with the improved accuracy.

To summarize, GLD-Net converges in 11 or 16 epochs in binary or multi-classification, an average of 11.5 epochs ahead of the best other methods. The average training and testing times of GLD-Net are 1312 s and 107 s, respectively. Its training time is 4.14 times that of random forest; the test time is only 50.47% of that of random forest, indicating the better practical efficiency of GLD-Net.

5. Conclusions

In this study, we propose GLD-Net, a new deep learning DDoS attack detection method based on topological and flow features. A graph model is introduced for feature extraction. Traffic features are added to edge features, and node features represent topological features. A dynamic DDoS topology feature construction algorithm is proposed by calculating the feature table and mapping topological entities on the time series. For non-Euclidean input, GAT mines complex topological relationships, and LSTM extracts sequence correlation in vectors. Finally, the fully connected layer obtains the traffic type through data integration. The experimental results show that DDoS detection with topology and flow features as input can solve the problem of limited accuracy due to incomplete feature input. It can also estimate the distribution of attack sources based on the detection results, which facilitates the rapid and accurate deployment of subsequent security strategies. In the future, we also need to design a more fine-grained differentiation model for different traffic types and explore unknown traffic discovery techniques. These related researches will expand the scope of application to escort system security in the current increasingly complex network confrontation situation.

6. Future Research

We verified the effectiveness of the proposed DDoS detection method GLD-Net through comparative experiments on two network security datasets. Nevertheless, there are still the following issues to be studied.

Question 1. We mainly verify the method's performance in distinguishing normal, fast, and slow traffic for the multi-classification. Whether this method is suitable for more fine-grained differentiation, such as HTTP applications, requires further verification.

Question 2. The neighborhood computation efficiency of GAT is not high, and it cannot cope with real-time training. Next, we need to study a lightweight GAT structure to improve the execution speed.

Question 3. This research mainly focuses on relationship mining in the existing traffic. It cannot discover unknown traffic such as 0-day attacks. Therefore, it is necessary to study novel deep learning methods that simultaneously identify known and anonymous traffic.

Data Availability

The NSL-KDD2009 and CIC-IDS2017 datasets used to support the finding of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this study.

Acknowledgments

The authors thank the deep learning team of the State Key Laboratory of Mathematical Engineering and Advanced Computing for their help and encouragement. Primarily, Dr. Qiu proposed significant revisions for the experiments in this article.

References

[1] H. Abusaimeh, "Distributed denial of service attacks in cloud computing," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 6, 2020.

[2] A. Agarwal, M. Khari, and R. Singh, "Detection of DDOS Attack Using Deep Learning Model in Cloud Storage Application," *Wireless Personal Communications*, 2021.

[3] F. Alatawi, "Defense mechanisms against distributed denial of service attacks: comparative review," *Journal of Information Security and Cybercrimes Research*, vol. 4, no. 1, pp. 81–94, 2021.

[4] Z. Wu, Q. Wei, K. Ren, and Q. Wang, "Dynamic defense for DDoS attack using openflow-based switch shuffling approach," *Dianzi Yu Xinxi Xuebao/Journal of Electronics and Information Technology*, vol. 39, no. 2, 2017.

[5] K. Singh, K. Singh Dhindsa, and B. Bhushan, "Distributed Defense: An Edge over Centralized Defense against DDos

Attacks," *International Journal of Computer Network and Information Security*, vol. 9, 2017.

[6] Y. Liu, T. Zhi, M. Shen, L. Wang, Y. Li, and M. Wan, "Software-defined DDoS detection with information entropy analysis and optimized deep learning," *Future Generation Computer Systems*, vol. 129, pp. 99–114.

[7] M. Mittal, K. Kumar, and S. Behal, "Deep Learning Approaches for Detecting DDoS Attacks: A Systematic Review," *Soft Computing*, 2022.

[8] A. Pektaş and T. Acarman, "Deep learning to detect botnet via network flow summaries," *Neural Computing & Applications*, vol. 31, no. 11, pp. 8021–8033, 2019.

[9] M. S. Elsayed, N. A. Le-Khac, S. Dev, and A. D. Jurcut, *DDoSNet: A Deep-Learning Model for Detecting Network Attacks*, in *Proceedings of the 2020 IEEE 21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, Cork, Ireland, Augest2020.

[10] J. He, Y. Tan, W. Guo, and M. Xian, "A Small Sample DDoS Attack Detection Method Based on Deep Transfer Learning," in *Proceedings of the 2020 International Conference on Computer Communication and Network Security (CCNS)*, Xi'an, China, Augest2020.

[11] C. Liaskos and S. Ioannidis, "Network topology effects on the detectability of crossfire attacks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1682–1695, 2018.

[12] K. Sharma and A. Mukhopadhyay, "Kernel naïve Bayes classifier-based cyber-risk assessment and mitigation framework for online gaming platforms," *Journal of Organizational Computing & Electronic Commerce*, vol. 31, no. 4, pp. 343–363, 2021.

[13] Q. Shafi and A. Basit, "DDoS Botnet Prevention Using Blockchain in Software Defined Internet of Things," in *Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, Islamabad, Pakistan, Janunary2019.

[14] P. Veličković, A. Casanova, P. Liò, G. Cucurull, A. Romero, and Y. Bengio, "Graph Attention Networks," 2018, https://arxiv.org/abs/1710.10903.

[15] C. Zhang, J. Cheng, X. Tang, V. S Sheng, Z. Dong, and J. Li, "Novel DDoS feature representation model combining deep belief network and canonical correlation analysis," *Computers, Materials & Continua*, vol. 61, no. 2, pp. 657–675, 2019.

[16] Y. Cui, Q. Qian, C. Guo et al., "Towards DDoS detection mechanisms in Software-Defined Networking," *Journal of Network and Computer Applications*, vol. 190, Article ID 103156.

[17] X. Yuan, C. Li, and X. Li, "DeepDefense: Identifying DDoS Attack via Deep Learning," in *Proceedings of the 2017 IEEE International Conference on Smart Computing (SMARTCOMP)*, Hong Kong, China, May 2017.

[18] M. Idhammad, K. Afdel, and M. Belouch, "Semi-supervised machine learning approach for DDoS detection," *Applied Intelligence*, vol. 48, no. 10, pp. 3193–3208, 2018.

[19] R. Doshi, N. Apthorpe, and N. Feamster, "Machine Learning DDoS Detection for Consumer Internet of Things Devices," in *Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW)*, Francisco, CA, USA, May 2018.

[20] F. S. De Lima Filho, F. A. F. Silveira, A. De Medeiros Brito Junior, G. Vargas-Solar, and L. F. Silveira, "Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning," *Security and Communication Networks*, 2019.

[21] R. K. Chouhan, M. Atulkar, and N. K. Nagwani, "A Framework to Detect DDoS Attack in Ryu Controller Based Software Defined Networks Using Feature Extraction and Classification," *Applied Intelligence*, 2022.

[22] N. Hoque, H. Kashyap, and D. K. Bhattacharyya, "Real-time DDoS attack detection using FPGA," *Computer Communications*, vol. 110, pp. 48–58, 2017.

[23] L. D. Tsobdjou, S. Pierre, and A. Quintero, "An Online Entropy-Based DDoS Flooding Attack Detection System with Dynamic Threshold," *IEEE Transactions on Network and Service Management*, vol. 19, 2022.

[24] A. Ahalawat, K. S. Babu, A. K. Turuk, and S. Patel, "A low-rate DDoS detection and mitigation for SDN using Renyi Entropy with Packet Drop," *Journal of Information Security and Applications*, vol. 68, Article ID 103212, 2022.

[25] Y. Gu, K. Li, Z. Guo, and Y. Wang, "Semi-supervised k-means ddos detection method using hybrid feature selection algorithm," *IEEE Access*, vol. 7, pp. 64351–64365, 2019.

[26] S. Pande, A. Khamparia, D. Gupta, and D. N. H. Thanh, "DDOS detection using machine learning technique," in *Studies in Computational Intelligence*vol. 921, 2021.

[27] I. Cvitic, D. Perakovic, B. B. Gupta, and K. K. R. Choo, "Boosting-based DDoS detection in Internet of things systems," *IEEE Internet of Things Journal*, vol. 9, no. 3, pp. 2109–2123, 2022.

[28] S. Kumar, H. G Sastry, V. Marriboyina et al., "Ddos detection in sdn usingmachine learning techniques," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 771–789, 2022.

[29] X. Liang and T. Znati, "A Long Short-Term Memory Enabled Framework for DDoS Detection," in *Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, Decemeber2019.

[30] R. Doriguzzi-Corin, S. Millar, S. Scott-Hayward, J. Martinez-Del-Rincon, and D. Siracusa, "Lucid: a practical, lightweight deep learning solution for DDoS attack detection," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 876–889, 2020.

[31] A. E. Cil, K. Yildiz, and A. Buldu, "Detection of DDoS attacks with feed forward based deep neural network model," *Expert Systems with Applications*, vol. 169, Article ID 114520.

[32] J. Boonchai, K. Kitchat, and S. Nonsiri, "The classification of DDoS attacks using deep learning techniques," in *Proceedings of the 2022 7th International Conference on Business and Industrial Research*, pp. 544–550, 2022.

[33] L. Wang and Y. Liu, "A DDoS Attack Detection Method Based on Information Entropy and Deep Learning in SDN," in *Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chongqing, China, June 2020.

[34] C. S. Shieh, W. W. Lin, T. T. Nguyen, C. H. Chen, M. F. Horng, and D. Miu, "Detection of unknown ddos attacks with deep learning and Gaussian mixture model," *Applied Sciences*, vol. 11, no. 11, p. 5213, 2021.

[35] J. G. Almaraz-Rivera, J. A. Perez-Diaz, and J. A. Cantoral-Ceballos, "Transport and application layer DDoS attacks detection to IoT devices by using machine learning and deep learning models," *Sensors*, vol. 22, no. 9, p. 3367, 2022.

[36] P. Toupas, D. Chamou, K. M. Giannoutakis, A. Drosou, and D. Tzovaras, "An Intrusion Detection System for Multi-Class Classification Based on Deep Neural Networks," in *Proceedings of the 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, Boca Raton, FL, USA, December 2019.

[37] H. Alzahrani, M. Abulkhair, and E. Alkayal, "A multi-class neural network model for rapid detection of IoT botnet

attacks," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 7, 2020.

[38] F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad, and G. A. Shah, "IoT DoS and DDoS Attack Detection Using ResNet," in *Proceedings of the 2020 IEEE 23rd International Multitopic Conference (INMIC)*, Bahawalpur, Pakistan, November2020.

[39] M. Rusyaidi, S. Jaf, and Z. Ibrahim, "Detecting distributed denial of service in network traffic with deep learning," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 1, 2022.

[40] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proença, "Adversarial Deep Learning approach detection and defense against DDoS attacks in SDN environments," *Future Generation Computer Systems*, vol. 125, pp. 156–167, 2021.

[41] R. Doriguzzi-Corin and D. Siracusa, "FLAD: adaptive federated learning for DDoS attack detection," vol. 1–12, 2022, https://arxiv.org/abs/2205.06661.

[42] A. Lohachab and B. Karambir, "Critical analysis of DDoS—an emerging security threat over IoT networks," *Journal of Communications and Information Networks*, vol. 3, no. 3, pp. 57–78, 2018.

[43] H. Kousar, M. M. Mulla, P. Shettar, and D. G. Narayan, "Detection of DDoS Attacks in Software Defined Network Using Decision Tree," in *Proceedings of the 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT*, Bhopal, India, June2021.

[44] J. Liang, J. Chen, X. Zhang, Y. Zhou, and J. Lin, "One-hot encoding and convolutional neural network based anomaly detection," *Qinghua Daxue Xuebao/Journal of Tsinghua University*, vol. 59, no. 7, 2019.

[45] A. A. Abdulrahman and M. K. Ibrahem, "Evaluation of DDoS attacks detection in a new intrusion dataset based on classification algorithms," *Iraqi Journal of Information & Communications Technology*, vol. 1, no. 3, pp. 49–55, 2019.

[46] X. Tang, R. Cao, J. Cheng, D. Fan, and W. Tu, "DDoS attack detection method based on V-Support vector machine," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*vol. 11983, , 2019.

[47] Ö. Kasim, "An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks," *Computer Networks*, vol. 180, Article ID 107390, 2020.

[48] M. Grassia, M. De Domenico, and G. Mangioni, "Machine learning dismantling and early-warning signals of disintegration in complex systems," *Nature Communications*, vol. 12, no. 1, p. 5190, 2021.

[49] O. E. Tayfour and M. N. Marsono, "Collaborative detection and mitigation of DDoS in software-defined networks," *The Journal of Supercomputing*, vol. 77, no. 11, pp. 13166–13190, 2021.

[50] X. Zhang, Y. Zou, and W. Shi, "Dilated convolution neural network with LeakyReLU for environmental sound classification," in *Proceedings of the International Conference on Digital Signal Processing*, London, UK, 2017-August.

[51] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Physica D: Nonlinear Phenomena*, vol. 404, Article ID 132306, 2020.

[52] X. Huang, H. Tan, G. Lin, and Y. Tian, "A LSTM-Based Bidirectional Translation Model for Optimizing Rare Words and Terminologies," in *Proceedings of the 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD)*, Chengdu, China, May2018.

[53] L. Zhang, Z. Shi, M. M. Cheng et al., "Nonlinear regression via deep negative correlation learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 982–998, 2021.

[54] L. Dhanabal and S. P. Shantharajah, "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 6, 2015.

[55] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, Article ID 107247.

[56] J. L. Leevy and T. M. Khoshgoftaar, "A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 Big Data," *Journal of Big Data*, vol. 7, no. 1, p. 104, 2020.

[57] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *Proceedings - International Carnahan Conference on Security Technology*, Chennai, India, 2019-October.

[58] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," in *Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, ON, Canada, July2009.

[59] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, 2018-January.

[60] P. Rodríguez, M. A. Bautista, J. Gonzàlez, and S. Escalera, "Beyond one-hot encoding: lower dimensional target embedding," *Image and Vision Computing*, vol. 75, pp. 21–31, 2018.

[61] M. Dimolianis, A. Pavlidis, and V. Maglaris, "A Multi-Feature DDoS Detection Schema on P4 Network Hardware," in *Proceedings of the 2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Paris, France, Feburary2020.

[62] J. Li, L. Lyu, X. Liu, X. Zhang, and X. Lyu, "FLEAM: a federated learning empowered architecture to mitigate DDoS in industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 4059–4068, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6732343, 26 pages
https://doi.org/10.1155/2022/6732343

Research Article

An Overview of Recent Advances of Resilient Consensus for Multiagent Systems under Attacks

Muhammad Muzamil Aslam,[1,2,3] **Zahoor Ahmed**,[4] **Liping Du**,[1] **Muhammad Zohaib Hassan**,[4] **Sajid Ali**,[3] **and Muhammad Nasir**[5]

[1]*School of Computer and Communication Engineering, University of Science and Technology, Beijing 100083, China*
[2]*Department of Electronics and Communication Engineering, University of Science and Technology China (USTC), Hefei, China*
[3]*Department of Information Sciences, University of Education Multan Campus, Lahore, Pakistan*
[4]*Department of Automation, Shanghai Jiaotong University, Shanghai 200240, China*
[5]*Department of Computer Engineering, The University of Lahore, Lahore, Pakistan*

Correspondence should be addressed to Liping Du; dlp2001@ies.ustb.edu.cn

Received 18 April 2022; Revised 26 May 2022; Accepted 31 May 2022; Published 2 August 2022

Academic Editor: Konstantinos Demertzis

Consensus control of multiagent systems (MASs) has been one of the most extensive research topics in the field of robotics and automation. The information sharing among the agents in the MASs depends upon the communication network because the interaction of agents may affect the consensus performance of the agents in a communication network. An unexpected fault and attack may occur on one agent and can propagate through the communication network into other agents. Thus, this may cause severe degradation of the whole MASs. In this paper, we first discussed MAS technologies. After that available technologies for the modeling of attacks and fundamental issues due to attacks on MAS attacks were discussed. We also introduced cooperative attack methodologies and model-based attack methodology. Objective of this article is to provide comprehensive study on recent advances in consensus control of MASs under attacks covering the published results until 2021. This survey presents different kinds of attacks, their estimation and detection, and resilient control against attacks. At the end, the survey accomplishes some potential recommendations for future direction to solve the key issues and challenges reported for secure consensus control of MASs.

1. Introduction

With advancement of communication and computer technologies, coordination control of MASs has got a lot of attention of researchers in different areas of engineering due to its broad applications in order to attain craving physical performances [1, 2]. There is speedy increment in progress of MASs because of improvement in communication, computing, and relevant technologies. MASs are also known as integration of communications, computations, physical processes, and controls which can play a key role in infrastructure [3, 4]. Cyber threats posture an actual and increasing problem, and to date, many countries struggle to counter them have lagged. However, capability to protect in contradiction of an attack or invasion must be upheld, and any country would be well served by discouraging its opponents from acting in the first place, at least when it comes to the most serious actions, namely, cyber warfare. There is vital role of cyber security in the era of technology also is biggest challenge to secure information in these days of technology. The first idea regarding cyber security in our mind is "cybercrime, cyber threat, or cyberattack"; those are increasing day by day. Several private and public sectors are taking various measures to secure such crimes, attacks, or threats.

Due to variety of applications [4], MASs have become research interesting area. Despite usefulness, MASs have a lot of security risks because of the interconnections of

various components and technologies. Since MASs are susceptible to spiteful threats and may result in the ailment of social life or loss of economic benefits. So, attack issues need considerable attention in the practical systems. In a smart transport system, an automatic controlled vehicle network can be a useful solution for an efficient and secure transport system and assists in problem-solving such as environmental protection, energy conservation, road congestion, and road accident. In contrast to the already present single-agent system, MASs are upgraded and scalable for enhancing energy efficiency and durability because they have the built-in ability for cooperative learning through the automatic decision. Recently, issues of MASs and their analysis have been well discussed in [5] but security issues of MASs have not been studied in the article and these systems are unsafe from cyberattacks and physical faults. So, we need to adjust, wait, or abandon at the time of system failure or attack [6]. MASs trust and access control issues have been highlighted. The researchers in [7, 8] highlighted MAS fault tolerant control procedure, and basic study was on reconfiguration way of topology. A very low-level attack or fault on an agent can disturb the system function and can cause damage to the system badly. Researchers have done wide research on cyber security and physical security in which detection and diagnosis, fault estimation problem, secure consensus, attack detection, and fault tolerant control were studied, and such research provides some basic methodologies and systematic suggestions for effective betterment of MASs security In Figure 1, there is framework of the intelligent transport system. Various kinds of attacks are present possible on the MASs such as man-in-middle attacks, deception attacks, cross-site scripting attacks, denial-of-service attacks, drive-by attacks, eavesdropping attacks, malware attacks, password attacks, phishing attacks, and SQL injection attacks.

In several fields of human activities, MASs have gained more attention, particularly the places where we need physical equipment, processes for control, and cooperating with the system and human, e.g., building automation networks, smart grids, and water sewerage plants. The progressing technologies, e.g., Industrial Internet or Industry 4.0 [9], are key points of MAS's importance; this idea transition will involve increasing autonomy, automation, and fulfilling fresh arrival of the production. The main motivation which is forcing the development of MASs is the conjunction of physical process and computational aptitudes, and components of the process happening in the physical surrounding are requirements. MAS scaling, e.g., large and small, is differentiated by the number of components involved [5].

These MASs are widely and geographically dispersed, heterogeneous, federated, and the critical system of life in which actuators and sensors are embedded networked for control physical world, monitoring, and sensing. There is no doubt that reserved scheduling via different own network or shared network plays a key role in the working of MAS. The selection of actuator or sensor for best performance of a specific action is an important decision, and also, proper management of actions is also important. Because of limited technology or physical constraints, there is risk of data transmission among network components, sensors, and actuators without any specific security instructions. In one point of view, interconnection of large scale of network components puts it in more complexity to save it from innate physical liabilities there. In another point of view, cyber integration generally sets up an underscore on protection and suppleness against threats and unseen from cyberspace [10, 11]. Hence, several new challenges are known to general or traditional control, software theory, and communication [12, 13].

The susceptibility of MASs in cyber parts permits attacks into the system in nonproductive and silent approach, e.g., insertion of some virus or worm in network component computerization or layers that should have medium access, could cause coordination packets distraction. In addition, there can be prohibited access of attacker to the monitoring centers and gaining decryption key in gain of normal operation [3]. It is clear that when there is no strict security protection, the attacker can either freely harm system dynamic or bring any agitation. For solution of optimization problem in selection of input or output that highlights attacks with least detect ability and maximum impact; there is a survey [14].

MAS consensus problem has been widely addressed [15, 16] with time delay. In MASs, we can found two types of delay, communication time delay (CTD) and input time delay (ITD). Here, CTD is linked between two agents interconnected time while ITD is related to data processing time. ITD may occur when controllers, actuators, or other components are linked to the network, while because of CTD, each agent may get information from a very near agent. CTD has been used in most real used applications such as computational and PWM delays in an LCL type grid-linked invertor and in controller area network bus of the distributed control network, induces delay. Here, we would like to discuss consensus problem of continuous time linear homogeneous MAS with CTD and ITD. ITD is considered multiple input time delays while CTD is single time delay. MAS consensus [14] problem has been widely studied from several studies, e.g., transportation control, coordinated defense system [17], power systems, and smart grids [18]. Consensus problem target is used to design a control law so that all agents can reach to a single point [14]. However, several present results on consensus problem were supposed as an ideal condition for investigation, e.g., secure network communication in between agents and unlimited resources network. In reality, there is presence of several impulsive issues, prime to a high security predicament and having influence on communication topology [14]. Nasty agents can grab this failing to interject communication between agents and disturb overall system stability in the multiagent network, while we can divide cyberattacks into the following cases on MASs, first is when there is an attack of nasty agent on agent and second is when nasty agent attacks to destroy the communication medium. In both cases, the communication graph is detached.

There is no high-level protection against future MAS threats or attacks. There are several types of key challenges

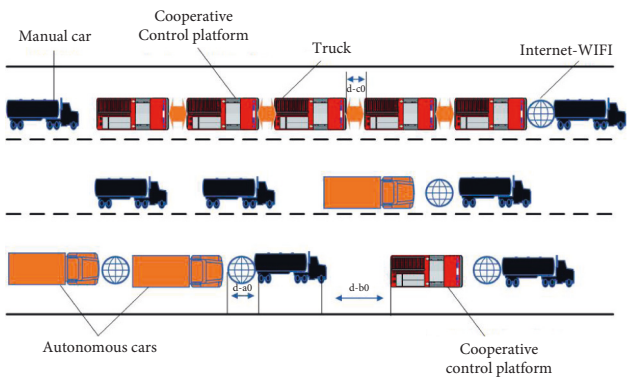


Figure 1: Intelligent transport system in coordination.

regarding MAS security, and these threats may be from the system inside such as sudden failure of the system, mobile networks security risks regarding human health, communication protocol weakness in smart grids, and limitations of physical systems. There is increasing requirement in new control system accuracy and reliability of each component. Any type of failure or fault of a system or component may lead to system performance degradation or reason for system instability or dramatic change in system operation [19].

Because communication networks consist of interacting agents which require safety and reliability to gain cooperative control and given challenges which are absent in single-agent systems such as MAS interconnection properties confuse the fault detection and identification and make worldwide and exact fault information complicated to accommodate and gain. Faults can vary both network and agent behavior suddenly. Single-agent fault can effect throughout the communication network. There may occur multiple faults at various time intervals and places. However, such challenging problems can be studied in proper form. Multiple agent composition configuration can give better termination than a single-agent system. Other than this, throughout failure of component or an agent that is not possible to adjust for the single-agent system may be effectively controlled by use of cooperation in between network connection and agents.

Inspired by the above study, practical and theoretical importance is given to review and classify some impressive MAS attacks (Figure 5) and working (Figure 6) to give a complete survey, Figure 10 describes secure and control approaches of DoS attack, and Figure 2 shows present work and key challenges required for study. Furthermore, it is of both theoretical and practical significance to present a study for safety of MASs at one platform and also to provide comprehensive survey of recent advances of resilient consensus for MAS under attacks. To see latest MAS complexity, security problems and system security are highly important. Threats can be physical, cyber, or containing both sides of MAS, and it needs a composite method for mitigation and identification of safety weaknesses. In present research, the purpose was to study weakness, challenges, mitigation schemes, attack types supposing scalability complexity of MASs, and distributive and security and safety challenges. The remainder of the paper is organized as follows: in Section 2, we studied MAS technologies; in Section 3, we studied attack modeling and methodology; in Section 4, we briefly discussed fundamental issues of MASs; in Section 5, we discussed problem formation; in Section 6, there is comprehensive study on cooperative attack methodologies; in Section 7, there is discussion on model-based attack methodology, in Section 8, deception attack detection or identification has been studied; and in Section 9, there are briefly discussed key challenges. Similarly, Figure 2 represents flowchart of paper. Table 1 explains all notations used in the manuscript.

2. MAS Supporting Technologies

Here, we study key technological improvements which are planned as well as links in between MAS and other basics which are studied to encourage technological developing chain that commanded the emergence of MASs. In several works, the concept such as smart object, embedded system, ubiquitous computing, smart environment, and sensor networks plays a key role in development of MASs. In case of time line, embedded systems are ancestors of all given knowledge appeared in the past and beginning fresh development in the field of microelectronics and replying critical problems such as remote control and automation. There is predefined functionality in embedded systems traversing across single or multiplied functions that are not easy to reprogram by the last user. The basic purpose of embedded systems was to control, design, and operate physical world process. Though, in beginning, embedded systems were closely compared to MAS or IoT, they were restricted by physical control function with the cyber space layer.

Figure 3 represents detailed general technologies supporting MASs. With the progress of technology and requirement to manage and control complicated systems, importance of the embedded system was clear and then there was an idea of the network embedded system. An important reason lacking down the availability of MAS was transition from one system to a connected system with more complexity. In view of this concept, sensor networks (SNs) strongly impact on the latest MAS. Actually, for information gathering, the SN contains a number of sensors deployed in specific place/areas. Presence of the SN eased improvements of smart objects, e.g., actuator or sensors containing microprocessor, power sources, and communication services. Thus, the SN cannot be supposed as freely unit, but as part of complicated systems. Fast development in modern computing systems containing MAS gave idea of ubiquitous computing. In addition, concept of computer system integration with daily activities enables them "unseen" for the last users. In some views, ubiquitous computing overlaps with other ideas, ambient intelligence, pervasive computing, and IoT. An important role of IoT is the establishment of capabilities for the last user to enhance present forms of everyday

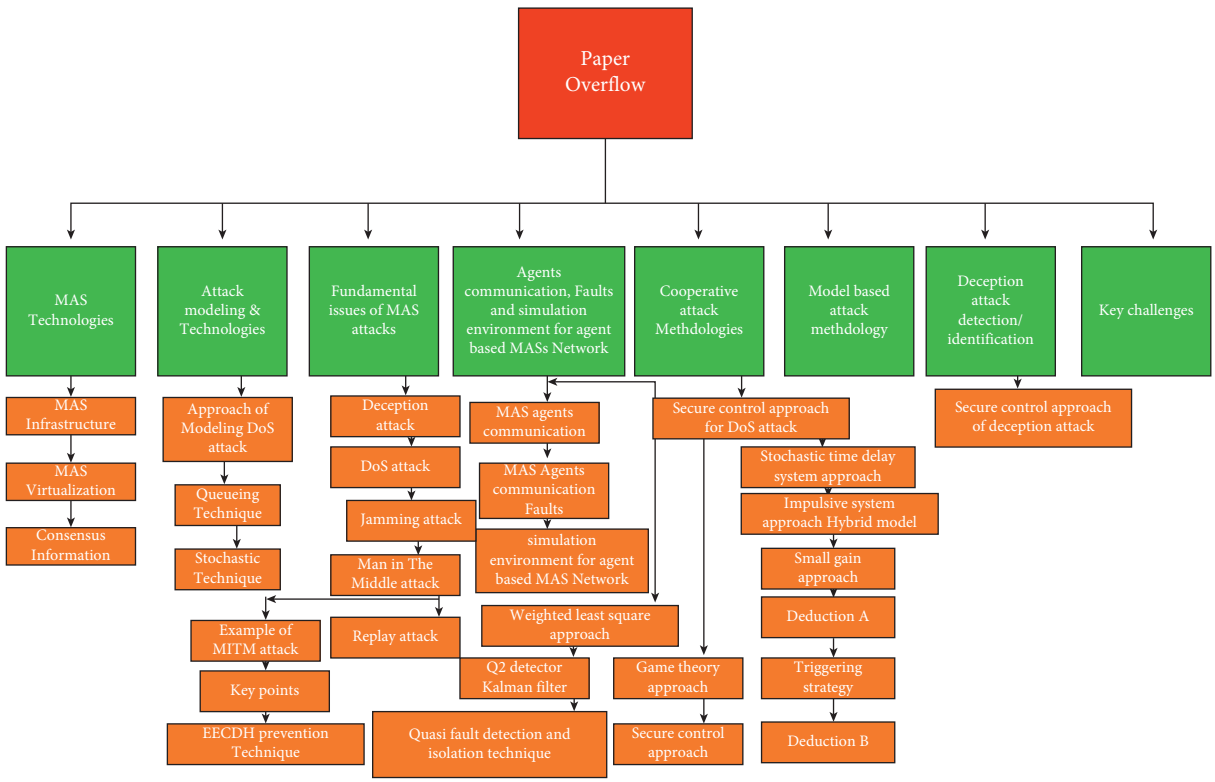


Figure 2: Paper flowchart.

Table 1: Explanation of notations.

Sr. no	Abbreviation	Explanation
1	MAS	Multiagent system
2	MASs	Multiagent systems
3	SN	Sensor network
4	IoT	Internet of Things
5	TCP	Transmission control protocol
6	IP	Internet protocol
7	NICV	Network Interface card virtualization
8	DoS	Denial-of-service
9	PC	Personal computer
10	LTI	Linear time-invariant
11	DDoS	Distributed denial of service
12	MITM	Man-in-the-middle
13	FDI	False data injection
14	PLC	Programmable logic controller
15	SCADA	Supervisory control and data acquisition
16	M2M	Machine to machine
17	SDN	Software define network
18	EECDH	Enhanced Elliptic Curve Differ-Hellman
19	WSN	Wireless sensor network
20	WLS	Weighted least square
21	LQG	Linear-quadratic-Gaussian
22	GT	Game theory

devices and improvement of private facilities by use of joined ubiquitous devices. It shows transition from the embedded system from predefined functionality to IoT and ubiquitous computing where an important need is adaptability and nimbleness. More and more improvements of computing systems run to presences of IoT and

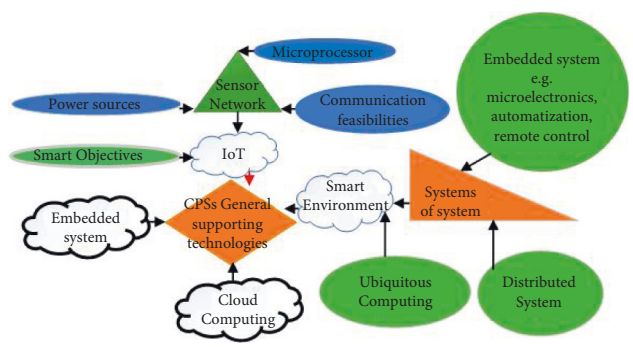


Figure 3: General technologies supporting MASs.

MAS. Actually, MAS is bit far from paradigm in which functional technologies are detached from information technologies to prototype where computational and physical elements are integrated. Although IoT has no specific definition, in some cases, IoT has been described as global infrastructure containing standard protocols and communication technologies that avail facilities given by "things" to diverse applications. In another case, IoT is a general term presenting the scenario types where smart objects are deployed, enabling worldwide communication through Internet or technologies. Hence, the concept of IoT is to represent the infrastructure of the devices worldwide. The scenario where worldwide connectivity is basic need, MAS is supposed to be in the first row of such systems that can join worldwide connectivity and can be distributed among enough units. Figure 4 shows MAS infrastructure.

Figure 4: MAS infrastructure.

2.1. MAS Infrastructure. In development of every system, there is an important role of infrastructure, the same spread on to MAS. In general, basic element of MAS is to be considered as actuator, sensor, communication network, and controller. An important thing is that MAS can be a close-loop or open-loop system. In way of IoT, the global network has access through open loop such as Big Data and cloud computing which are addition in infrastructure of MAS. Hence, MAS may contain a huge number of heterogeneous devices containing actuators, sensors, etc. In addition, it also has roughly limitations on MAS, communication technologies, for instance, and requirement of safety protocols. We can say that such heterogeneity is the main challenge in MAS, so different kinds of devices gain care from a system. To see challenges in big and small systems, among challenges, for example, unified integrity, mobility also affects the entire system. It represents that transferring of devices may cause various problems and need to be taken in regular working of the system. Infrastructure is complicated span because it consists of both hardware and software. MAS architecture is complicated summarizing cyber and physical space. MAS architecture has been studied in [20], contains five-layer MAS infrastructure. In [14], it has been studied four-level MAS architecture. If MAS needs to be in connection with worldwide/global networks, such as Internet, TCP/IP can be the best candidate in which two last layers, physical and data, accordingly are shown by the single level. Transport, network layers, and application form other three layers part. Explained categorization of physical threats in fault estimation, detection, and tolerant of MASs has been studied in Table 2.

2.2. MAS Virtualization. In MAS, the basic purpose of virtualization is to abstract or hide complicated detail such as technical detail from above laying layers and to permit stretchy sharing resource, so that resources or working given as facilitation. So, MAS joints physical and cyber space and contains throughout process from signal to complexity of applications. There are some virtualization techniques such as network virtualization which is divided into subtypes such as router virtualization, network interface card virtualization (NICV), and link virtualization, application virtualization, and device virtualization.

Table 2: Types of fault and work on fault estimation, fault detection, and fault tolerant.

Fault type	Fault estimation	Fault detection	Fault tolerant
Actuator fault	[21]	[22]	[23, 24]
Sensor fault	[25, 26]	[27]	[28, 29]
Actuator and sensor fault	[25, 30]	[31, 32]	[19, 33, 34]

2.3. Consensus Information. Suppose a decision-making agent network with $\dot{x}_i = u_i$ attentive in accessing via local communication with nearby on $G = (V, E)$. By accessing consensus, it seems asymptotically joining to single-dimensional contract space characterized by

$$x_1 = x_2 =, \ldots = x_n. \tag{1}$$

Such contract can be represented as $x = \alpha 1$ in which $1 = (1, \ldots, 1^T)$ and $\alpha \in \mathbb{R}$ is joined decision of all agents in the group. Suppose $A = [a_{ij}]$ is adjacency matrix of G graph. Agent i set in nearby is N_i and represented as

$$N_i = \left\{ j \in V \colon a_{ij} \neq 0 \right\}; V = \{1, \ldots, n\}, \tag{2}$$

where i agent communicate with j agents which is neighbor of i. All set nodes and nearby agents represent E edge set of graph as $E = \left\{ (i, j) \in V \times V \colon a_{ij} \neq 0 \right\}$. $G(t) = (V, E(t))$, and a dynamic graph is that where edges set $E(t)$ and $A(t)$ adjacency matrix varying time. $N_i(t)$ nearby set in the dynamic graph of each agent is well. Such dynamic graphs are helping for explaining the mobile sensor technology network and flocks. The linear system is

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij} \left(x_j(t) - x_i(t) \right). \tag{3}$$

It is a distributed consensus algorithm, and it follows that sum of state of all node is an invariant quantity. When applying this condition at time $t = 0$ and $t = \infty$, we get

$$\alpha = \frac{1}{n} \sum_i x_i(0), \tag{4}$$

while in another way, if there is asymptotically access of consensus, then mandatory cooperative result is equal to initial state average of all nodes. With such variance characterization, the consensus algorithm is known as average consensus algorithm and has several applications in distributed computing on networks such as sensor fusion in the sensor network.

In compact form, system dynamic can be represented as

$$\dot{x} = -Lx, \tag{5}$$

where L = graph Laplacian G and it can be defines as

$$L = D - A, \tag{6}$$

where $D = \mathrm{diag}(d_1, \ldots, d_n)$ is degree matrix of G with zero off-diagonal elements.

L has right eigenvector of 1 linked with zero eigenvalue because of the identity $L_1 = 0$.

3. Attack Modeling and Methodology

3.1. Approach of Modeling DoS Attack. We will study two important techniques for DoS attacks modeling in MASs: one is queueing technique and also another is stochastic technique.

3.1.1. Queueing Technique. Like computers, firewalls and routers are considered networking devices, providing poor performance in supervision of DoS attacks, though trade with maximum rate because of memory resource constrains, interrupt processing, and input output processing and central processing units. However, packet loss and delay jitter are vastly affected under attack which can cause disturbance to the control system performance, e.g., mean squared error, rise and settling, and overshoot of percentage. The transmission of packet in the network control system under DoS attack is approached to smear simple techniques which are based on several input queues [35–37]. Two techniques are discussed as follows:

(1) DoS attack is launched by attackers to an endpoint from the system or PC to surrounding area nearby to endpoint. In this way, a huge number of packets are lost.

(2) DoS attack can be launched by attackers with the use of remote system to initial edged routers foremost to leisurely down network connection between controller and a remote plant.

DoS is considered as singularity that can save control signal from required time preserved. It is done by single host. It shows that control and measurement channel may be provoked individually. Hence, it can be supposed that in the process of DoS attack, it seems complicated to receive or send data. Suppose $\{m_j\}_{\cdot j \in X_0}$ in which $m_0 > 0$, which is DoS 0/1 transition sequence; here, 0 is for "off" and 1 is for "on" situation, e.g., that time when DoS variation a transition from 0 to 1 and that time there is possibility of interruption of communication, so

$$M_j \triangleq \{m_j\} W[m_j, m_j + \rho_j]. \tag{7}$$

M_j is representing time interval of jth DoS attack. That length may be $\rho_j \in \mathrm{R}_{\geq 0}$, this is the time when there is no communication, consider $\rho_n = 0$, and here, jth DoS attack is shown as individual pulse at m_j time. An input is generated by an actuator which is based on fresh received controller data through DoS attack. Given $\rho, \varphi \in \mathrm{R}_{\geq 0}$ with $\varphi \geq \rho$, suppose that

$$\begin{aligned} &\therefore(\rho, \varphi) \triangleq \coprod_{j \in X_0} M_j \prod [\rho, \varphi], \\ &\underline{{}^{\circ}\rho\varphi \triangleq \rho\varphi} \\ &\therefore\rho\varphi. \end{aligned} \tag{8}$$

It represents that at each interval $[\rho, \varphi], \therefore(\rho, \varphi)$, $^{\circ}(\rho, \varphi)$ and $[\rho, \varphi]$ are representing to time instants set, when communication is permitted and stopped, respectively. Applying control signal to all $\rho \in \mathrm{R}_{\geq 0}$, it can be written as

$$\begin{aligned} w(\rho) &= \mathrm{PQ}\left(\rho_{P_{(\rho)}}\right), \\ w(\rho) &= \mathrm{PQ}\left(\rho_{P_{(\rho)}}\right). \end{aligned} \tag{9}$$

It shows that for all $\rho_j \in \mathrm{R}_{\geq 0}.P(\rho)$ modern fruitful control approach, the same to proposed techniques is used in [38]. In concluded form, we can say that there is disadvantage of this start and end of approach is not found and is most useful to post records.

3.1.2. Stochastic Technique. Generally, cyberattacks are performed when the system is weak to detect the threat and results in security defilement. In practice, such attacks are introduced by the series of actions to compare security services such as confidentiality, integrity, and availability of MAS applications such as telecommunication, military, banking, smart power grids, and transportation systems. To trace the threats and cyberattacks such as file-less malware, advance persistent threat, and zero days, CPS has become focus of interest. Other than this, a number of techniques were introduced to predict cyberattacks against MAS, in which several techniques have been developed using a stochastic approach such as Hawkes process model, Markov chain model, negative binomial distribution model, and Poisson model. In the fast-growing trend of CPS attacks, attacks are considered to be launched externally against MASs within a given amount of time and using stochastic distribution models such as Bernoulli model [38, 39] and Markov model [40]; from the LTI system, the Bernoulli model can be seen:

$$\begin{cases} Y(P+1) = B_Y(P) + \mu(P) A_W(P) + u(P), \\ F(P) = \gamma(P) H_Q(P) + h(p). \end{cases} \tag{10}$$

For the Markov model, we supposed the following:

$$\begin{cases} Y(P+1) = B_Q(P) + \mu(\delta(P+1)) A_W(P) + u(P), \\ F(P) = H_Q(P) + h(P). \end{cases} \tag{11}$$

From (6), $h(p)$ = measurement noise and $u(P)$ = process noise.

These measurement noise and process noise are commonly known as independent and identically distributed. Here, with 0 mean Gaussian random vector and covariance Q, $\mu(P), \gamma(P)$ are independent identically distributed. Bernoulli is relevant to existence of DoS attack on measurement and process noises [41].

Now, see (7).

$\mu((P+1)) \in \{0,1\}$ is known as Markov controlled DoS attack sequence which stops transmitting of control signal packets to actuator in which (P) is similar to interior state of attacker [42].

4. Fundamental Issues of Multiagent Systems

Basic purpose of MAS classification is to explain overview and fundamental issues regarding cyberattacks. Some general examples of cyberattacks are distributed denial of service (DDoS), man-in-the-middle (MITM), deception attack, password attack, and malware attack. Cyberattack is an offensive action, while if there is possibility of occurring of attacks, then it is known as cyber threat, while cyber risk is interconnected with the word threat which estimates the probability of proportional loss which may occur. Figure 5 shows basic types and subtypes of security attacks.

Here, we will discuss few cyberattacks and fundamental issues discussed in latest research. Figure 6 shows working of MAS security attacks.

4.1. Deception Attack. There is hastily emerging phenomenon of use of deception technology in contemporary cyber security as a feasible means active and intelligent postbreach defense. Similar to any unruly technology it happens with fallacies. Cyber security needs to be changed from being dependent on largely detecting untrue things within a cloud of healthy activity to being focused on stopping cybercrime, which tends to tempt, phish, deceive, and trap users. Deception tactic often proves to be healthy for defense and attack. Deception technology has progressed far yonder the honeypot perception. Now a day's deception is being active in baiting and luring attackers to a deception environment. Deception, also known as malicious attacks and false data injection (FDI) attacks, is defined and studied in [9, 43–45], e.g., nominated malicious system Stuxnet which is able to be reprogrammed and running code in PLCs in SCADA system cause aberration from required conduct. In power grids, transmission system adversaries can send attack to hack remote terminal units, e.g., in substations, there are sensors [12]. For another example of such kind of attack, see study [14, 46, 47]. Deception changes cyber security by providing sole breadcrumbs and traps for industry specific environment, legacy system, IoT, and devices where low cost regularly excludes security structures. Authors in [48] considered measurement output to encounter deception attack based on Bernoulli distribution during signal transmission. To describe random property of deception attack, Bernoulli distribution has been deployed [14]. Deception attack has been used in the term of limited time boundness [14].

4.2. DoS Attack. Denial-of-service (DoS) attack denies or makes slow to the authentic users to access a resource web, e.g., emails and network. DoS attack is policies, and those are usually used for profession of communication capitals in order to forbid the measurement transmission and cause supreme possible worsening of performance of the system. The common DoS model has been studied in [49] in which DoS topographies are discussed with DoS duration and DoS frequency. Similarly, improvement in this idea has been studied for production with the output controller of dynamic feedback.

Complicated form of DoS is distributed denial of service (DDoS) [50, 51] which is also known as coordinated attack, where a huge number of cooperated machines work to achieve DoS attack [52]. However, because it can be easily created, so DDoS is easily available, has high impact and low cost of systems, consisting ability of fully detach an association [53]. It is represented that there is instability of power grids because of attacks and could give long delay jitter on network control system packets. The division of DoS attack in radio frequency identification is because of the reasons studied in [54, 55], e.g., desynchronization attack, system jamming, kill command attack, tag data modification, and random DoS attack as shown in Figure 5.

In conclusion of this DoS attack, aforementioned forms of DoS attacks are implemented for the classification of DoS attack in radio frequency identification systems as studied in [29, 54, 56]. Therefore, they could be present in many forms of MASs.

4.3. Jamming Attack. Such DoS attack refers to condition when one channel is occupied by an attacker for prevention of other node from its use which causes blocking of communication. For obtaining optimal defense mechanism for the network control system, stochastic game theory is applied [29, 57, 58]. Dynamic collaboration among attackers and sensor transmitters in the network control system was projected as the double-player stochastic game. In stochastic game, cost functions contain source cost used for conduction of attack actions, cyber-layer defense, and as possible harmed dynamic act of the network control system. Interaction effect between defender and attacker on dynamic concert of the network control system was supposed by the following cost function. Finally, a stochastic dynamic programming delinquent has been explained for gaining optimal defense mechanism.

In [59–61], security in remote state approximation of MASs has been studied. Communication between remote approximator and sensor node was taken through wireless channel that may be attacked by a jamming attacker. Best decision of process making of both attacking and communicating was discussed in case of consideration of energy constrains for both attacker and the sensor. Markov theory was used for gaining equivalent solutions, and constrained relax delinquent was designed.

For maximization of linear quadratic Gaussian used optimal jamming attack, it controls cost function while supposed energy constraints studied in [59, 60, 62]. Corresponding cost and optimal jamming agenda were consequent after studying the usage of cost function under a free attack agenda. The fresh analytically model was studied in the influence of attack jamming on broadcasting. A jamming attack for optimal energy efficient by wireless channel under jammer attacker energy constraints is studied in [63].

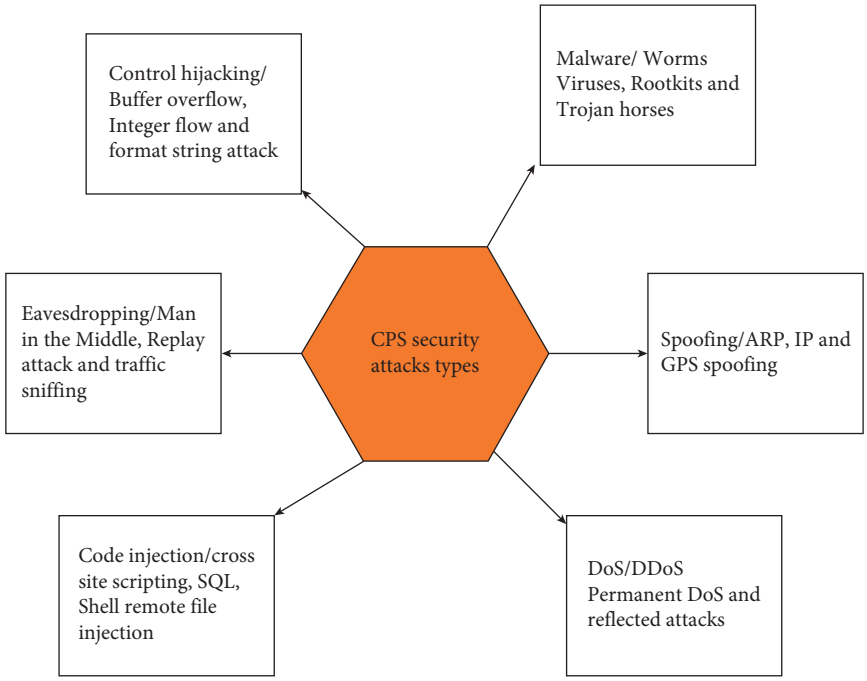


Figure 5: Types/subtypes of MAS security attacks.

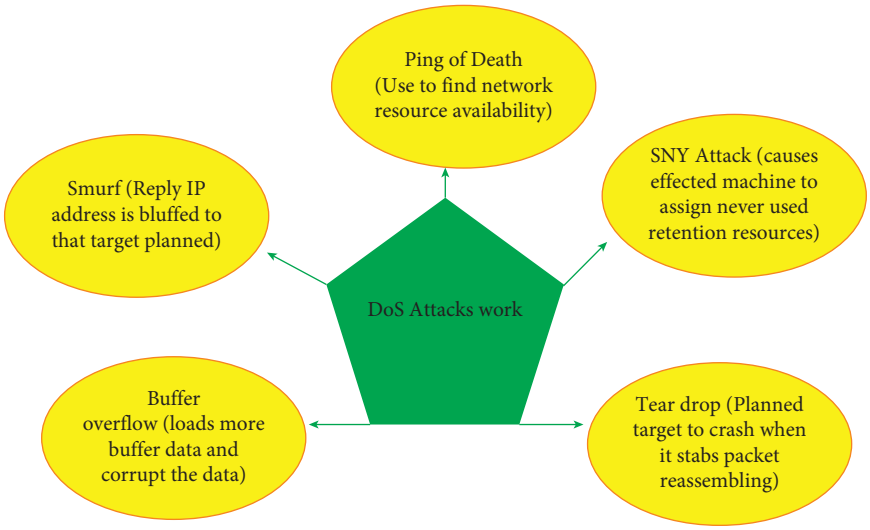


Figure 6: Working of DoS attacks.

These attacks forced by power constrained pulse width-modulated jammers are supposed to be moderately recognized, which is jammer period and unchanging inferior destined and jammers asleep periods are identified. Controller synthesis problem that is an event based for network control systems and strong event triggered communication scheme was studied in [27]. In conclusion, piecewise Lyapunov function is applied to guarantee exponential stability of the system.

4.4. Man-in-the-Middle (MITM) Attack. Generally name "man in the middle" is derived from a game scenario known as basketball, where two players aim to throw a ball to each other, and one of them tries to clutch it [64]. MITM is well-known computer security attack, which gives a great challenge to security professional. Actually, it hits real data flowing between confidentiality, endpoint, and integrity of the data. For analyzing and categorizing the MITM scope, researchers should read [65] survey.

Basically, MITM takes benefit of authentication protocol weakness used by communication networks. Usually, the third party is responsible for authentication that issues certificates because the system of certificate production becomes another way of solid weakness. MITM attacks permits unauthorized parties to snoop data by backdoor.

4.4.1. Example of MITM Attack. About in 2015 [66], it was known that Lenovo machines are available with already installed adware known as superfish which injects adds on browser such as Internet Explorer and Google Chrome. Superfish installs a compiled certificate into Windows certificate database. It leaves all secure stock layer (SSL) certificates given by Hypertext transfer protocol secure (HTTPS) that are linked with personal certificates. Such situation may provide an access to hackers to receive sensitive information such as bank account information, transactions, or user lifestyle..

Internet of Things (IoT) is progressing from smart home to smart cities and making our lives dependent. With the passage of time, billions of M2M will be interconnect with each other, and the big problem is to manage such a big problem for network administrator. Intensive-security methods, classical computing method as antivirus, and encryption are not directly installed software. With network infrastructure, it is compulsory to make IoT devices more secure.

Opposite to the traditional security network, software define network (SDN) [63] gives several new features, as centralized control and network programmer, which skilled the owner to manage network automatic in a dynamic and flexible way. We can see that IoT future [67] is SDN dependent. Open floe channels security issues of IoT, e.g., MITM, are studied in [68].

Distributed methodology for agent network permission targeted for execution of the distributed algorithm to control MITM attack is studied in [69, 70] which intends steering algorithm result towards erratic values of risky configurations. An example of MITM attack is shown in Figure 7.

Figure 7(a) shows victims without attackers, and Figure 7(b) shows victims with attacker, contenting message between A and B without notice.

4.4.2. Key Points. There are following key points of MITM:

(1) MITM permits hackers to intercept confidential data.

(2) MITM is session hijacking type.

(3) MITM permits hackers to insert malicious data and sites in form misty from genuine data.

(4) MITM exploits real-time nature of data transfer and communication to go hidden.

(5) MITM contains hackers inserting themselves as proxies or relays in an ongoing data transfer or appropriate conversation.

Researchers can study the following literature for their research interest of MITM attack [71–73].

4.4.3. EECDH Prevention Technique. Enhanced Elliptic Curve Differ-Hellman (EECDH) prevention technique for MITM attack is well studied in current research [45] which improves the security level. Keep secure MITM attack where communication carrier clears themselves before cooperating their keys, to use Differ-Hellman key exchange for communal verification, so that during cloud sharing, data privacy sustained.

4.5. Replay Attack. Such kind of deception attack happened when adversary succeed in recording some of the transmission data, e.g., in MAS, sensing data are injected [74, 75]. This form of attack is supposed to happen in two ways, e.g., in 1st way of Figure 7 recording data of attacker from system and injecting the same data in the system, and another way is attack could be outside carrying that subject to physical systems represented in Figure 8 [76]. Likewise, an attacker formed communication connection in between two last points to enclosure observed messages in various areas in globe generally present in WSN [77]. In designing, such attack could be assumed as changeable delays with unidentified data on variable rates and upper bounds. Applying time-delay system concept jointly by optimization methods acceptable max upper bound can be premeditated [3].

There is no requirement of system information in such form of attack, containing information on designed estimator and controllers; for detection, this activity makes it complicated. Adopting counters and time-stamp in the transmitted data is solution opposite to such attack. Two phases, first and second of replay attack, has been shown in 8.

There is not enough research that studies controlling of MASs subject to replay attacks, e.g., recording vista control variation direct to replay attack is discussed in [78], which gives an explicit and simple connection in between computing, attacking horizons, and infinite-horizon cost. Then, asymptotic exponential stability of the system is ensured by availing enough condition set; see one more example study [74, 79]. Feasibility terms of replay attack and countermeasure suggestions those enhance the possibility of detection by supposing control performance are discussed in [80]. And integrity attack on the control system is countermeasured and analyzed proficient of showing these attacks were not assumed. For further example of such attack, we can see [79, 81, 82].

5. Agents Communication, Faults, and Simulation Environment for Agent-Based MAS Network

In this section, we will study MAS agent communication, faults, and simulation environment.

5.1. MAS Agent Communication. We know that MAS contains self-directed agent group which works in cooperation with each other through other communication medium to gain considered goals, and find number of usage in different areas such as physics, biology, mathematics, social science, and computer engineering.

Since 1962 [83], agent's communication has been studied. Mainly used communication approaches are message passing, speech act, and blackboard. In message passing, agents directly message each other as shown in Figure 9(a). There is use of broadcast or point to point agent

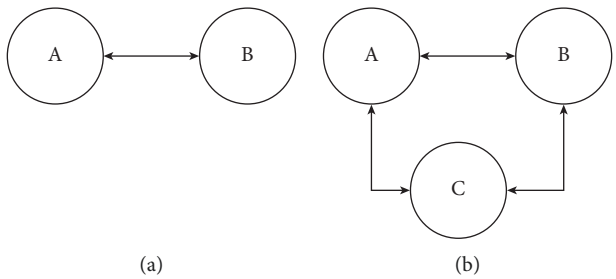


Figure 7: Victim with and without attack.

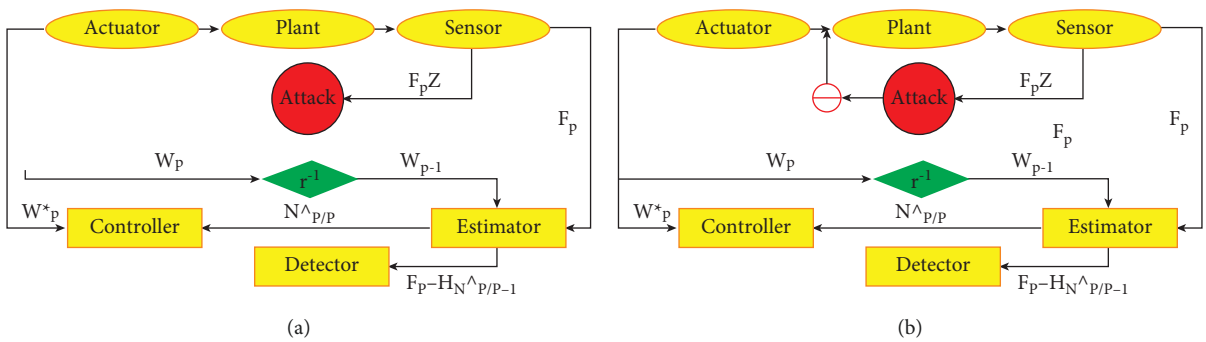


Figure 8: First and second phase of replay attack.

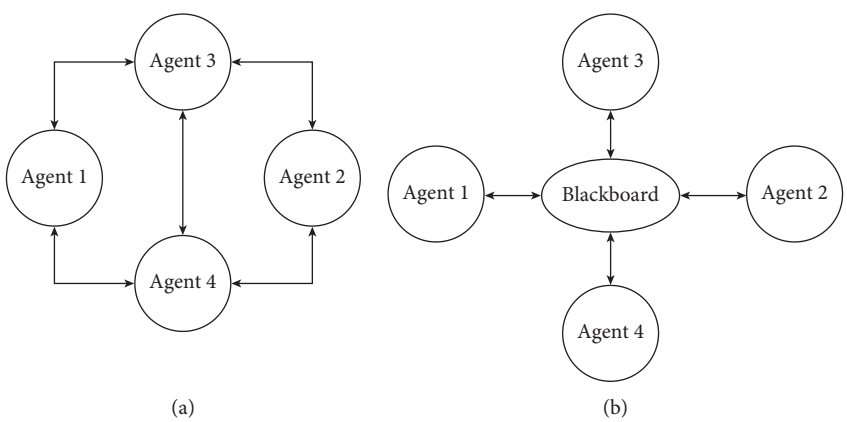


Figure 9: An overview on communication approaches in MAS.

communication to communicate with other agents. In the broadcast communication, one agent sends message to all nearby agents while in former agent one can directly talk to another agent in case of other agent address information.

In speech act, researchers in [84] studied that some sentences or utterance verbs are act as speech acts that vary physical environment such as in general environment if general person uses sentence that "I now make you man and wife" such kind of sentence have impact on physical environment by introducing new condition and rule. Agent action can be as a speaker, which produces utterance to vary listener belief [84, 85].

In blackboard communication, agents share data with each other in collaboration by use of central repository known as blackboard, Figure 9(b). In this, data of each agent are stored in blackboard which are accessible and readable by other agents. Blackboard uses control information for controlling agent's access. It is important for message semantics that need to confirm communicating agents with each other who have the same understanding of exchanged data. Sometimes in heterogeneous agents, it can be a challenging task.

5.2. MAS Agent Communication Faults. Basically, for MAS, two kinds of faults are considered known as network fault and agent fault. Network fault occurs in the communication network and may disturb communication performance while agent fault that can occur in internal components of actuators, agents, and sensors. There are

three types of classification of agent's faults, sensors faults, agents fault, and actuator fault which may affect agent dynamic.

5.3. Simulation Environment for Agent-Based MAS Network. Here, we are studying various evaluations and modeling methods used for metrics performance that differ depends upon task and MAS application of the considered agent-based system in comparison with state of the art. There are three basic evaluation methods, MATLAB, Java agent development framework, and GAMA.

Using MATLAB, we study MAS performance, especially with mathematical complex environment. In addition, it is adjustable to Java agent development framework for more work on MAS performance.

Java agent development framework is mostly used among simulators in MAS. Its admiration stalks from following properties. It benefits from third-party libraries and also is Java based. It is also written on foundation of intelligent physical agent's standard. For designing MAS, it has graphical interface. It supports simulation distributed systems, is open source, and can link to Matlab, and also it skins complexity of MAS.

Third, GAMA is simulation and modeling platform for agent-based system development. There are some advantages of GAMA such as it supports widely level MAS that contains a huge number of agents, it is useful for simulation purpose of any kind of MAS application, and it supports intuitive agent-based language such as GAML.

Studied simulation methods are specifically for MAS, while because of large-level usage, specific evaluation methods are used for system analyzing in particular application and can be deployed for agent-based system simulation performance to see issues in that application.

This is an important method which is usually used in the deception attack in sensor networks which contains the hypothesis test with predefined probabilities of binary hypothesis [86, 87]. Cooperative spectrum sensing performance limit is evaluated subject to Byzantine attacks; however, false data affect the fusion center, because of that output of wrong sensor increased [88, 89]. On binary hypothesis, a similar ratio detector is considered to manage with already determined fixed error for security of smart grid in sensor network data [20, 87, 90], energy frame work that is deep learning based, and block chain based is well studied in [20, 89]. A detector based on progressed similar has been studied in [91–93] and supposed for unobservable and observable circumstances in the SCADA system. An example of such methods has been applied in [94]. Important hypothesis applied is that all node transmitted packets are arbitrary; by this way, probability of next packet will not affect in verdict a packet to be nasty. In conclusion, there can be several forms of attacks those can affect single or several packets. For calculation of trust values, it is to be considered that node is transferring "X" packets; here, j packets are supposed to be normal. Observation of $x(X) = j$ distribution is studied by given binomial distribution:

$$K(x(X) = j|K) = \binom{X}{j} K^j (1-k)^{X-j}. \tag{12}$$

Here, K shows that i^{th} packet probability is normal k(K) = no. of normal packets this model intends to guess the probability of $K(W_{X+1} = 1|x(X) = j)$ and result out either the $X + 1$ packet is in normal form. With the use of Bayesian theorem, the following probability distribution is calculated:

$$\begin{aligned}(K)&\left(W_{X+1} = \frac{1}{x(X)}, = j\right)\\ &= K\left(W_{X+1} = 1x(X) = j\right)K(x(N) = j.\end{aligned} \tag{13}$$

Here, we can apply marginal probability distribution:

$$\begin{aligned}k(x(X) = j) &= \int_0^1 k(x(X) = p|k)g(k).vk,\\ K\left(W_{X+1} = 1, x(X) = j\right) &= \int_0^1 K(x(X) = j|k)g(k)k.vk.\end{aligned} \tag{14}$$

There are no data for k; now, it is considered that it can be found by uniform prior distribution $g(k) = 1$; here, $k \in [0, 1]$. Hence, we can rewrite above equations (12)–(14) as

$$\begin{aligned}K\left(W_{X+1} = 1, x(X) = j\right) &= \frac{\int_0^1 K(x(X) = j|k)g(k)k.vk}{\int_0^1 k(x(X) = p|k)g(k).vk},\\ &= \frac{p+1}{X+2}.\end{aligned} \tag{15}$$

In resultant from equation (15), both of normal packets number j and X which is whole packets can be found in WSN, after the collection of traffic information. By applying suitable threshold, we can find malicious node. Some numerical results of such malicious nodes has been studied in [92, 95]. With the use of this model, we can find malicious node.

5.4. Weighted Least Square (WLS) Approach. It is an effectual consistent attack detection method for dimension data. Mostly, it is applied in power systems and smart grids [96–99]. By comparison of predefined threshold and constructed measurement residual, we carried out a bad resultant.

Suppose

$$r = MY + c. \tag{16}$$

Here, $M = [m_{\text{ab}}]_{i*j}$ is known as measurement Jacobian matrix with full column rank, when $j > x$, m and r are considered states vector and measurement, respectively, and c is system noise effecting. Estimated delinquent is used to solve the m^* of variable m, which is better for measurement of meter r w.r.t equation (10). Estimated measurement r^* and observed measurement have a difference that is defined

as $z = r - r^* = r - MY^*$. WLS problem is to find an estimate m^* which slow the index performance $D(Y^*)$, which can be found by the given formula:

$$\min_{y*} D(Y^*) := (r - MY^*)^Z U(r - MY^*). \quad (17)$$

In this weight matrix, $U := \sum -1$.

To simulate 1st order optimal situation, $D(Y^*)$ is studied in [43]:

$$y^*\left(M^Z UM := Cr\right), \quad (18)$$

where "C" is pseudoinverse of M and $CM = 1$.

5.5. Q^2 Detector Kalman Filters. Here, we used characteristics of Kalman filter residual instead of WLS, to make it feasible for good or bad data:

$$\begin{aligned} Q_{p+1} &= B_{Q_p} + A_{b_p} + u_p, \\ F_p &= H_{Q_p} + h_p. \end{aligned} \quad (19)$$

In this, $b_p \in R^k, Q_p \in R^k$ and $F_p \in R^j$ are the control input, state variable, and system measurements, respectively; $h_p \approx \nabla(0, R)$and $u_p \in R^k$ are measurement noise and process noise, respectively. We can calculate $Q^*_{p/p}$ with the use of given Kalman filter:

$$\begin{aligned} \mathcal{N}^*_{0|-1} &= N^*_0, \\ K_{0|-1} &= \varepsilon, \\ Q^*_{p+1} &= B_{Q_p} + A_{b_p}, \\ K_{p+1|p} &= BK_pB^Z + S, \\ P_p &= K_{p|p-1}H^Z + R\big)^{-1}, \\ Q^*_p &= K_{p|p-1} + P_p\left(F_p - HQ^*_{p|p-1}\right), \\ K_p &= K_{p|p-1} - P_pHK_{p|p-1}. \end{aligned} \quad (20)$$

It has been known that Kalman filter exists in $F_i - HQ^*_{i|i-1}$ for equation (19), with LQG controller, and Kalman filter is Gaussian independent identically distributed [22]. Suppose

$$e_p := \sum_{i=p-\mathrm{T}+1}^{p} \left(F_i - \mathrm{C}Q^*_{i|i-1}\right)^Z \sigma^{-1}\left(F_{i-}\mathrm{C}Q^*_{i|i-1}\right). \quad (21)$$

Here, e_p = has an Q^2 distribution among jT independent degree in normal operation which shows the lower probability of greater e_p. T is window size. Here,

$$e_p \underset{>M_1}{\overset{<M_0}{}} \rho. \quad (22)$$

where M_0 represents null hypothesis, M_1 represents under attack hypothesis, ρ represents threshold discussed in [100, 101] used for the SCADA system, and the same results have been shown in [102]. Q^2 detector with cosine has been explained in [95, 103, 104] for the detection of false data injection attacks which affects smart grid. An algorithm is discussed for the detection of deception attack in an application which could be remote state application, smart sensors used for data receiving [105]. Second application of such kind has been studied for the detection of bias injection attacks for stochastic rectilinear dynamical scheme [12, 106–109]. Multiclass support vector machine was discussed for building an intrusion detection model.

5.6. Quasi Fault Detection and Isolation Techniques (FDI). FDI is famous and widely used in the networked control system. It identifies the fault presence, location, and fault type because it contains the monitoring system. This technique is helpful in detection of exterior attack in MASs. Attacks were supposed as unidentified inputs which effects both of states and measurements [110–113]. Using graph theory, undetectable attacks were characterized and also with the use of distributed and centralized monitor have been planned for the detection of distinguishing attacks. Based on geometric approach, a fault detection method has been implemented for detection of cyberattacks and fault in power systems or networks [114, 115]. The common system was applied for the detection of deception attacks and sensor actuator [44, 114]. Likewise, a model diagnosis system and free-fall detection were studied for designing cyberattack detector for the distribution system of water [26]. For detection and differentiation of both cyberattacks and fault, an intelligent generalized predictive controller was intended [25, 116]. Cyberattack can be targeted by recognized weakness in the system, which is weak point of FDI, different from losers commonly arbitrary or random. For designing the required robust system, this technique needs cautious investigation in order. In conclusion, it is important because of highly study, which lot of research has been done on for detection of cyber security in power systems, e.g., [117–120].

5.6.1. Argument. Bayesian detection with binary hypothesis has been broadly studied and pragmatic in sensor network data fusion [23, 121,122]. In meaning of state approximation, there is need of system noise in a stochastic framework for the attack detection method, because it allows a probabilistic state approximation [64, 123]. Hence, there is need to smear mean and variance for state disruption distribution shown as freely variables. Providing of confident state approximation is important in many present applications, e.g., system guidance and navigation, target tracking, and attack [64, 123]. In short, modeling the state distributions in some sets supposed to be unidentified, but limited noises are more suitable. Therefore, $\aleph^2$ is mostly useful based on holding discrepancies in approximation conduct which forecast by a model. Resultant unidentified but limited sounds are suboptimal, and attack detection feasibility is reduced.

6. Cooperative Attack Methodologies

Comparing to other kinds of attacks, e.g., DoS attack and replay attack, we got no much more interaction of researchers. This is because of phenomenon that fixed on controlling of MASs. Both defending and detection against DoS attacks in a state approximation delinquent were

studied in [124, 125]. Data of sensor are transferred to the estimator by a packet, falling communication network. Already defined Kalman filtering [126, 127] for approximation of state for the untrustworthy communication system is pragmatic in this network. First of all, hypothesis testing detection problem is articulated, while supposing already known knowledge of network statics. Secondly, there were considered two preventing policies with the use of secure coding packet slant for recompensing the absence data, and another is improved on transmission power up gradation to control the blocking upshot of attack.

In [126, 127], game theory approach has been studied, e.g., collaboration between attacker and sensor is designed as zero sum stochastic game, which finger to DoS attack in remote state approximation. Presence of Nash equilibrium was primarily studied for such kind of game, and later on the best policies were planned for fixing sensor transmission power. For calculation of asymptotic performance of remote approximator, planned form game is applied in [128, 129].

For nonlinear chaotic systems, sliding mode control with actuator fault and decentralized sliding mode for heterogeneous MASs problem of fault tolerant control considering both DoS and network fault is discussed in [25, 131]. It is to be supposed that network fault contains of deterioration, signal attenuation, and perturbations of couplings those are in nonlinear form. Reimbursement of perturbed couplings and faulted were gained to apply a strategy which is known as slide mode planned strategy; by way of unidentified constraints of approximation, then the mathematical analysis method and Lyapunov stability theory were applied to assurance the asymptotic management of the nonlinear confused system. We see another example of approximation of delinquent in MASs exposed to DoS attack studied in [7, 130–132].

6.1. Secure Control Approaches for DoS Attack. For controlling MASs exposed to DoS attack, many researchers worked; those are shown in Figure 10.

6.1.1. Stochastic Time-Delay System Approach. Here, DoS is designed as stochastic process with signal delay. Deception and DoS attacks are supposed to be freely stirring and designed as Bernoulli distributed white sequences in [12, 131]. Suppose the discrete time stochastic system with measurements and noise effecting the system, as

$$\begin{cases} Q_{P+1} = \left(B_0 + \prod_{i=1}^{z} \tau_{a,p} B_a \right) Q_P + \mathrm{BW}_P \Big), \\ F_P = \left(H_0 + \prod_{a=1}^{r} \tau_{i,P} H_a \right) Q_P. \end{cases} \tag{23}$$

Here, $F_P \in R^{x_f}$ is measurement of sensor and $Q_{P+1} \in R^{x_m}$ is state vector, and $W_P \in R^{x_w}$ is input of controller. $B_a\,(a = 0, 1, \ldots, z)$, A and $H_a\,(a = 0, 1, 2, \ldots, l)$ are with appropriate dimension and constant matrices. $\tau_{a,p} \in R\,(a = 1, 2, \ldots, z)$ and $\tau^*_{a,p} \in R\,(a = 0, 1, 2, \ldots, l)$ are multiplicative noises with unity variance and zero means, and jointly uncorrelated with P and a, z and l those are positive integer, A rank is considered to be x_w, and to study this kind of problem, we see the given model of attack:

$$F^{\varphi}_{P_r} = \left(\mu^{\cong\varphi}_{p_r} + \partial^{\varphi}_{P_r} h^{\varphi}_{P_r} \right) + \left(1 - \mu^{\varphi}_{P_r} \right) F^{\varphi}_{P_{r-1}}, \tag{24}$$

where $F^{\varphi}_{P_r}$ = data received by controller and $h^{\varphi}_{P_r} \in R^{x_f}$ = stands for attackers injected signals as in (23):

$$h^{\varphi}_{P_r} = -F^{\cong\varphi}_{P_r} + {}^{\varphi}_{p_r}, \tag{25}$$

where ${}^{\varphi}_{p_r}$ is freely fixed signal which satisfies

$$\begin{aligned} &\left\| \delta^{\varphi}_{P_r} \right\| \le \mathrm{epsilon}_2, \\ &\mathrm{Prob}\left\{ \mu_{P_r} = 0 \right\} = 1 - \widehat{\mu}, \\ &\mathrm{Prob}\left\{ \mu_{P_r} = 1 \right\} = \widehat{\mu}, \\ &\mathrm{Prob}\left\{ \partial_{P_r} = 0 \right\} = 1 - \widehat{\partial}, \\ &\mathrm{Prob}\left\{ \partial_{P_r} = 1 \right\} = \widehat{\partial}. \end{aligned} \tag{26}$$

μ_{P_r} and ∂_{P_r} = white sequences Bernoulli distributed with 0 and 1 value. Probability is given in (26). In (26),$\partial_{\cdot} \in [(0, 1)]$and $\partial \in (0, 1)$ are identified constants. Some enough states are gained to confirm the security needs of the system in which we gained some enough conditions.

6.1.2. Impulsive System Approach Hybrid Model. Here, we showed a system that is beneath DoS attack showed by the impulsive system. Resilient control techniques and resources aware are designed with malicious DoS attack studied in [133–135]. Specifically, an event-based control scheme which is output that is output based was pragmatic to control to get the communication strategy and control in lass of nonlinear feedback systems which is provoked by exogenous troubles.

Suppose

$$p: \begin{cases} \widehat{Q}_K = g_K\left(Q_K, W, U\right), \\ F = f_r\left(Q_K\right), \end{cases} \tag{27}$$

where p = plant. Here, $U \in R^{x_u}$ = uproar input, $F^* \in R^{x_f}$ = state vector, $W \in R^{x_W}$ = control input, and$F \in R^{x_f}$ output of plant:

$$E: \begin{cases} \widehat{Q}_q = g_q\left(Q_q, \widehat{F}\right), \\ W = f_K\left(Q_q, \widehat{F}\right). \end{cases} \tag{28}$$

In this, $Y_c \in R^{x_q}$ = controller state, $F^* \in R^{x_f}$ = fresh gained measured output, and $W \in R^{x_w}$ = controller output; resultant output is $r = e(Q)$ in which $r \in R^{x_r}$ and $Q = (Q_k, Q_q)$. Here, attack is DoS and interval of attack is represented by $\{M_x\}_{x \in X} \in I_{\mathrm{Dos}}$; at this time period, there is no communication between controller and sensor since attack. DoS attack collection time is given as

$$\varnothing \triangleq \sum_{x \in X} M_x. \tag{29}$$

To apply framework, hybrid model F^* updating can be

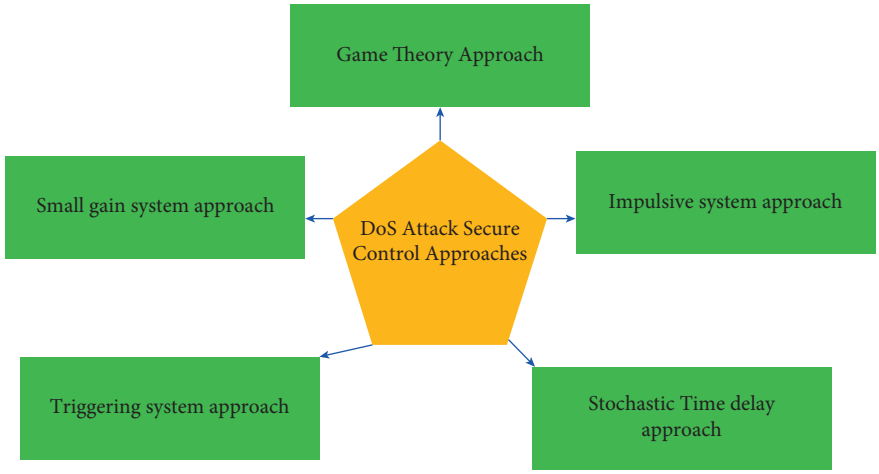


Figure 10: Secure control approaches of DoS attack.

$$\widehat{F*} = \begin{cases} F, \text{ When } \rho_b \nexists \varnothing, \\ \widehat{F}, \text{When } \rho_b \exists \varnothing. \end{cases} \tag{30}$$

Transmission error $c \triangleq F^* - f$ can be

$$c^+ = \begin{cases} F, \text{ When } \rho_b \nexists \varnothing, \\ \widehat{F}, \text{When } \rho_b \exists \varnothing. \end{cases}$$

$$\rho_{\text{miet}} = \begin{cases} \dfrac{1}{L_r}\left(\dfrac{r(1-\gamma)}{2(0/0)((x/l)-1)+1+0}\right), \gamma > L, \\ \dfrac{1}{L}\dfrac{1-\gamma}{1+\gamma}, \gamma = L, \\ \dfrac{1}{L_r}\arctan h\left(\dfrac{z(1-\gamma)}{2(\gamma/\gamma+1)((\aleph/L)+1+\gamma)}\right), \gamma < L. \end{cases} \tag{31}$$

Equation (31) represents maximum allowed transmission interval limited ρ_{miet} that is characterized, where $z = |(\gamma/L)^2 - 1|, L \geq 0$ is fixed, $\gamma \in (0,1)$ shows present information in local area at a mechanism known as event triggered mechanism, and we obtained γ as

$$\langle \nabla h(Q), g(g,c,u) \leq -\sigma(\|Q\|) - \sigma(\|F\|) - M^2(Q,W) - \epsilon_1(U(c)) + \sigma^2 U^2(c) + \vartheta^2 \|U^2\| \rangle. \tag{32}$$

This condition explanation is well studied in [135]. At the end, to suppose DoS attack in normal form, these DoS attacks are banned in form of duration and frequency.

6.1.3. Small Gain Approach. Distribution system stabilization problem exposed to DoS frequency characterization, DoS attack, and preserved stability duration is studied in [135]. To preserve communication resources, a hybrid communication technique is also supposed. By using of hybrid transmission technique, zero behavior can be saved and load communication can be compact efficiently. e.g., a large-scale system contains X interacting subsystem is supposed with given model:

$$\aleph_a^*(\varphi) = B_a Q_a(\varphi) + A_a Q_a(\varphi) + \coprod_{b \in X_a} M_{\text{ab}} Q_b(\varphi). \tag{33}$$

In this, B_a, A_a and M_{ab} are with suitable breadth. $Q_a(\varphi)$ and $W_a(\varphi), \varphi \in R_{>0}$ are control and state input of the subsystem. Input control applied to "a" subsystem is

$$W_a(\varphi) = P_a Q_a \varphi_P^i + \coprod_{b \in X_i} L_{\text{ab}} Q_b \varphi_p^b, \tag{34}$$

where L_{ab} = controller coupling gain.

Suppose $\{m_x\} x \in X_0, m_0 \geq 0$ representing off/on DoS transmission, e.g., DoS displays time instant transmission from 0 to 1. Hence,

$$M_x \triangleq \{m_x\} \cup [m_x, m_x + \rho_x]. \tag{35}$$

Equation (35) shows x^{th} DoS time instant. ρ_x is length on which there is DoS attack on the network; suppose

$$\therefore(\rho,\varphi) \triangleq \sum_{x \in X_0} M_x \sum(\rho,\varphi). \tag{36}$$

This is subclass of (ρ, φ), and also there is DoS attack on the network.

(1) Hypothesis A. Equation (36) is supposed to be constant in DoS frequency, in this $\tau \in R_{>0}$ and $\rho_O \in R_{>0}$, as

$$x(\rho,\varphi) \leq \tau + \frac{\varphi - \rho}{\rho_O}. \tag{37}$$

(2) Hypothesis B. $p \in R_{\geq 0}$ and $Z \in R_{>1}$ are present as constant in DoS duration:

$$|\therefore(\rho,\varphi)| \leq p + \frac{\rho - \varphi}{Z}. \tag{38}$$

(3) Hypothesis C. When there is no DoS attack, an intersampling interval ∇ is present, e.g.,

$$\|c_a(\varphi)P\| \le \epsilon_a \|Q_a(\varphi)\|. \tag{39}$$

ϵ_a is appropriate design constraint.

6.1.4. Deduction A. Equation (33) is representing the distributed system, and equation (34) is for control input, so for this distributed system and control input communication of a plant controller on collective network with Hypothesis C, and sampling ∇ interval. For any DoS attack, the large-scale system is asymptotically constant. Hypothesis A and B with freely τ and p and ρ_O and Z are as follows:

$$\begin{gathered}\frac{1}{Z}+\frac{\nabla_*}{\rho_O}<\frac{\tau_1}{\tau_1+\tau_2},\\ \|\mathscr{L}_1(P)\|+\|\mathscr{L}_2(P)\|\le\mathscr{L}.\end{gathered} \tag{40}$$

And its subsets are discussed, and second deduction detail is discussed in [136].

6.1.5. Triggering Strategy. Equation (33) is representing the distributed system and equation (34) is for control input, so for this distributed system and control input communication of a plant controller on collective network with Hypothesis C, and sampling ∇ interval. For any DoS attack, the large-scale system is asymptotically constant. Hypothesis A and B with freely τ and p and ρ_O and Z are as follows:

$$\frac{1}{Z}+\frac{\nabla_*}{\rho_O}<\frac{\tau_1}{\tau_1+\tau_2}. \tag{41}$$

We supposed a plant jammer-operator, in which communication between plant and operator is effected by jammer studied in [137]. For reduction of the system communication system, an event triggered time order was assumed. Suppose $W \in R^j$ and $Q \in R^x$ be input and state vector, respectively. Given system is to be supposed

$$\begin{aligned}\aleph^*(\varphi) &= B_Q(\varphi)+A_W(\varphi),\\ w(\varphi) &= PQ(\varphi_P), \quad \forall_\varphi \in \left[\varphi_p, \varphi_{P+1}\right],\end{aligned} \tag{42}$$

where B, A, and P, are proper dimensions matrices, and $\{\varphi_P\}_{p\ge1}$ = triggering time sequence. Now, suppose $c(\varphi) = Q(\varphi_P) - Q(\varphi)$, For all $\in \lfloor \varphi_{P,}\varphi_{P+1} \rfloor$, system is stable, if function $W(Q) = Q^Z K_P$ connected with $\lfloor E \rfloor > 1$ is supposed the control $W(\varphi)$ which at time Z_P updated, to see given triggering law

$$|c(\varphi_P)|^2 = \in \frac{\lfloor E\rfloor - 1}{\lfloor \mathrm{KAP}\rfloor.^2}|Q(\varphi_K)|^2, \quad P \ge 1. \tag{43}$$

This law time sequence is

$$\varphi_{p,x} = \left\{\varphi_1 \text{ satisfying above equation } \varphi_1 \in \langle (x-1)Z,\ (x-1)Z + Z_0^1\rangle\right\} \cup \{xZ\}. \tag{44}$$

Here, equation (42) is with asymptomatically stable, and equation (44) is triggering law.

6.1.6. Deduction B. See equations (42) and (44) if assumed conditions mollifies. It is studied in [137]:

$$\begin{gathered}\frac{(1-\epsilon)Z_1^0(|E|)-1}{2} > |K|\log(\mu),\\ \mu := \exp\left(\left(Z - Z_0^1\right)\alpha(B+\mathrm{AP})\right)\\ + \frac{\mathrm{AP}}{\alpha(A+\mathrm{AP})} * \left(\left|\frac{\mathrm{AK}}{B}+1\right|\right)\left\{1-\exp\left(\left(Z-Z_0^1|A|\right)\right) * \left(1-\exp\left(\left(Z-Z_0^1\right)\alpha(B+\mathrm{AP})\right)\right), \alpha(B+AP) < 0.\end{gathered} \tag{45}$$

Control strategy for the linear and nonlinear system with the use of the triggered method subjected to DoS attacks depends on study of ISS-Lyapunov function has been described in [138–140]. Maximum %age of time loosing response data deprived of the foremost system is instability was characterized and an event-based controller for that presence of minimum inside sampling time is definite has been supposed.

6.2. Game Theory (GT) Approach. GT deals with planned collaboration in between several named players and decision makes [107, 140, 141]. Each player preference order in between many options is increased in an impartial purpose for player, and all players try to optimize own impartial function. It depends on the alternate of another player in any nontrivial game, and this process of optimization depends on the selection of second players [142]. For applications of game theory in the network, we can study the literature

[143–145]. For getting secure control in a lot of research studies, this method was pragmatic. Disadvantage which is because of DoS attack is designed as Markov process depends on the game among defending strategies and attack [145]. Using Lyapunov theory, four theorems were derived for assurance of the stability of system. For handling computation complexity of optimal strategies for both players, a Nash Q-learning algorithm is studied [144]. Sensor data are transmitted through a large number of channels remotely, making them vulnerable to malicious attacks. There is need to select one channel to sensor in between these paths with less probability to attack with data transmitting data. It is also decided by attackers that which channel is suitable for attack, e.g., [119]. From literature review, we can find some more examples of applying such kind of approach [142, 146].

CoFence Mechanism is assumed for DoS defense attack that endorsed "domain help domain" cooperative network between the NFV-based domain network. Furthermore, there is a dynamic resource allocation characterization for game, and we establish a game model to get incentive-compatible, effective, reciprocal, and fair resource allocation method to work on Nash equilibrium [147]. In [148], the authors supposed conflict between attacker and defender and designed a game theory framework for collective security detection.

6.3. Secure Control Approach. For the event triggering system or discrete time system, stochastic time-delay approach can be applied subject to arbitrary DoS attack. The system is designed using Markov process and Bernoulli process with identified statically information to govern the freely present DoS attack. In an event triggered system, impulsive system approach can be applied and is powerful in network control systems as studied in [135]. To reduce the communication in between system part triggering strategy is enough, since signal sent only specific condition of triggering is despoiled, that will minimize burden of communication.

Need of throughout information of the system is one limitation in game theory. With imperfect and incomplete information, game theory application is a developing field in network privacy and security. In addition, there is need of agents for correct estimation of security game limitations. For security measures and attack prevention, observation capabilities avail required basis [80, 149].

7. Model-Based Attack Methodology

There are two possible forms of occurrence of deception attacks: one is that targeted attacks which defined states are effected and random attacks where arbitrary measurements are defined [48]. In view of control engineering, it is designed as stochastic process [48, 150]. We supposed the following system for best understanding of this idea:

$$\begin{cases} Q(P+1) = B_Q(P) + A_W(P), \\ F^*(P) = H_Q(P), \\ F(P) = F^*(P) + \mu(P)h(P). \end{cases} \tag{46}$$

$W(P) \in R^{x_w}, Q(P) \in R^{x_n}$ and $F(P) \in R^{x_b}, F^*(P) \in R^{x_b}$ are the control input, states, received signal, and measured output, respectively, and $\mu(P)$ is Bernoulli distributed with deception occurrence possibility with values one and zero; so,

$$\begin{cases} \text{Prob}\{\mu(P) = 1\} = \mu^*, \\ \text{Prob}\{\mu(P) = 0\} = 1 - \mu^*. \end{cases} \tag{47}$$

Description of deception attack is

$$h(P) = -F^* + \tau(P). \tag{48}$$

(1) Note. It is considered that data transferred by attackers mean injected fault data could be subtracted into two steps according to representation of equation (48)

$-F^* =$ for cancellation of original signal, and $\tau(P)$ is supposed to be freely limited energy signal as characterized in [150]:

$$\coprod_{P=0}^{\infty} \tau^Z(P)\tau \le \tau^{-2}. \tag{49}$$

For the time varying class system, variance-constrained distributed problem direct to several divisions of noises, unidentified but limited turbulences, there is also study of deception over sensor network [151]. Present measurement at each node is gathered from both of neighbors and single sensors. There is insertion of deception signals into right signals of input used for controlling W_P and output measurements $F_{a,p}$ during data transmission process shown in Figure 11. An article for designing of deception attack is studied in which nasty signals are inserted by the adversary into both measurement and control data during information communication process via network communication. Following signals effect signals.

8. Deception Attack Detection or Identification

It is conscious issue of deception attack estimation for prevention of any detection mechanism since attack form, the main target of affecting stability of the system. Bias injection issue hitting Kalman filter in the system containing chi-square detector is studied [152]. It proved that worst situation problem quadratically constrained can be reduced as quadratic program permits to gain criterion that is useful for selection of sensors for safe and condition on number of sensors need to keep the attack effect with encoded threshold.

Centralized security problem for stochastic system linear time-invariant with multidate-sensors fusion subject to deception attack is studied in [6]. Data transferred on each sensor by adversaries as extra signal which makes feasible boundary situations such as [25, 150]. For formulation individual rate discrete time systems, there was use of lifting technique. Using stochastic analysis techniques, enough conditions were gained for gaining already determined original system security level. For effecting uniform quantization, deception attack was supposed in distributive

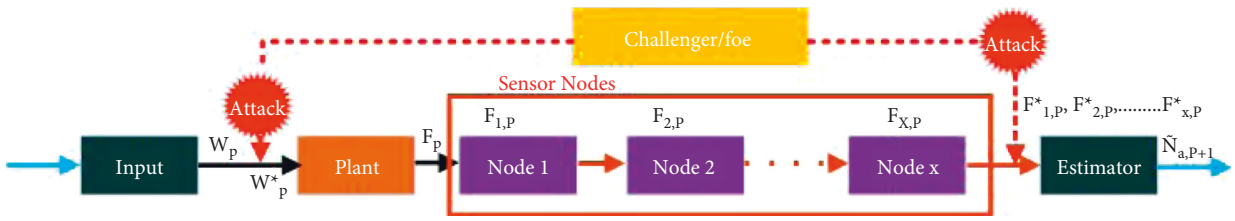


FIGURE 11: Deception attack schematic.

recursive filtering problem of stochastic system discrete time-delayed [6, 83, 153]. We supposed the following system for showing its working:

$$\begin{aligned} Q(P+1) = &\left(B_0(P) + \prod_{r=1}^{Z} \epsilon_Z(P) B_r(P)\right) Q(P) \\ &+ \left(B_0^{\varnothing}(P) + \prod_{r=1}^{Z} \epsilon_r(P) B_r^{\varnothing}(P)\right) Q(P+\rho) \\ &+ A(P) \in (P). \end{aligned} \tag{50}$$

With sensors "x" studied as

$$\begin{aligned} \widehat{F}_a(P) &= \left(H_0(P) + \widehat{\epsilon}_a(P) H_a(P)\right) Q(P) + O(P) h_a(P), \\ a &= 1, 2, 3, \ldots, x. \end{aligned} \tag{51}$$

$Q(P)$ state that directly cannot be observed. $\widehat{F}_a(P)$ is sensor "a" output without quantization. $\widehat{\epsilon}_a(P)$ and $h_a(P)$ are white distortions with zero means and conversance unity, collectively uncorrelated "P" and a, $\widehat{\epsilon}_a(P) \in R$, and $\epsilon_Z(P) \in R$ ($Z = 1, 2, 3, \ldots, z$) are multiplicative noises with unity variance and zero means, and jointly correlated in P. ρ and z are positive integers. $B_r(P), B_r^o(P)$ and $H_a(P)$ are identified fixed matrices with well-suited dimensions. In equations (48)–(50), the same dimension effect is studied. Upper limitation for error filtering covariance has been studied in [31].

For modeling of distributed state estimator, event triggered scheme is applied to the wireless sensor network for false data injection attack [31, 154]. Each sensor estimate is checked if it attacked at all-time step before transmission of data to nearby sensor, and it may stop in case of attacked. Using event triggered scheme, an optimal estimator gain is supposed to reduce mean square estimation fault covariance, and modeled distributed estimator stability is certain with enough condition of driving. Already the discussed Bayesian method was pragmatic for both estimation and detection of states for MASs subject to turn attack signal and wrong measurements [154, 155].

Kalman filter which ensures a safe state estimation algorithm for the stochastic dynamic system was studied in [156–158]. Adversary caused freely subset of sensors is to be supposed in this problem, and an upper limit on sensors effected by attacks was designed to uphold an adequate state approximation fault. Insecure estimation situation is studied for the control system of network direct to fault data insertion attack containing a $\aleph^2$ detector. In addition, a precise algorithm was used, and defense of rare communication channels instead of defending all is studied. Nonlinear stochastic discrete time-delay filtering problem in systems pretentious by arbitrary deception attack and arbitrary sensor saturation were studied in [159]. Suppose the system

$$\begin{aligned} Q(P+1) = &B_Q(P) + B_{\varnothing} \aleph(P - \varnothing(P)) + Ag(Q(P)) \\ &+ A_{\varnothing} g_{\varnothing}(Q(P - \varnothing(P))) + O_{\epsilon}(P). \end{aligned} \tag{52}$$

$Q(P) \in R^{x_n}$ = state vector $\in (P) \in R$ = zero mean Gaussian, and $B, B_{\varnothing}, A, A_{\varnothing}$, and O are identified constant matrices with suitable dimensions. The following condition is satisfied by nonlinear functions g and $g_{\varnothing}$:

$$\begin{aligned} &[g(Q) - P_1 Q].^Z [g(Q) - P_2 Q] \le 0, \\ &[g_{\varnothing}(Q) - Z_1 Q]^Z [g_{\varnothing}(Q) - Z_2 Q] \le 0. \end{aligned} \tag{53}$$

In this, P_1, P_2, Z_1, and Z_2 are appropriate dimensions real matrices. $P_1 = P_1 - P_2$ and $Z = Z_1 - Z_2$ are positive symmetric definite matrices. Given filter designed is supposed in this system:

$$\widehat{Q}(P+1) = Y\widehat{Q}(P) + X_f(P). \tag{54}$$

To ensure the required security level in the filtering system, an enough condition is derived with stochastic analysis technique. For filter obtaining, inequality linear matrix with constraints of nonlinear is resolved.

8.1. Secure Control Approaches of Deception Attack. Deception attack affected discrete time stochastic nonlinear system problem of security control with quadratic cost criterion is studied in [160]. Both actuating and measurement signals were directed to deception as in Figure 11:

$$W(P) = \widehat{W}(P) + \partial(P) \in (P). \tag{55}$$

In Figure 12, B1 and B2 are supposed to be attacker, in system false data system, e.g., $\in (P) = -\widehat{W}(P) + \mathscr{L}_1(P)$ and $h(P) = -\widehat{F}(P) + \mathscr{L}_2(P)$.,

$$W(P) = \widehat{W}(P) + \partial(P) \in (P), \tag{56}$$

$$F(P) = \widehat{F}(P) + L(P) h(P). \tag{57}$$

where $W(P)$ = actuator input, $\widehat{W}(P)$ = controller outputs directed to attacks, $F(P)$ = controller gained signal, $\widehat{F}(P)$ = sensor measurement directed to attacks, $\epsilon(P)$ and $h(P)$ = transmitted signals by attacker, and $\partial(P)$ and $\gamma(P)$ = Bernoulli distributed mutually independent with stochastic variable one and zero, with following probabilities:

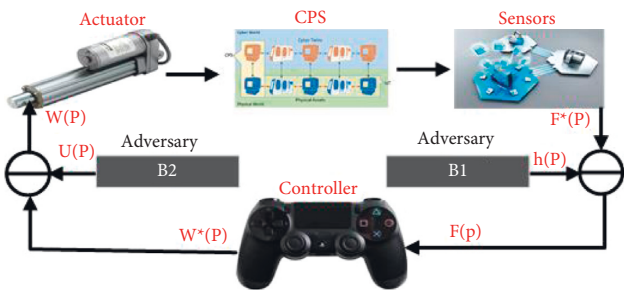


Figure 12: Deception attack schematic.

$$\begin{aligned} &\text{Prob}\{\partial(P) = 1\} = \hat{\partial}, \\ &\text{Prob}\{\partial(P) = 0\} = 1 - \hat{\partial}, \\ &\text{Prob}\{L(P) = 1\} = \hat{L}, \\ &\text{Prob}\{L(P) = 0\} = 1 - \hat{L}. \end{aligned} \tag{58}$$

In Figure 8, B1 and B2 are supposed to be attacker, in system false data system, e.g., $\in(P) = -\hat{W}(P) + \mathscr{L}_1(P)$ and $h(P) = -\hat{F}(P) + \mathscr{L}_2(P)$.

Structuring a dynamic output field or controller feedback is the basic purpose of this delinquent, e.g., given security in possibility is attained while gaining higher limited of the already choose quadratic cost function. Hence, to derive some enough situations by matrix discriminations form in input-to-state framework stability in possibility stochastic analysis approach was pragmatic. To apply matrix inverse lemma controller obtained upper bound.

In [161, 159], there is study of the secure network predictive control system and an architecture for secure and dependent automotive MASs, integrating data message digest algorithm, encryption standard algorithm, predictive control recursive network method, and time-stamp strategy. Predictive control recursive networks rely on time delays, which is pragmatic to ensure the performance of the system, especially when a deception attack influences it.It will accomodate the consequences of attacks and network flaws such as package disorder, package dropout, and time-varying delay.

We studied consensus control and consensus management problem in [162]. There was latest definition of quasi-consensus given for describing the consensus performance with constraints on each agent to keep within few ellipsoidal regions at all-time instant, which based on given topology. In addition, measured result is available for controller from both nearby and individual agents. For gaining quasi-consensus, enough situations are gained with the use of recursive matrix for required control system inequalities.

A resilient control system [139, 163] has been supposed for network control systems effected by false data injections attacks, so that using measurement data and control input they could not be find. Attack of zero variable on plant state variable is not identified during attack, and it seems after result of attack. Hence, a strong Gaussian controller which is linear quadratic is supposed so that there is online updating of Kalman filter from data transferred by an active version of comprehensive prospect ratio detector with the capability to speedy improve of behavior after attack [164].

Table 3: Categorization of cyber threats in cyberattacks, detection, and consensus of MASs.

Attack types	Work on cyberattack detection of MASs	Secure consensus of MASs
Actuator fault	[7, 54,167]	[90]
Sensor fault	[168, 169]	[27]
Actuator and sensor fault	[30, 169]	[31, 170]

Table 4: MAS challenges.

Sr. number	Challenges	Research work on MAS challenges
1	Security	[155, 172]
2	Learning	[173, 174]
3	Fault detection	[21, 33, 34, 175]
4	Localization	[176, 177]
5	Task allocation	[178–180]
6	Organization	[176, 177, 181]
7	Formation	[138, 182]
8	Connectivity	[167, 175, 183, 184]
9	Consensus	[167, 175, 183, 184]
10	Control ability	[175, 185, 186]
11	Synchronization	[175, 186]

Actuator attacks and sensor attacks for controller of MASs were proposed in [74, 164–166], and a progressive adaptive strong control scheme is discussed for adversarial mitigating attack in the cyber physical system. Nussbaum function with speedy progressive rate and estimation mechanism is adaptive bound. The double-step back step method was applied to mitigate effects of actuator attack and sensor attacks, and to apply exponentially decaying barrier Lyapunov function, a variable state was controlled.

A feasible control delinquent of MASs data-driven direct to actuator attack class is studied in [166], an unidentified nonstop time linear physical system containing outside instabilities was supposed, and input control signal sent via network layers is supposed to be vulnerable to cyberattacks. For eradicating actuator attack effect, nearby optimal performance and stability of MASs can be gained by data-based adaptive essential sliding-mode control approach. Use of abnormal monitor detection mechanism contains detector threshold information, frequency characteristics, and attack structure for a set of frequency constrained actuator, and sensor attack can be studied in [166]. Explained categorization of cyber threats in cyberattack detection of MASs and secure consensus of MASs has been studied in Table 3.

9. Key Challenges

Report [82] tells us clearly that there is no high-level security against upcoming attacks or threats. In addition, an open integrated ecosystem idea for cooperation of security issues was studied. Though, in security system, there should be collaboration of stalk holders, it will have advantage form face threat understanding.

Table 5: Summary of different strategies for MAS.

Application	Goal	Architecture	DRL algorithm	Q function estimator	Reference
Maximum power point tracking (MPPT) control	To solve MPPT problem of photovoltaic system in practical condition	Photovoltaic system microgrid base	Deep deterministic policy gradients algorithm	Deep neural network (DNN)	[153]
Secondary control/ frequency regulation	With enough load, restoring stability	Hybrid distributed power system as separated small grid	Actor-critic based algorithm	Neural network (NN)	[175]
Bus voltage stability	Designing DC-DC control converter	DC microgrid	Q network/deep Q network	NN	[166]

Table 6: Comparison of different methodologies of system security.

Benefits	Drawbacks	Methodology
Resiliency to GPS spoofing attack. Speedy and correct finding of anomaly in gained data. Resiliency to GPS spoofing attack	Require a mathematical model of system because there may be complication without any mathematical model when applying to system	Luenberger observer and artificial neural network (ANN) [172]
Resiliency to GPS spoofing and expected attacks	Susceptible to unforeseen attack. Costly against abrupt attack has low accuracy	Machine learning-based algorithm [39]
Low computation load. Resiliency to GPS spoof attack	Susceptible against suspicions and system disturbances, model accuracy dependent complicated to apply on nonlinear system	Model-based detection algorithm [148]
Secure conventional disturbances in between communication connections	Susceptible to GPS spoofing attack	Communication [62, 65]

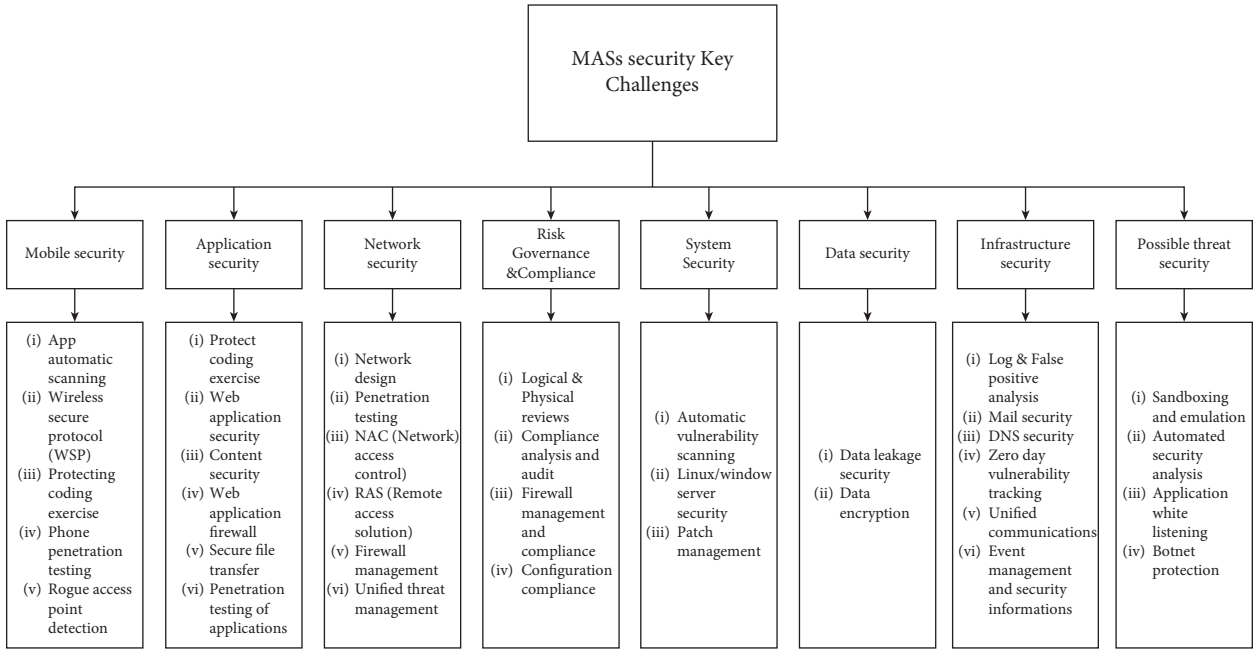


Figure 13: Overall MAS security challenges.

Key challenge for MASs is that there should be no system outside attack, but also from inside, e.g., a worker who is not interested to learn more about the board system. Designing of protective filter based on results of attack measurement for getting high security is one of the key challenges. To see present filtering technologies, those are not sufficient for security assurance, since it is complicated for defenders to get an idea about the time and trick of cyberattack. The Kalman filter method is not enough for MASs, and it is difficult to gain attacker in statically characterization of signal transmitted [107, 171]. In research studies, interest control problem and filtering in concern of security are getting more and attention, e.g., [79] in which there assurance of security against integrity attacks with the use of minimax optimization advancement [167].

With the deficiency of federal reliable power, agent identity verification and creating trust between agents are a big challenge. We can call it decentralization.

Basically, agents use knowledge or information which they get or need from decision-making process environment or another nearby agents. It makes an agent susceptible against malicious entries which may share false data to have effect on agent decision.

Highly important problem is needed to differentiate an accidental failure from an attack. The resulting sign from these accidental situations has chance of similarity, but reaction should not be the same. Fault is repairable. MASs should have capability to defend itself against attacks. Understanding of operations in MASs may be interrupted by malicious attack. A lot of attack stories have been presented in Table 1 of [78].

For MASs, integrity is an important requirement. There is need to pay attention toward sensor networks as well as to data integrity and superstructure. There is also not much more methodology in progressing of secure MAS, so there are several patented results, which may base on possibly exposed approaches.

For designing of applications, we need to see both quality of service and security assurance. For practical applications, considering of multiple attacks is an important point that we can face simultaneously. Present planning of security against several forms of attacks is insufficient for industrialization. In addition, security need and resource constraints as energy limitation and communication bandwidth in practically required to be supposed simultaneously.

Mobile devices are considered threats carrying because these are using several services and external networks. With the progressing of smart wearable mobile applications, loT of challenges presents in progressive of these applications' security measures, as there can be risk of human health and life.

For smart grids, basic challenges are communication protocol weakness, heterogeneity of protocols, and technology and limitation of physical systems. Table 4 represents research work on MAS challenges. Table 5 shows summary of different strategies for MAS while Table 6 represents comparison of different methodologies of system security. General security needs in MASs are integrity (which gives surety that since generation, there is no modification in message), authentication (that is sure that each agent is the one claim to be), confidentiality (which gives surety that only allowed agents are able to read specific data), availability, and authorization. Figure 13 represents overall security challenges in MASs.

10. Conclusion and Future Directions

MASs are virtually all around. They can be retrieved and switched remotely, such topographies make susceptible to cyberattacks. There is physical environment process on virtualization and cyber space as a key role for notion plays a central role in MAS. This article explained high-level inclusive discussion regarding various features of MASs that will aid new researchers to cover basic idea of MASs, key challenges in progressing MAS attack, e.g., system failure, virtualization and mobility, and MAS performance methods. First, we studied various attack types in MASs; second, we discussed threats with consistent subtypes and then their possible detection methodologies. After that we give detailed study of MAS attacks and their detection methodologies. Furthermore, an important work of this paper is subjected on several MAS aspects regarding security issues and key challenges. This article will play an important role for researchers to get maximum knowledge about MAS attacks and also to serve as an insightful and overall resources on MASs for researchers.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors hereby confirm that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Key RD Program of China (the major project no. 2020YFB1807900) of China.

References

[1] X. Ge, Q.-L. Han, and Z. Wang, "A threshold-parameter-dependent approach to designing distributed event-triggered H_{∞} consensus filters over sensor networks," *IEEE Transactions on Cybernetics*, vol. 49, no. 4, pp. 1148–1159, 2019.

[2] B. Liu, H.-T. Zhang, H. Meng, D. Fu, and H. Su, "Scanning-chain formation control for multiple unmanned surface vessels to pass through water channels," *IEEE Transactions on Cybernetics*, vol. 52, no. 3, pp. 1850–1861, 2022.

[3] D. Ding, Q.-L. Han, Y. Xiang, X. Ge, and X.-M. Zhang, "A survey on security control and attack detection for industrial cyber-physical systems," *Neurocomputing*, vol. 275, pp. 1674–1683, 2018.

[4] V. Belenko, V. Chernenko, V. Krundyshev, and M. Kalinin, "Data-driven failure analysis for the cyber physical infrastructures," in *Proceedings of the 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*, pp. 1–5, IEEE, Taipei, Taiwan, May 2019.

[5] M. Wolf and D. Serpanos, "Safety and security in cyber-physical systems and internet-of-things systems," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 9–20, 2018.

[6] D.-B. Pan, G. Zhang, S. Jiang, Y. Zhang, and B.-Y. Cui, "Delay-independent traffic flux control for a discrete-time lattice hydrodynamic model with time-delay," *Physica A: Statistical Mechanics and Its Applications*, vol. 563, p. 125440, Article ID 125440, 2021.

[7] B. Chen, L. Yu, D. W. C. Ho, and W.-A. Zhang, "Networked Fusion Estimation under Denial-of-Service Attacks," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3835–3840, 2017.

[8] M. Nasir, M. Muzamil Aslam, and Z. Ahmed, "Stabilization of an Arbitrary Order Transfer Function with Time Delay H2 Controller," 2021, https://www.researchgate.net/publication/357620374_Stabilization_of_an_arbitrary_order_transfer_function_with_time_delay_H2_Controller.

[9] B. Mrugalska and M. K. Wyrwicka, "Towards lean production in industry 4.0," *Procedia Engineering*, vol. 182, pp. 466–473, 2017.

[10] H. He and J. Yan, "Cyber-physical attacks and defences in the smart grid: a survey," *IET Cyber-Physical Systems: Theory & Applications*, vol. 1, no. 1, pp. 13–27, 2016.

[11] M. Muzamil Aslam, J. Zhang, B. Qureshi, and Z. Ahmed, "Beyond6g-consensus traffic management in crn, applications, architecture and key challenges," in *Proceedings of the 2021 IEEE 11th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 182–185, IEEE, Beijing, China, June 2021.

[12] Z. Ahmed, M. M. Khan, M. A. Saeed, and W. Zhang, "Consensus control of multi-agent systems with input and communication delay: a frequency domain perspective," *ISA Transactions*, vol. 101, pp. 69–77, 2020.

[13] C. H. Ho, S. C. Chan, Y. Hou, and Y. Hou, "A robust statistical approach to distributed power system state estimation with bad data," *IEEE Transactions on Smart Grid*, vol. 11, no. 1, pp. 517–527, 2020.

[14] M. M. Aslam, L. Du, Z. Ahmed, H. Azeem, and M. Ikram, "Consensus performance of traffic management system for cognitive radio network: an agent control approach," in *Communications in Computer and Information Science,* vol. 1138, pp. 138–145, Springer, 2019.

[15] B. Ning, Q.-L. Han, Z. Zuo, J. Jin, and J. Zheng, "Collective behaviors of mobile robots beyond the nearest neighbor rules with switching topology," *IEEE Transactions on Cybernetics*, vol. 48, no. 5, pp. 1577–1590, 2018.

[16] B. Ning, Q.-L. Han, and L. Ding, "Distributed finite-time secondary frequency and voltage control for islanded microgrids with communication delays and switching topologies," *IEEE Transactions on Cybernetics*, vol. 51, no. 8, pp. 3988–3999, 2021.

[17] A. Khan, F. Aftab, and Z. Zhang, "Self-organization based clustering scheme for fanets using glowworm swarm optimization," *Physical Communication*, vol. 36, p. 100769, Article ID 100769, 2019.

[18] A. Khan, F. Aftab, and Z. Zhang, "Bicsf: bio-inspired clustering scheme for fanets," *IEEE Access*, vol. 7, pp. 31446–31456, 2019.

[19] H. Yang, Q.-L. Han, X. Ge et al., "Fault-tolerant cooperative control of multiagent systems: a survey of trends and methodologies," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 4–17, 2020.

[20] B. Kailkhura, Y. S. Han, S. Brahma, and P. K. Varshney, "Distributed bayesian detection in the presence of byzantine data," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5250–5263, 2015.

[21] X.-Z. Jin, W.-W. Che, Z.-G. Wu, and Z. Zhao, "Adaptive consensus and circuital implementation of a class of faulty multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 1, pp. 226–237, 2022.

[22] S. Pal, B. Sikdar, and J. Chow, "Detecting data integrity attacks on scada systems using limited pmus," in *Proceedings of the 2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 545–550, IEEE, Sydney, NSW, Australia, November 2016.

[23] A. Stief, J. R. Ottewill, J. Baranowski, and M. Orkisz, "A pca and two-stage bayesian sensor fusion approach for diagnosing electrical and mechanical faults in induction motors," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9510–9520, 2019.

[24] L. Zhao and G.-H. Yang, "Adaptive sliding mode fault tolerant control for nonlinearly chaotic systems against dos attack and network faults," *Journal of the Franklin Institute*, vol. 354, no. 15, pp. 6520–6535, 2017.

[25] A. A. Yaseen and M. Bayart, "Cyber-attack detection with fault accommodation based on intelligent generalized predictive control," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 2601–2608, 2017.

[26] Steve Starrett, *World Environmental and Water Resources congress 2009*, American Society of Civil Engineers, Kansas City, p. 6664, 2009.

[27] S. Hu, H. Xiao, and C. Yi, "A novel detrended fluctuation analysis method for gear fault diagnosis based on variational mode decomposition," *Shock and Vibration*, vol. 2018, pp. 1–11, 2018.

[28] C. Liu, B. Jiang, R. J. Patton, and K. Zhang, "Decentralized output sliding-mode fault-tolerant control for heterogeneous multiagent systems," *IEEE Transactions on Cybernetics*, vol. 50, no. 12, pp. 4934–4945, 2020.

[29] L. Zhao and G.-H. Yang, "Adaptive fault-tolerant control for nonlinear multi-agent systems with dos attacks," *Information Sciences*, vol. 526, pp. 39–53, 2020.

[30] H. Wang, Y. Kang, L. Yao, H. Wang, and Z. Gao, "Fault Diagnosis and Fault Tolerant Control for T-S Fuzzy Stochastic Distribution Systems Subject to Sensor and Actuator Faults," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 11, pp. 3561–3569, 2021.

[31] A. Atta Yaseen and M. Bayart, "Cyber-attack detection in the networked control system with faulty plant," in *Proceedings of the 2017 25th Mediterranean Conference on Control and Automation (MED)*, pp. 980–985, IEEE, Valletta, Malta, July 2017.

[32] K. Michail, K. M. Deliparaschos, S. G. Tzafestas, and A. C. Zolotas, "Ai-based actuator/sensor fault detection with low computational cost for industrial applications," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 1, pp. 293–301, 2016.

[33] W. Zou, C. K. Ahn, and Z. Xiang, "Fuzzy-approximation-based distributed fault-tolerant consensus for heterogeneous switched nonlinear multiagent systems," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 10, pp. 2916–2925, 2020.

[34] C. Deng and C. Wen, "Distributed resilient observer-based fault-tolerant control for heterogeneous multiagent systems under actuator faults and dos attacks," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 3, pp. 1308–1318, 2020.

[35] Y. Shoukry and P. Tabuada, "Event-triggered state observers for sparse sensor noise/attacks," *IEEE Transactions on Automatic Control*, vol. 61, no. 8, pp. 2079–2091, 2016.

[36] F.-F. Wang and Q. H. Liu, "A Bernoulli-Gaussian Binary Inversion Method for High-Frequency Electromagnetic Imaging of Metallic Reflectors," *IEEE Transactions on Antennas and Propagation*, vol. 68, no. 4, pp. 3184–3193, 2020.

[37] T. Sui, Y. Mo, D. Marelli, X. Sun, and M. Fu, "The vulnerability of cyber-physical system under stealthy attacks," *IEEE Transactions on Automatic Control*, vol. 66, no. 2, pp. 637–650, 2021.

[38] K. Granström, M. Fatemi, and L. Svensson, "Poisson multi-Bernoulli mixture conjugate prior for multiple extended target filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 1, pp. 208–225, 2020.

[39] B. Xu, J. Shi, M. Lu, J. Cong, L. Wang, and B. Nener, "An automated cell tracking approach with multi-Bernoulli filtering and ant colony labor division," *IEEE/ACM*

Transactions on Computational Biology and Bioinformatics, vol. 18, no. 5, pp. 1850–1863, 2021.

[40] S. Amin, G. A. Schwartz, and S. Shankar Sastry, "Security of interdependent and identical networked control systems," *Automatica*, vol. 49, no. 1, pp. 186–192, 2013.

[41] A. Mishra, R. Singh Chowhan, and A. Mathur, "Sniffer detection and load balancing using aglets in a cluster of heterogeneous distributed system environment," in *Proceedings of the 2016 IEEE 7th Power India International Conference (PIICON)*, pp. 1–6, IEEE, Bikaner, India, November 2016.

[42] G. K. Befekadu, V. Gupta, and P. J. Antsaklis, "Risk-sensitive control under Markov modulated denial-of-service (dos) attack strategies," *IEEE Transactions on Automatic Control*, vol. 60, no. 12, pp. 3299–3304, 2015.

[43] P. C. Evans and M. Annunziata, "Industrial Internet: Pushing the Boundaries," *General Electric Reports*, pp. 488–508, 2012.

[44] J. Yan, F. Guo, and C. Wen, "False data injection against state estimation in power systems with multiple cooperative attackers," *ISA Transactions*, vol. 101, pp. 225–233, 2020.

[45] M. Tian, Z. Dong, and X. Wang, "Analysis of false data injection attacks in power systems: a dynamic bayesian game-theoretic approach," *ISA Transactions*, vol. 115, pp. 108–123, 2021.

[46] S. Xiao, Q.-L. Han, X. Ge, and Y. Zhang, "Secure distributed finite-time filtering for positive systems over sensor networks under deception attacks," *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 1220–1229, 2020.

[47] T. Zou, A. S. Bretas, C. Ruben, S. C. Dhulipala, and N. Bretas, "Smart grids cyber-physical security: parameter correction model against unbalanced false data injection attacks," *Electric Power Systems Research*, vol. 187, p. 106490, Article ID 106490, 2020.

[48] J. Hao, R. J. Piechocki, D. Kaleshi, W. H. Chin, and Z. Fan, "Sparse malicious false data injection attacks and defense mechanisms in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1–12, 2015.

[49] C. De Persis and P. Tesi, "Input-to-state stabilizing control under denial-of-service," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2930–2944, 2015.

[50] K.-L. Miao, J.-W. Zhu, and W.-A. Zhang, "Distributed guaranteed cost control of networked interconnected systems under denial-of-service attacks: a switched system approach," in *Proceedings of the 2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 911–915, IEEE, Nanjing, China, May 2018.

[51] Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (sdn) and distributed denial of service (ddos) attacks in cloud computing environments: a survey, some research issues, and challenges," *IEEE communications surveys & tutorials*, vol. 18, no. 1, pp. 602–622, 2016.

[52] O. A. Wahab, J. Bentahar, H. Otrok, and A. Mourad, "Optimal load distribution for the detection of vm-based ddos attacks in the cloud," *IEEE transactions on services computing*, vol. 13, no. 1, pp. 114–129, 2020.

[53] Y. Ali, Y. Xia, L. Ma, and A. Hammad, "Secure design for cloud control system against distributed denial of service attack," *Control Theory and Technology*, vol. 16, no. 1, pp. 14–24, 2018.

[54] J. H. Sarker and A. M. Nahhas, "Mobile rfid system in the presence of denial-of-service attacking signals," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 955–967, 2017.

[55] D. Kshirsagar and S. Kumar, "An Efficient Feature Reduction Method for the Detection of Dos Attack," *ICT Express*, vol. 7, no. 3, pp. 371–375, 2021.

[56] M. Long, C.-H. Wu, and J. Y. Hung, "Denial of service attacks on network-based control systems: impact and mitigation," *IEEE Transactions on Industrial Informatics*, vol. 1, no. 2, pp. 85–96, 2005.

[57] S. Liu, P. X. Liu, and A. El Saddik, "A stochastic game approach to the security issue of networked control systems under jamming attacks," *Journal of the Franklin Institute*, vol. 351, no. 9, pp. 4570–4583, 2014.

[58] M. Zhang, J. Chen, S. He, L. Yang, X. Gong, and J. Zhang, "Privacy-preserving database assisted spectrum access for industrial internet of things: a distributed learning approach," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 8, pp. 7094–7103, 2020.

[59] L. Peng, L. Shi, X. Cao, and C. Sun, "Optimal attack energy allocation against remote state estimation," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 2199–2205, 2018.

[60] R. Gan, Y. Xiao, J. Shao, and J. Qin, "An Analysis on Optimal Attack Schedule Based on Channel Hopping Scheme in Cyber-Physical Systems," *IEEE transactions on cybernetics*, vol. 51, no. 2, pp. 994–1003, 2021.

[61] J. Qin, M. Li, L. Shi, and X. Yu, "Optimal denial-of-service attack scheduling with energy constraint over packet-dropping networks," *IEEE Transactions on Automatic Control*, vol. 63, no. 6, pp. 1648–1663, 2018.

[62] H. Zhang, P. Cheng, L. Shi, and J. Chen, "Optimal dos attack scheduling in wireless networked control system," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 3, pp. 843–852, 2016.

[63] A. Benslimane and H. Nguyen-Minh, "Jamming attack model and detection method for beacons under multi-channel operation in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6475–6488, 2017.

[64] Y. Pan, Z. Zheng, and D. Fu, "Bayesian-based water leakage detection with a novel multisensor fusion method in a deep manned submersible," *Applied Ocean Research*, vol. 106, p. 102459, Article ID 102459, 2021.

[65] M. Conti, N. Dragoni, and V. Lesyk, "A survey of man in the middle attacks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2027–2051, 2016.

[66] I. Paul, "Lenovo preinstalls man-in-the-middle adware that hijacks," *Superfish may make it trivial for attackers to spoof any HTTPS website*, 2015.

[67] Di Wu, D. I Arkhipov, E. Asmare, Z. Qin, and J. A. McCann, "Ubiflow: mobility management in urban-scale software defined iot," in *Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM)*, pp. 208–216, IEEE, Hong Kong, China, April 2015.

[68] C. Li, Z. Qin, E. Novak, and Q. Li, "Securing SDN Infrastructure of IoT-Fog Networks From MitM Attacks," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1156–1164, 2017.

[69] G. Oliva, S. Cioaba, and C. N. Hadjicostis, "Distributed calculation of edge-disjoint spanning trees for robustifying distributed algorithms against man-in-the-middle attacks," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 4, pp. 1646–1656, 2018.

[70] J. Huang, D. W. C. Ho, F. Li, W. Yang, and Y. Tang, "Secure remote state estimation against linear man-in-the-middle attacks using watermarking," *Automatica*, vol. 121, p. 109182, Article ID 109182, 2020.

[71] Y. Mirsky, N. Kalbo, Y. Elovici, and A. Shabtai, "Vesper: using echo analysis to detect man-in-the-middle attacks in LANs," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1638–1653, 2019.

[72] A. Esfahani, G. Mantas, J. Ribeiro et al., "An efficient web authentication mechanism preventing man-in-the-middle attacks in industry 4.0 supply chain," *IEEE Access*, vol. 7, pp. 58981–58989, 2019.

[73] A. A. M Mazharul Amin and M S Mahamud, "An alternative approach of mitigating arp based man-in-the-middle attack using client site bash script," in *Proceedings of the 2019 6th International Conference on Electrical and Electronics Engineering (ICEEE)*, pp. 112–115, IEEE, Istanbul, Turkey, April 2019.

[74] M. Zhu and S. Martinez, "On the performance analysis of resilient networked control systems under replay attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 804–808, 2014.

[75] Y. Mo and B. Sinopoli, "Secure estimation in the presence of integrity attacks," *IEEE Transactions on Automatic Control*, vol. 60, no. 4, pp. 1145–1151, 2015.

[76] H. Tan, B. Shen, Y. Liu, A. Alsaedi, and B. Ahmad, "Event-triggered multi-rate fusion estimation for uncertain system with stochastic nonlinearities and colored measurement noises," *Information Fusion*, vol. 36, pp. 313–320, 2017.

[77] R. Merco, Z. Abdollahi Biron, and P. Pisu, "Replay attack detection in a platoon of connected vehicles with cooperative adaptive cruise control," in *Proceedings of the 2018 Annual American Control Conference (ACC)*, pp. 5582–5587, IEEE, Milwaukee, WI, USA, June 2018.

[78] S. Garg, K. Kaur, G. Kaddoum, K. K. R Choo, and R. C. Kwang, "Toward secure and provable authentication for internet of things: realizing industry 4.0," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4598–4606, 2020.

[79] M. A. Quddus, O. Shahvari, M. Marufuzzaman, J. M. Usher, and R. Jaradat, "A collaborative energy sharing optimization model among electric vehicle charging stations, commercial buildings, and power grid," *Applied Energy*, vol. 229, pp. 841–857, 2018.

[80] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Bacşar, and J.-P. Hubaux, "Game theory meets network security and privacy," *ACM Computing Surveys*, vol. 45, no. 3, pp. 1–39, 2013.

[81] Y. Mo and S. Bruno, "Secure control against replay attacks," in *Proceedings of the 2009 47th annual Allerton conference on communication, control, and computing (Allerton)*, pp. 911–918, IEEE, Monticello, IL, USA, October 2009.

[82] B. Tang, L. D. Alvergue, and G. Gu, "Secure networked control systems against replay attacks without injecting authentication noise," in *Proceedings of the 2015 American Control Conference (ACC)*, pp. 6028–6033, IEEE, Chicago, IL, USA, July 2015.

[83] B. J. Austin, V. Heine, and L. J. Sham, "General theory of pseudopotentials," *Physical Review*, vol. 127, no. 1, pp. 276–282, 1962.

[84] R. Kibble, "Speech acts, commitment and multi-agent communication," *Computational & Mathematical Organization Theory*, vol. 12, no. 2-3, pp. 127–145, 2006.

[85] M. Colombetti and M. Verdicchio, "An analysis of agent speech acts as institutional actions," in *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: Part 3*, pp. 1157–1164, DBLP, Bologna, Italy, 2002.

[86] D. Ding, Z. Wang, G. Wei, and F. E. Alsaadi, "Event-based security control for discrete-time stochastic systems," *IET Control Theory & Applications*, vol. 10, no. 15, pp. 1808–1815, 2016.

[87] S. Amin, X. Litrico, S. Sastry, and A. M. Bayen, "Cyber Security of Water SCADA Systems-Part I: Analysis and Experimentation of Stealthy Deception Attacks," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1963–1970, 2013.

[88] G. Falco, C. Caldera, and H. Shrobe, "Iiot cybersecurity risk modeling for scada systems," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4486–4495, 2018.

[89] S. Adepu and A. Mathur, "Distributed attack detection in a water treatment plant: method and case study," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 1, pp. 86–99, 2021.

[90] G. Wen, X. Zhai, Z. Peng, and A. Rahmani, "Fault-tolerant secure consensus tracking of delayed nonlinear multi-agent systems with deception attacks and uncertain parameters via impulsive control," *Communications in Nonlinear Science and Numerical Simulation*, vol. 82, p. 105043, Article ID 105043, 2020.

[91] E. Drayer and T. Routtenberg, "Detection of false data injection attacks in smart grids based on graph signal processing," *IEEE Systems Journal*, vol. 14, no. 2, pp. 1886–1896, 2020.

[92] S. Li, K. Xue, D. S. L. Wei, H. Yue, N. Yu, and P. Hong, "Secgrid: a secure and efficient sgx-enabled smart grid system with rich functionalities," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1318–1330, 2020.

[93] M. Muzamil Aslam, M. Nauman Irshad, and A. Z. E. E. M. Hassan, "Cost effective & energy efficient intelligent smart home system based on iot," *Afyon Kocatepe Üniversitesi Uluslararası Mühendislik Teknolojileri ve Uygulamalı Bilimler Dergisi*, vol. 3, no. 1, pp. 10–20.

[94] M. A. Ferrag and L. Maglaras, "Deepcoin: a novel deep learning and blockchain-based energy exchange framework for smart grids," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1285–1297, 2020.

[95] Y. Dong, N. Gupta, and N. Chopra, "False data injection attacks in bilateral teleoperation systems," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 1168–1176, 2020.

[96] D. Ye and T.-Y. Zhang, "Summation detector for false data-injection attack in cyber-physical systems," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2338–2345, 2020.

[97] A. S. Musleh, G. Chen, and Z. Y. Dong, "A survey on the detection algorithms for false data injection attacks in smart grids," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2218–2234, 2020.

[98] W. Meng, W. Li, C. Su, J. Zhou, and R. Lu, "Enhancing trust management for wireless intrusion detection via traffic sampling in the era of big data," *IEEE Access*, vol. 6, pp. 7234–7243, 2018.

[99] S. Aoufi, A. Derhab, and M. Guerroumi, "Survey of false data injection in smart power grid: attacks, countermeasures and challenges," *Journal of Information Security and Applications*, vol. 54, p. 102518, Article ID 102518, 2020.

[100] G. Yadav and K. Paul, "Assessment of scada system vulnerabilities," in *Proceedings of the 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1737–1744, IEEE, Zaragoza, Spain, September 2019.

[101] Ş. Dalgaç, F. Karadağ, M. Bakır, O. Akgöl, E. Ünal, and M. Karaaslan, "Chiral metamaterial-based sensor applications to determine quality of car lubrication oil," *Transactions of the Institute of Measurement and Control*, vol. 43, no. 7, pp. 1640–1649, 2021.

[102] Y. Mo and B. Sinopoli, "On the performance degradation of cyber-physical systems under stealthy integrity attacks," *IEEE Transactions on Automatic Control*, vol. 61, no. 9, pp. 2618–2624, 2016.

[103] D. B. Rawat and C. Bajracharya, "Detection of false data injection attacks in smart grid communication systems," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1652–1656, 2015.

[104] A. Karatepe, O. Akgöl Yadgar, and I. Abdulkarim, "A monopole microwave-assisted electrochemical sensor for the detection of liquid chemicals," *Digest Journal of Nanomaterials & Biostructures (DJNB)*, vol. 16, no. 3, 2021.

[105] Z. Guo, D. Shi, K. H. Johansson, and L. Shi, "Optimal linear cyber-attack on remote state estimation," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 4–13, 2017.

[106] Z. Ahmed, M. A. Saeed, A. Jenabzadeh, and Z. Weidong, "Frequency domain analysis of resilient consensus in multi-agent systems subject to an integrity attack," *ISA Transactions*, vol. 111, pp. 156–170, 2021.

[107] M. M. Aslam, L. Du, X. Zhang, Y. Chen, Z. Ahmed, and B. Qureshi, "Sixth generation (6g) cognitive radio network (crn) application, requirements, security issues, and key challenges," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–18, 2021.

[108] S. Keele, "Guidelines for Performing Systematic Literature Reviews in Software Engineering," *EBSE*, vol. 259, 2007.

[109] M. M. Aslam, L. Du, Z. Ahmed, M. N. Irshad, and H. Azeem, "A deep learning-based power control and consensus performance of spectrum sharing in the cr network," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–16, 2021.

[110] L. Cheng, Ke Tian, D. Yao, L. Sha, and R. A. Beyah, "Checking Is Believing: Event-Aware Program Anomaly Detection in Cyber-Physical Systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 973pp. 353–1441, USA, 2019.

[111] Y. I. Abdulkarim, Ş. Dalgaç, F. O. Alkurt et al., "Utilization of a triple hexagonal split ring resonator (srr) based metamaterial sensor for the improved detection of fuel adulteration," *Journal of Materials Science: Materials in Electronics*, vol. 32, no. 19, pp. 24258–24272, 2021.

[112] S. Dalgac, V. Akdogan, S. Kiris et al., "Investigation of methanol contaminated local spirit using metamaterial based transmission line sensor," *Measurement*, vol. 178, p. 109360, Article ID 109360, 2021.

[113] M. Nasir, M. F. Hayat, A. Jamal, and Z. Ahmed, "Frequency domain consensus control analysis of the networked multi-agent system with controller area network bus-induced delay," *Journal of Vibration and Control*, p. 107754632110224, Article ID 107754632110224, 2021.

[114] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[115] J. Miloševič, T. Tanaka, H. Sandberg, and K. H. Johansson, "Analysis and mitigation of bias injection attacks against a kalman filter," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8393–8398, 2017.

[116] W.-J. He, H.-T. Zhang, Z. Chen et al., "Temperature control for nano-scale films by spatially-separated atomic layer deposition based on generalized predictive control," *IEEE Transactions on Nanotechnology*, vol. 14, no. 6, pp. 1094–1103, 2015.

[117] X. Li and K. W. Hedman, "Enhancing power system cyber-security with systematic two-stage detection strategy," *IEEE Transactions on Power Systems*, vol. 35, no. 2, pp. 1549–1561, 2020.

[118] B. Li, Y. Chen, S. Huang, R. Yao, Y. Xia, and S. Mei, "Graphical evolutionary game model of virus-based intrusion to power system for long-term cyber-security risk evaluation," *IEEE Access*, vol. 7, pp. 178605–178617, 2019.

[119] F. Li, R. Xie, B. Yang et al., "Detection and identification of cyber and physical attacks on distribution power grids with pvs: an online high-dimensional data-driven approach," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 10, no. 1, pp. 1282–1291, 2022.

[120] Y. Zhang, V. V. G. Krishnan, J. Pi et al., "Cyber physical security analytics for transactive energy systems," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 931–941, 2020.

[121] B. Kailkhura, Y. S. Han, S. Brahma, and P. K. Varshney, "Asymptotic analysis of distributed bayesian detection with byzantine data," *IEEE Signal Processing Letters*, vol. 22, no. 5, pp. 608–612, 2015.

[122] A. D'Addabbo, A. Refice, G. Pasquariello, F. P. Lovergine, D. Capolongo, and S. Manfreda, "A bayesian network for flood detection combining sar imagery and ancillary data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 6, pp. 3612–3625, 2016.

[123] H. Wang, J. Ruan, B. Zhou et al., "Dynamic data injection attack detection of cyber physical power systems with uncertainties," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 10, pp. 5505–5518, 2019.

[124] H. Zhang, Y. Qi, H. Zhou, J. Zhang, and J. Sun, "Testing and defending methods against dos attack in state estimation," *Asian Journal of Control*, vol. 19, no. 4, pp. 1295–1305, 2017.

[125] T. Hirakawa, K. Ogura, B. Bahadur Bista, and T. Takata, "A Defense Method against Distributed Slow http dos attack," in *Proceedings of the 2016 19th international conference on network-based information systems (NBiS)*, pp. 152–158, IEEE, Ostrava, Czech Republic, September 2016.

[126] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[127] J. Shi, G. Qi, Y. Li, and A. Sheng, "Stochastic convergence analysis of cubature kalman filter with intermittent observations," *Journal of Systems Engineering and Electronics*, vol. 29, no. 4, pp. 823–833, 2018.

[128] Y. Wu, Y. Li, and L. Shi, "A game-theoretic approach to remote state estimation in presence of a dos attacker," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 2595–2600, 2017.

[129] Y. Li, D. E. Quevedo, S. Dey, and L. Shi, "A game-theoretic approach to fake-acknowledgment attack on cyber-physical systems," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 1, pp. 1–11, 2017.

[130] B. Chen, D. W. C. Ho, W.-A. Zhang, and L. Yu, "Distributed dimensionality reduction fusion estimation for cyber-physical systems under dos attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 2, pp. 455–468, 2019.

[131] C. Yang, W. Yang, and H. Shi, "Dos attack in centralised sensor network against state estimation," *IET Control Theory & Applications*, vol. 12, no. 9, pp. 1244–1253, 2018.

[132] D. He, C. Chen, S. Chan, and J. Bu, "Dicode: dos-resistant and distributed code dissemination in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 11, no. 5, pp. 1946–1956, 2012.

[133] Y. Xu, M. Fang, Z.-G. Wu, Y. J. Pan, Y.-J. Chadli, and T. Huang, "Input-based event-triggering consensus of multiagent systems under denial-of-service attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 4, pp. 1455–1464, 2020.

[134] V. S. Dolk, P. Tesi, C. De Persis, and W. P. M. H. Heemels, "Output-based event-triggered control systems under denial-of-service attacks," in *Proceedings of the 2015 54th IEEE Conference on Decision and Control (CDC)*, pp. 4824–4829, IEEE, Osaka, Japan, December 2015.

[135] V. S. Dolk, P. Tesi, C. De Persis, and W. P. M. H. Heemels, "Event-triggered control systems under denial-of-service attacks," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 93–105, 2017.

[136] S. Feng, P. Tesi, and C. De Persis, "Towards stabilization of distributed systems under denial-of-service," in *Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 5360–5365, IEEE, Melbourne, VIC, Australia, December 2017.

[137] H. Shisheh Foroush and S. Martinez, "On event-triggered control of linear systems under periodic denial-of-service jamming attacks," in *Proceedings of the 2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 2551–2556, IEEE, Maui, HI, USA, December 2012.

[138] S. Su, Z. Lin, and A. Garcia, "Distributed synchronization control of multiagent systems with unknown nonlinearities," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 325–338, 2016.

[139] L. Ding, Q.-L. Han, B. Ning, and D. Yue, "Distributed resilient finite-time secondary control for heterogeneous battery energy storage systems under denial-of-service attacks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4909–4919, 2020.

[140] E. Mousavinejad, X. Ge, Q.-L. Han, F. Yang, and L. Vlacic, "Resilient tracking control of networked control systems under cyber attacks," *IEEE Transactions on Cybernetics*, vol. 51, no. 4, pp. 2107–2119, 2021.

[141] D. Smirnov and A. Golkar, "Design optimization using game theory," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 2, pp. 1302–1312, 2021.

[142] Q. Zhu and Tamer Basar, "Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: games-in-games principle for optimal cross-layer resilient control systems," *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 46–65, 2015.

[143] A. Riahi Sfar, Y. Challal, P. Moyal, and E. Natalizio, "A game theoretic approach for privacy preserving model in iot-based transportation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4405–4414, 2019.

[144] J. Zhang, Y. Zhu, and Z. Chen, "Evolutionary game dynamics of multiagent systems on multiple community networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4513–4529, 2020.

[145] S. Guan, J. Wang, H. Yao, C. Jiang, Z. Han, and Y. Ren, "Colonel blotto games in network systems: models, strategies, and applications," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 637–649, 2020.

[146] N. Hou, Z. Wang, D. W. C. Ho, and H. Dong, "Robust partial-nodes-based state estimation for complex networks under deception attacks," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2793–2802, 2020.

[147] B. Rashidi, C. Fung, and E. Bertino, "A collaborative ddos defence framework using network function virtualization," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2483–2497, 2017.

[148] K. Wang and M. Du, "Game-theory-based Active Defense for Intrusion Detection in Cyber-Physical Embedded Systems," *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 1, pp. 1–21, 2016.

[149] Y. He, H. Li, X. Cheng, Y. Liu, C. Yang, and L. Sun, "A blockchain based truthful incentive mechanism for distributed p2p applications," *IEEE Access*, vol. 6, pp. 27324–27335, 2018.

[150] D. Ding, Z. Wang, Q.-L. Han, and G. Wei, "Security control for discrete-time stochastic nonlinear systems subject to deception attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 5, pp. 779–789, 2018.

[151] L. Ma, Z. Wang, Q.-L. Han, and H.-K. Lam, "Variance-constrained distributed filtering for time-varying systems with multiplicative noises and deception attacks over sensor networks," *IEEE Sensors Journal*, vol. 17, no. 7, pp. 2279–2288, 2017.

[152] E. Mousavinejad, F. Yang, Q.-L. Han, and L. Vlacic, "A novel cyber attack detection method in networked control systems," *IEEE Transactions on Cybernetics*, vol. 48, no. 11, pp. 3254–3264, 2018.

[153] X. You, S. Dian, R. Guo, and S. Li, "Exponential stability analysis for discrete-time quaternion-valued neural networks with leakage delay and discrete time-varying delays," *Neurocomputing*, vol. 430, pp. 71–81, 2021.

[154] H. Zhang, X. Zhou, Z. Wang, and H. Yan, "Maneuvering target tracking with event-based mixture kalman filter in mobile sensor networks," *IEEE Transactions on Cybernetics*, vol. 50, no. 10, pp. 4346–4357, 2020.

[155] X. Wang, M. Maghami, and G. Sukthankar, "Leveraging network properties for trust evaluation in multi-agent systems," in *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, vol. 2, pp. 288–295, IEEE, Lyon, France, August 2011.

[156] T. Shinohara, T. Namerikawa, and Z. Qu, "Resilient reinforcement in secure state estimation against sensor attacks with a priori information," *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5024–5038, 2019.

[157] Y. Yuan, H. Yuan, L. Guo, H. Yang, and S. Sun, "Resilient control of networked control system under dos attacks: a unified game approach," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1786–1794, 2016.

[158] S. Mishra, Y. Shoukry, N. Karamchandani, S. N. Diggavi, and P. Tabuada, "Secure state estimation against sensor attacks in the presence of noise," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 49–59, 2016.

[159] V. A. Kumar, D. Das, and I. E. E. E. Senior Member, "Data Sequence Signal Manipulation in Multipath Tcp: The Vulnerability, Attack and its Detection," *Computers & Security*, vol. 103, Article ID 102180, 2021.

[160] S. Q. Ali Shah, F. Z. Khan, and M. Ahmad, "The Impact and Mitigation of Icmp Based Economic Denial of Sustainability Attack in Cloud Computing Environment Using Software Defined Network," *Computer Networks*, vol. 187, Article ID 107825, 2021.

[161] Z.-H. Pang and G.-P. Liu, "Design and implementation of secure networked predictive control systems under

deception attacks," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 5, pp. 1334–1342, 2011.

[162] L. Ma, Z. Wang, and Y. Yuan, "Consensus control for nonlinear multi-agent systems subject to deception attacks," in *Proceedings of the 2016 22nd International Conference on Automation and Computing (ICAC)*, pp. 21–26, IEEE, Colchester, UK, September 2016.

[163] Y. Pang, H. Xia, and M. J. Grimble, "Resilient nonlinear control for attacked cyber-physical systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 6, pp. 2129–2138, 2020.

[164] T. Rhouma, K. Chabir, and M. N. Abdelkrim, "Resilient control for networked control systems subject to cyber/physical attacks," *International Journal of Automation and Computing*, vol. 15, no. 3, pp. 345–354, 2018.

[165] G. Franzè, F. Tedesco, and W. Lucia, "Resilient control for cyber-physical systems subject to replay attacks," *IEEE Control Systems Letters*, vol. 3, no. 4, pp. 984–989, 2019.

[166] A. Abbaspour, A. Sargolzaei, and K. Yen, "A neural network based resilient control design for distributed power systems under faults and attacks," in *Proceedings of the 2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, pp. 1–6, IEEE, Palermo, Italy, June 2018.

[167] Q.-K. Li, H. Lin, Xi Tan, and S. Du, "H consensus for multiagent-based supply chain systems under switching topology and uncertain demands," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 12, pp. 4905–4918, 2018.

[168] L. E. Venghi, F. Aguilera, P. Martin de la Barrera, C. Hernán, and C. H. De Angelo, "Detection and isolation of current-sensor and open-switch faults in electric traction drives," *IEEE Latin America Transactions*, vol. 19, no. 8, pp. 1335–1346, 2021.

[169] P. S. Kumar, L. Xie, M. S. M. Halick, and V. Vaiyapuri, "Stator end-winding thermal and magnetic sensor arrays for online stator inter-turn fault detection," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5312–5321, 2021.

[170] D. Zhang, G. Feng, Y. Shi, and D. Srinivasan, "Physical safety and cyber security analysis of multi-agent systems: a survey of recent advances," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, pp. 319–333, 2021.

[171] J. Jasper and S. R Isaac, "A Secure Routing Scheme to Mitigate Attack in Wireless Adhoc Sensor Network," *Computers & Security*, vol. 102, Article ID 102197, 2021.

[172] A. Abbaspour, A. Sargolzaei, P. Forouzannezhad, K. K. Yen, and A. I. Sarwat, "Resilient control design for load frequency control system under false data injection attacks," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 9, pp. 7951–7962, 2020.

[173] T. Liang, Y. Lin, L. Shi, J. Li, Y. Zhang, and Y. Qian, "Distributed vehicle tracking in wireless sensor network: a fully decentralized multiagent reinforcement learning approach," *IEEE Sensors Letters*, vol. 5, pp. 1–4, 2020.

[174] Y. Hou, Y.-S. Ong, J. Tang, and Y. Zeng, "Evolutionary multiagent transfer learning with model-based opponent behavior prediction," *IEEE transactions on systems, man, and cybernetics: Systems*, vol. 51, no. 10, pp. 5962–5976, 2021.

[175] W. Meng, P. X. Liu, Q. Yang, and Y. Sun, "Distributed synchronization control of nonaffine multiagent systems with guaranteed performance," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1571–1580, 2020.

[176] B. Horling and V. Lesser, "A survey of multi-agent organizational paradigms," *The Knowledge Engineering Review*, vol. 19, no. 4, pp. 281–316, 2004.

[177] T. T. Nguyen, S. Nahavandi, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: a review of challenges, solutions, and applications," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3826–3839, 2020.

[178] C. Yu, M. Zhang, F. Ren, and G. Tan, "Multiagent learning of coordination in loosely coupled multiagent systems," *IEEE Transactions on Cybernetics*, vol. 45, no. 12, pp. 2853–2867, 2015.

[179] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 156–172, 2008.

[180] X.-F. Xie and J. Liu, "Multiagent optimization system for solving the traveling salesman problem (tsp)," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics: A Publication of the IEEE Systems, Man, and Cybernetics Society*, vol. 39, no. 2, pp. 489–502, 2009.

[181] J. Shi, J. Wan, H. Yan, and H. Suo, "A survey of cyber-physical systems," in *Proceedings of the 2011 international conference on wireless communications and signal processing (WCSP)*, pp. 1–6, IEEE, Nanjing, China, November 2011.

[182] H. Zhang, J. H. Park, D. Yue, and X. Xie, "Finite-horizon optimal consensus control for unknown multiagent state-delay systems," *IEEE Transactions on Cybernetics*, vol. 50, no. 2, pp. 402–413, 2018.

[183] L. Hao, X. Zhan, J. Wu, T. Han, and H. Yan, "Bipartite finite time and fixed time output consensus of heterogeneous multiagent systems under state feedback control," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 6, pp. 2067–2071, 2020.

[184] S. Xu, J. Cao, Q. Liu, and L. Rutkowski, "Optimal control on finite-time consensus of the leader-following stochastic multiagent system with heuristic method," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 6, pp. 3617–3628, 2021.

[185] D. Li, S. S. Ge, and T. H. Lee, "Fixed-time-synchronized consensus control of multiagent systems," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 1, pp. 89–98, 2021.

[186] M. S. Mahmoud and B. J. Karaki, "Output-synchronization of discrete-time multiagent systems: a cooperative event-triggered dissipative approach," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 1, pp. 114–125, 2020.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4720169, 7 pages
https://doi.org/10.1155/2022/4720169

Research Article

Design and Protection Strategy of Distributed Intrusion Detection System in Big Data Environment

Rong Chen

Shanghai Customs College, Shanghai 201204, China

Correspondence should be addressed to Rong Chen; rong_chen016@126.com

Received 30 May 2022; Revised 13 June 2022; Accepted 16 June 2022; Published 29 June 2022

Academic Editor: Konstantinos Demertzis

One of the important research topics is protecting the host from threats by developing a reliable and accurate intrusion detection system. However, since the amount of data has grown fast due to the emergence of big data, the performance of traditional systems designed to identify breaches has suffered several flaws. One of them, for example, is known as single-point failure; low adaptability and a high false alarm rate are also typical. Hadoop is used to detect intrusions to tackle these difficulties. The Java system is used to create a framework with a significant data flow that detects intrusions when a distributed system is built. The proposed solution employs a distributed operating system for data collection, storage, and analysis. The results indicate that external distributed denial of service (DDoS) attacks are recognized quickly. The single-point failure issue is overcome, alleviating the bottleneck problem of data processing ability.

1. Introduction

More complicated and costly security issues appear to be treated efficiently as computer network technology rapidly advances. Hence, they trigger to cause the development of diverse technologies that deal with information security in several aspects. For example, while the firewall generally defends information security as an intrusion detection technology, the upgraded version is called a defense firewall [1, 2]. The intrusion detection system aims typically at collecting and crunching critical information and utilizes some predefined rules or protocols to detect irregular actions or transitions occurring supposedly against security policies and the predefined set of rules that were constructed using historical records when network technology is under investigation. By doing so, unauthorized, abnormal, or out-of-predefined rules are timely alerted to the security unit of the system. The system's efficiency is related to an issue called the single-point failure that the intrusion detection system faces often. Besides, the implementation constraints of processing vast and complicated data under the significant data cases grow more complex [3, 4]. Therefore, a novel approach called intrusion detection systems employing distributed systems under a big data environment could bring plenty of advantages for protecting information security.

Big data brings several issues to deal with, namely, a vast data set whose structure changes fast and includes diverse data types such as numeric, categoric, and unstructured. Besides, the standard computer configuration processing mode cannot satisfy the different requirements to process big data. So, Hadoop, launched by the Apache community [5] as a platform to crunch big data, employs a mapping reduction model (MapReduce) to conduct fundamental tasks of sorting and calculations. Moreover, Hadoop provides a cluster architecture to distribute tasks to the constituents of the distributed system. Hadoop functions as independent clusters, so the whole system continues to operate when any failure occurs regarding one or several nodes. A distributed computing proficiency is associated with the cluster architecture of Hadoop. In other terms, when each node of a cluster is assigned to take care of large-scale data analysis tasks, the efficiency of data analysis is greatly improved. Therefore, this capability can be smoothly

implemented to detect intrusions. In conclusion, the issues related to single-point failure and bottlenecks due to the low capacity of data processing can be effectively resolved.

Parallel programs featuring distributed architecture can be written, run, executed, and employed to run calculations regarding massive data sets on Hadoop clusters, an open-source distributed parallel programming widely implemented by enterprises and research institutions. MapReduce, the computing model of Hadoop, constructed by Google, is an effective tool.

The Hadoop distributed file system (HDFS) uses two key technologies. While the first one is the storage tool, the other is called the MapReduce distributed computing framework. The storage mechanism provides the underlying support for Hadoop. HDFS generally consists of Client, Datanode, and Namenode. When a cluster utilizes the architecture of Hadoop, a host called the Namenode and several hosts called the Datanode are available. The client represents the program employing HDFS. Namenode is a responsible host in the Hadoop cluster to save information of data nodes and distribute computational tasks and the final protocol. Datanode is responsible for data storage and processing. To ensure data security, HDFS increases the amount of redundant data moderately. The specific method is to save three copies of the same data in different Datanodes.

The parallel computing process running on a large-scale cluster is split into two functions: Map and Reduce. The other key technology is called MapReduce distributed computing framework, a mode to process and generate large datasets. The calculation process of MapReduce is based on decomposing large datasets into thousands of small datasets. Then, some datasets are distributed to a cluster node to process and produce intermediate outcomes. Finally, these results are obtained by several nodes to form the final outputs.

Big data technology is combined with the process of intrusion detection in [6], and an extensible quasi-real-time intrusion detection system employing Hadoop is suggested, which uses Hive and Mahout technologies to detect p2p botnet attacks. The Hive module functions as surveilling and processing network traces. Since the Mahout module provides parallel solid processing capability, constructing a decision tree model of random forest can be smoothly realized to detect botnets. A distributed intrusion detection system based on cloud computing utilizing the K-means clustering algorithm is proposed by [7], which resulted in higher detection accuracy. An unsupervised method to reduce dimensionality issues that combine t-SNE and a hierarchical neural network is suggested to detect the behavior of attacks [8]. By doing so, mapping high-dimensional network data space is shrunk into low-dimensional space. A method based on employing a distributed ensemble learning to cope with a misbehavior-aware on-demand collaborative intrusion detection system is proposed in [9], whose advantage is to reduce the number of life threats and road congestions caused by network attacks on VANET.

However, the research to cope with intrusion detection systems concerning significant data implementation issues has been at a developmental stage, so there is no longer a full-fledged method. Therefore, the research direction is on the track of improving the conventional ones toward the ones satisfying the necessities of big data.

The motivation of this research is summarized two-fold: increasing detection accuracy and adaptability to big data implementations by taking care of the single-point failure problems when the available distributed intrusion detection system model is a concern. Thus, a distributed intrusion detection system dealing with big data is proposed. In this effort, Hadoop's cluster computing environment [10] and its operational storage features are employed to utilize the Amazon DynamoDB database and Java architecture to design more robust intrusion detection components.

The rest of the manuscript is structured as follows: Section 2 presents the aspects of data such as data collector, transceiver, data analysis, and data-based alert systems. Section 3 is allocated to both results and discussion. Section 4 concludes the research by stressing the advantages of the proposed method.

2. The Proposed Method

Big data architecture is intended to manage data input, processing, and analysis that are too massive or complicated for typical database systems. The threshold at which businesses join the extensive data domain varies depending on the users' and tools' capabilities. It might imply hundreds of gigabytes of data for some and hundreds of terabytes for others. The definition of big data evolves as technologies for dealing with large datasets improve. This word increasingly refers to the value you can extract from your data sets using sophisticated analytics rather than the amount of the data, which tends to be pretty huge in many circumstances. The data landscape has evolved. What you can and are expected to do with data have shifted. Storage costs have dropped substantially, but the methods for collecting data have expanded. Some data comes at a quick speed, requiring ongoing collection and observation. Other data comes more slowly but in massive quantities, often in the form of decades of historical data. You might be dealing with advanced analytics or a cyber security dilemma. These are the problems that big data architectures aims to overcome.

Figure 1 depicts the logical components that comprise the proposed big data architecture. Its architecture has features: a data detector, data collector, agent, transceiver, and data analysis center.

It must be noted that there is not another comparable model to use as a pilot. Consequently, to avoid bias or incorrect impressions, the paper presents the performance of the proposed model in an innovative dataset. The experiments were made exclusively for this research approach and presented only in this paper. Every element in this design is included in the suggested individual solution, as discussed in the following sections.

2.1. The Detection of Data. A system coping with acquiring data and analyzing events as a unit located at the bottom of the system is called a data detector (DD). DDs are

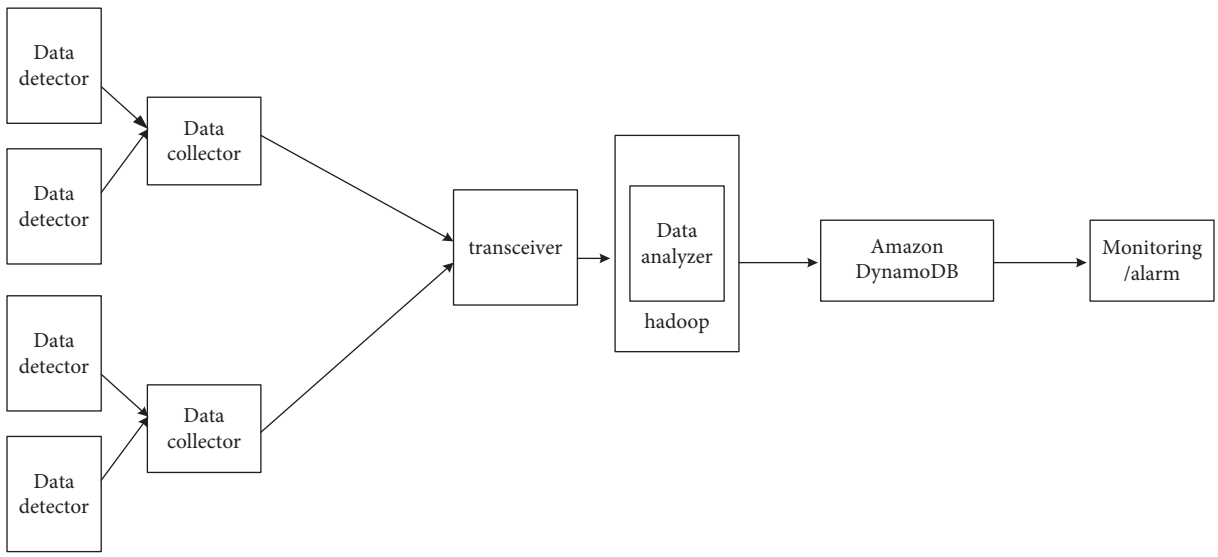


Figure 1: The structure diagram of the distributed intrusion detection system.

categorized into two types, host-based and network-based DDs, per different data resources. For general indexes such as central processing unit (CPU), memcachd (MEM) utilization, transmission control protocol (TCP) connections [11], and network bandwidth, a DD employs a capture service to determine indexes with a minimum interval length, then sends data to middleware, and finally returns it to the data center. For different indexes such as user behavior logs and WEB servers running records on other computers, a DD regularly utilizes a log monitoring service to capture the latest log information. Moreover, the system has no restrictions on the DD [12].

2.2. Data Collector. The DD is unique to each monitoring host. The data acquisition agent controls all local DDs. When the DD is in charge of transmitting data to the transceiver, it needs to send data to the agent and then the agent transmits it to the transceiver. The Chukwa of Apache software is employed as the data collector [13], which detects the contents of files written by DDs. If a new range exists in the file, it will be input into the collector of Chukwa following the specific rules and the collection will be input into Amazon DynamoDB [14].

The monitored host's CPU utilization, memory, TCP connections, and network bandwidth are collected. The DD, written in Java, employs SNMP-V3 based on the TCP convention family to handle the dispatching line. Then, the DD establishes a connection with the SNMP service through the SNMP driver package [4]. The reference is made to port 161 and obtains information by transmitting an analogous SNMP monitoring ID to the server. The workflow of the DD is composed of four sections. First, the DD is initialized and then linked to the SNMP service of the monitored machine [15]. Second, the data detector traverses each node in the SNMP service tree according to the OID set by the developer, and the stop condition is utilized to find the node with the same settings. Third, different threads are opened in the DD, and finally, each line grabs the data according to the minimum level and writes the captured data into the development file.

2.3. Data Transceiver and Middleware. Data transceiver divides the system monitoring network into multiple areas, and each site is composed of a group of transceivers and numerous data collectors. The operation of the data transceiver will sort out and process the received data [16]. A single-point failure may occur at one of these areas corresponding to a transceiver. Therefore, the transceiver adopts a redundancy strategy, and multiple transceivers can coexist in one area. When the data acquisition agent transmits data, it randomly selects a transceiver to transfer the data, which helps efficiently balance the available load [17]. The data middleware functions to cluster new information and loosen up the data analysis center to investigate information in the data. For this purpose, a message queue called RocketMQ is employed as the data receiving and sending middleware in this manuscript.

2.4. Data Analysis Center. The distributed intrusion detection system is constructed by the mode of "distributed detection and storage, and centralized analysis." While the monitoring host stores some intrusion or suspicious data at the bottom of the system locally and hoards the rest in the data analysis center, it utilizes the characteristics of suspicious events to pinpoint the intrusion behaviors that cannot be detected otherwise.

The Hadoop cluster framework, whose two types of nodes have specific roles, is implemented [18]. While one is called the common computing node, the other is called the task submission node. The task submission node will periodically submit data to analyze tasks in a cluster. Then, the corresponding process is described as follows: the number of partitioned tasks is firstly computed based on data size. Then, each task is partitioned and distributed to each cluster node to make calculations. If the calculation task determines specification steps, the partition outcomes of each task will

be summarized, calculated, and presented. Otherwise, the results will be provided as output directly. The Hadoop framework deals with the whole process by allocating and scheduling tasks and recovering errors. Thus, users only define computational tasks of MapReduce, processing methods used for data fragmentation, and specification methods.

The proposed protocol has an intriguing quirk: the reducers never directly communicate with one another, but only through the mappers in the following round. MapReduce handles grouping and message passing, along with engineering challenges like fault tolerance or load balancing, which are all controlled by MapReduce. The proposed mathematical protocol's most significant suggestion is to encode a limit on the total amount of space consumed. Specifically, the algorithm takes a key-value pair list as input:

$$k_i, v_{i,i=1}^{N}. \tag{1}$$

As a whole size,

$$n = \sum_{i=1}^{N} |k_i| + |v_i|. \tag{2}$$

The mapper m is a Turing machine that takes a single key-value pair k, v as input and returns a list of key-value pairs as output:

$$k_1', v_1', \ldots, k_s', v_s'. \tag{3}$$

If $\mathrm{MRC}[f(n), g(n)]$ is the round bound and the second argument is the time bound, the logarithmic number of rounds is defined as

$$\mathrm{MRC}^i = \mathrm{MRC}\left[\log^i(n), \mathrm{poly}(n)\right]. \tag{4}$$

On the other hand, the mapper ρ is a Turing machine that takes a single key-value k as input and a list of values $v_1, \ldots, v_m$ and produces a new list as $v_1^{'}, \ldots, v_M^{'}$.

Each reducer determines what the finishing state would be if the G is a graph that had started in state s after processing the chunk of the input for each conceivable state s in G. As a result, the output of reducer j would be an encoding of the following table:

$$\begin{aligned} &s_1 \longrightarrow T_j(s_1), \\ &s_2 \longrightarrow T_j(s_2), \\ &\vdots \\ &s_{|S|} \longrightarrow T_j\left(s_{|S|}\right). \end{aligned} \tag{5}$$

If

$$\mathrm{MRC}[\mathrm{poly}(n), 1] \subsetneq \mathrm{MRC}[\mathrm{poly}(n), \mathrm{poly}(n)], \tag{6}$$

and

$$\mathrm{MRC}[1, n] \subsetneq \mathrm{MRC}[n, n] \subset \mathrm{MRC}\left[1, n^2\right] \subsetneq \mathrm{MRC}\left[n^2, n^2\right] \subset \ldots. \tag{7}$$

The process is

$$\mathrm{MRC}[1, \mathrm{poly}(n)] = \mathrm{MRC}[\mathrm{poly}(n), \mathrm{poly}(n)]. \tag{8}$$

The size (number of edges) and storage space needed in the computer memory to neighborhood list of a graph (number of nodes) are

$$\left(\sum_{i=1}^{n} (1 + \deg(v_i))\right) = O(n + m). \tag{9}$$

We have

$$\deg(v_i) = d_{\mathrm{out}}(v_i), 1 \le i \le n. \tag{10}$$

Therefore, the total complexity required to implement the algorithm is

$$O(n) + \sum_{v \in V} O(|\mathrm{Adj}(v)|) = O(n + m). \tag{11}$$

The cost function we want to minimize during the reduction process is as follows:

$$\frac{1}{2}\|w\| + C\sum_{i=1}^{n} \log\left(\exp\left(-y_i\left(w^T x_i + b\right)\right) + 1\right), \tag{12}$$

where $C > 0$ and b are the coefficients representing the penalty of incorrect results.

So, to be able to estimate the probability distribution of the process, we limit the history to n processes:

$$P(w_T | w_1, w_2, \ldots, w_{T-1}) \approx P(w_T | w_{T-n}, \ldots, w_{T-2}, w_{T-1}). \tag{13}$$

Through maximum likelihood estimation, we calculate

$$P(w_3 | w_1, w_2) = \frac{\mathrm{count}(w_1, w_2, w_3)}{\sum_w \mathrm{count}(w_1, w_2, w)}. \tag{14}$$

Therefore,

$$\begin{aligned} x &= [C(w_1); C(w_2); \ldots; C(w_n)]. \\ \hat{y} &= P(w_i | w_{1:k}) = LM(w_{1:k}) = \mathrm{softmax}\left(hW^2 + b^2\right). \\ h &= g\left(xW^1 + b^1\right). \\ x &= [C(w_1); C(w_2); \ldots; C(w_n)]. \\ C(w) &= E_{[w]}. \end{aligned} \tag{15}$$

where $w_i \in V, E \in \mathbb{R}^{|V| \times d_w}, W^1 \in \mathbb{R}^{n \cdot d_w \times d_{hid}}, b^1 \in \mathbb{R}^{d_{\mathrm{hid}}}, W^2 \in \mathbb{R}^{d_{\mathrm{hid}} \times |V|}, b^2 \in \mathbb{R}^{|V|}$.

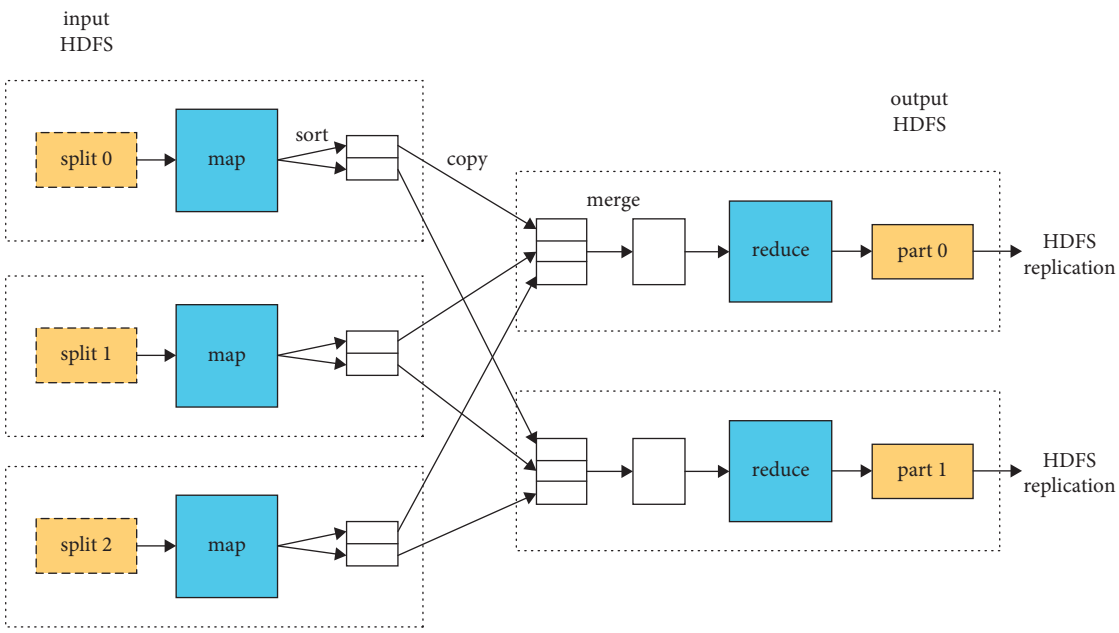


Figure 2: MapReduce processing.

Finally, we propose a method that first linearly increases the learning rate and then decreases it linearly for reducers. The proposed algorithmic system is as follows:

$$\text{cut} = [T \cdot \text{cut\_frac}],$$

$$p = \begin{cases} \dfrac{t}{\text{cut}}, & \text{if } t < \text{cut}, \\ 1 - \dfrac{t - \text{cut}}{\text{cut.}\ (1/\text{cut\_frac} - 1)}, & \text{else}, \end{cases} \tag{16}$$

$$\eta_t = \eta_{\max} \cdot \frac{1 + p \cdot (\text{ratio} - 1)}{\text{ratio}}.$$

The idea is to adapt the parameters to the characteristics of a particular set of processes. The model should first converge quickly to a suitable area of the parameter space and then improve the parameters.

2.5. Monitoring of the System and Alarming Service. The monitoring system mainly surveils the CPU, MEM, TCP connection, network bandwidth, and other fundamental sections of each host, as well as the running status of the monitoring host [19]. The external invasion trace can be determined during the process, and the corresponding measures can be taken when data detection is utilized [20]. The system has an alert facility located at the highest stage, and its function is to judge whether the system is within the normal operation bounds. The alarm service receives information from the data analysis [21] center and monitoring system. When there would be abnormal data [22] or an aberrated host running state, the alarm service would send an alarm to inform the administrator that the system could be under attack [23].

The spring MVC framework and velocity template technology are employed to implement the module to monitor the system. This module in charge of surveilling is composed of modules to allocate the management of a user, monitor the system, and manage Amazon DynamoDB [24].

The login page of the user management module accepts information in a template form. Whenever information is input to log in, the token must be verified [25]. A user verified as logged in by the token directly reaches the home page. In contrast, the receipt confirms the login information, which results in successful verification. Then, it will jump into the login interface directly. Otherwise, it will be directed to the error interface.

The monitoring module contains two functions: index view and index definition. While JavaScript and velocity technology implement the index view function, the index definition function is implemented by the JavaScript Hight Charts drawing function library. The Amazon DynamoDB management module is employed to manage online Amazon DynamoDB to reduce the management complexity [26].

3. Results and Discussion

The DD module in this manuscript's distributed intrusion detection system grabs data at the minimum level. The CPU utilization rate captured by the DD module was gauged between 0:00 to 6:00 am on December 5, to depict the operation condition of the CPU system, which is shown in Figures 2 and 3.

Currently, the main security problem in implementing big data services has been distributed denial of service (DDoS) attack, which is taken as the research object. It must be noted that while firewalls and intrusion prevention systems (IPS) are essential for network security, they are insufficient to guard against complicated DDoS assaults. Modern DDoS attack tactics need a multifaceted strategy

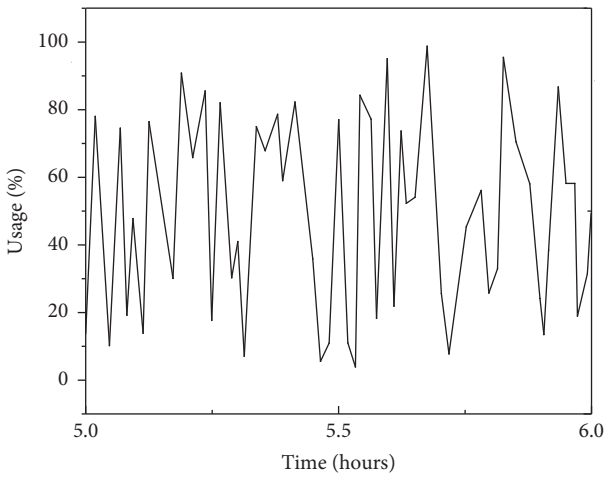


Figure 3: The figure plot showing the trend of the CPU index (Server 1).

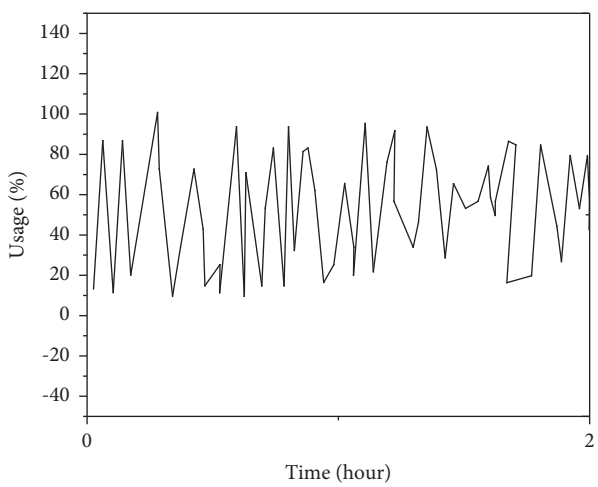


Figure 4: The figure plot showing the trend of the CPU index (Server 2).

that allows users to examine Internet infrastructure and network availability. Consider the following capabilities for improved DDoS protection and quicker mitigation of TCP SYN flood DDoS attacks:

(1) Support for both inline and out-of-band deployment to guarantee the network has no single point of failure

(2) It has broad network visibility, including the ability to observe and analyze traffic from many networks segments

(3) Various threat information sources, such as statistical anomaly detection, customized threshold alerts, and fingerprints of known or new threats, ensure rapid and precise detection

(4) There is scalability to handle assaults of various sizes, from low end (e.g., 1 Gbps) to high end (e.g., 10 Gbps and 40 Gbps)

The offense instrument of the DDoS is utilized to propose the distributed intrusion system.

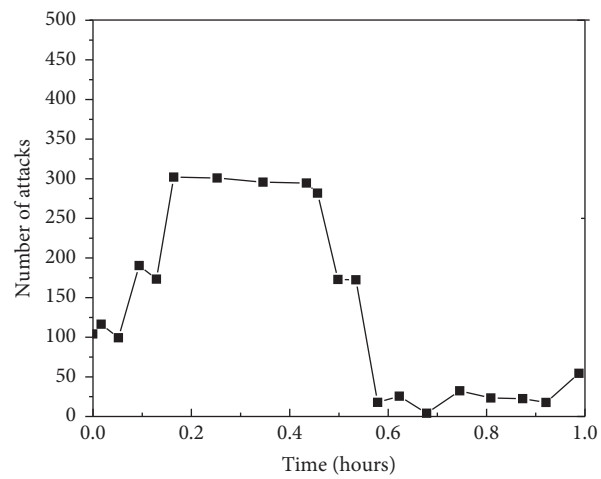


Figure 5: The figure plot depicts the changing trend of the CPU when the system is underattacked or invaded.

Figure 4 depicts that the number of TCP connections fluctuates from 0 to 100 between 0 and 20 minutes, and the changing trend is found to be relatively stable, proving that the system has not received attacks.

In Figure 5, between 20 through 40 minutes, the DDoS attacks are launched on the monitored system and the number of TCP connections increases to 300. Therefore, external attacks and intrusions can be detected through the monitoring system.

4. Conclusion

The Java system is employed to devise and execute a distributed intrusion detection framework by implementing the Hadoop framework when big data is considered. Thus, a novel protection method is put forward. The proposed method bringing advantages to the literature can be summarized as follows: (1) Distributed data acquisition, distributed processing, and distributed analysis are realized. (2) Monitoring CPU, MEM, and TCP indexes of the controlled host, external attacks, and intrusions are well detected. The corresponding alert services are provided when the DD collaborates with the data collector, transceiver middleware, and the center responsible for analyzing data.

The proposed method resolves the issues related to single-point failure and the low operating efficiency of the original distributed intrusion system. However, it still has some limitations. In the future research agenda, the combination with the machine learning algorithm would potentially contribute to the proposed intrusion detection methods, a capability that improves the ability of self-learning and adaptivity of the system. In conclusion, the most critical aspect will be reached by having both efficient and precise detection accuracy.

MapReduce is a programming framework, not an algorithm in and of itself, and complexity analysis is usually reserved for algorithms. But, a future expansion of the proposed approach will be the complexity analysis of MapReduce operations by getting the appropriate variables, for example,

$$O\left(n\log n * s * \left(\frac{1}{p}\right)\right), \tag{17}$$

where n is the number of items, s is the number of nodes, and p is the ping time between nodes (assuming equal ping times between all nodes in the network).

Data Availability

Data will be provided upon request to the author.

Conflicts of Interest

The author declare no conflicts of interest.

References

[1] D. Gonzalez-Cuautle, A. Hernandez-Suarez, G. Sanchez-Perez et al., "Synthetic minority oversampling technique for optimizing classification tasks in botnet and intrusion-detection-system datasets," *Applied Sciences*, vol. 10, no. 3, p. 794, 2020.

[2] L. Li, Y. Yu, S. Bai, Y. Hou, and X. Chen, "An effective two-step intrusion detection approach based on binary classification and k -nn," *IEEE Access*, vol. 6, no. 3, pp. 12060–12073, 2018.

[3] Y. Gao, X. Chen, and X. Du, "A big data provenance model for data security supervision based on PROV-DM model," *IEEE Access*, vol. 8, pp. 38742–38752, 2020.

[4] M. M. Rathore, H. Son, A. Ahmad, A. Paul, and G. Jeon, "Real-time big data stream processing using GPU with spark over Hadoop ecosystem," *International Journal of Parallel Programming*, vol. 46, no. 3, pp. 630–646, 2018.

[5] M. Li, R. Li, and P. P. C. Lee, "Relieving both storage and recovery burdens in big data clusters with R-STAIR codes," *IEEE Internet Computing*, vol. 22, no. 4, p. 1, 2017.

[6] K. Singh, S. C. Guntuku, A. Thakur, and C. Hota, "Big data analytics framework for peer-to-peer botnet detection using random forests," *Information Sciences*, vol. 278, pp. 488–497, 2014.

[7] K. Logesh and D. Sumathi, "Journal of critical reviews K-means algorithm based distributed intrusion detection system for cloud computing environment," *Journal of Critical Reviews*, vol. 7, no. 15, 2020.

[8] H. Yao, C. Li, and P. Sun, "Using parametric t-distributed stochastic neighbor embedding combined with hierarchical neural network for network intrusion detection," *International Journal on Network Security*, vol. 22, no. 2, pp. 265–274, 2020.

[9] F. A Ghaleb, F. Saeed, M. Al-Sarem et al., "Misbehavior-aware on-demand collaborative intrusion detection system using distributed ensemble learning for VANET," *Electronics*, vol. 9, no. 9, p. 1411, 2020.

[10] Y. Fu, N. Xiao, H. Jiang, G. Hu, and W. Chen, "Application-aware big data deduplication in cloud environment," *IEEE Transactions on Cloud Computing*, vol. 7, no. 4, pp. 921–934, 2019.

[11] P. Sadotra and C. Sharma, "A new distributed intrusion detection system in computer network: an approach to detect malicious intrusion threats at initial stage," *Oriental Journal of Computer Science and Technology*, vol. 10, no. 2, pp. 326–332, 2017.

[12] O. Achbarou, M. A. E. Kiram, S. Elbouanani, and Y. Xie, "A new distributed intrusion detection system based on multi-agent system for cloud environment[J]," *International Journal of Communication Networks and Information Security*, vol. 10, no. 3, pp. 526–533, 2018.

[13] S. R. Khonde and V. Ulagamuthalvi, "Ensemble-based semi-supervised learning approach for a distributed intrusion detection system," *Journal of Cyber Security Technology*, vol. 3, no. 3, pp. 163–188, 2019.

[14] S. Khonde and U. Venugopal, "Hybrid architecture for distributed intrusion detection system," *Ingénierie des Systèmes d'Information*, vol. 24, no. 1, pp. 19–28, 2019.

[15] J. Liu, W. Zhang, T. Ma et al., "Toward security monitoring of industrial Cyber-Physical systems via hierarchically distributed intrusion detection," *Expert Systems with Applications*, vol. 158, Article ID 113578, 2020.

[16] S. Singh and A. Yassine, "Big data mining of energy time series for behavioral analytics and energy consumption forecasting," *Energies*, vol. 11, no. 2, p. 452, 2018.

[17] C. Yudong and C. Yuejie, "Harnessing structures in big data via guaranteed low-rank matrix estimation[J]," *IEEE Signal Processing Magazine*, vol. 35, no. 4, pp. 14–31, 2018.

[18] K. Yang, Q. Yu, and S. Leng, "Data and energy integrated communication networks for wireless big data[J]," *Entia Sinica*, vol. 4, pp. 713–723, 2018.

[19] Y. Jing, Y. Bian, Z. Hu, L. Wang, and X.-Q. S. Xie, "Deep learning for drug design: an artificial intelligence paradigm for drug discovery in the big data era," *The AAPS Journal*, vol. 20, no. 3, p. 58, 2018.

[20] B. Hong, H. Wang, and Z. Cao, "An effective fault-tolerant intrusion detection system under distributed environment," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–9, Article ID 2716881, 2021.

[21] S. Demertzis, V. Demertzi, and K. Demertzis, "Data analytics for climate and atmospheric science," *International Journal of Big Data Mining for Global Warming*, vol. 03, no. 01, Article ID 2150005, 2021.

[22] Y. Bian and X. Tang, "Abnormal detection in big data video with an improved autoencoder," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–6, Article ID e9861533, 2021.

[23] P. Gattineni and G. R. S. Dharan, "Intrusion Detection Mechanisms: SVM, random forest, and extreme learning machine (ELM)," in *Proceedings of the 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 273–276, September 2021.

[24] A. Cuzzocrea, "Big data lakes: models, frameworks, and techniques," in *Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1–4, NY City China, Janury 2021.

[25] G. Ra, T. Kim, and I. Lee, "VAIM: verifiable anonymous identity management for human-centric security and privacy in the Internet of things," *IEEE Access*, vol. 9, pp. 75945–75960, 2021.

[26] Z. Lijun, H. Guiqiu, L. Qingsheng, and D. Guanhua, "An intuitionistic calculus to complex abnormal event recognition on data streams," *Security and Communication Networks*, vol. 2021, pp. 1–14, 2021.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 9922167, 7 pages
https://doi.org/10.1155/2022/9922167

Research Article

Increasing Cyber Defense in the Music Education Sector Using Blockchain Zero-Knowledge Proof Identification

Ying Zhang

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Ying Zhang; zhangying198002@126.com

Received 4 May 2022; Revised 23 May 2022; Accepted 26 May 2022; Published 28 June 2022

Academic Editor: Konstantinos Demertzis

Music creation and its promotion are encouraged both in music education and through activities organized in the context of artistic creation as part of the education in question. Although copyright registration is the primary way authors protect their rights, this is not feasible in most cases, as the processes take a long time to complete and incur high costs. We utilize modern innovative technologies and their developments in copyright protection matters to increase security and trust in music education. In particular, an advanced model of ensuring the methods and innovation produced in music education processes is proposed, using blockchain technology and smart contracts. But given that, even in an advanced system like the proposed one, authentication evidence can be easily intercepted, this work proposes a single and robust identification scheme based on an innovative zero-knowledge proof (ZNP) system, which allows one side of communication to convince the other of its validity.

1. Introduction

In recent years, cloud computing has been widely used in many areas of everyday life, mainly for data storage [1, 2]. This raises questions about the reliability and how to manage the data in question. The multitude of these services targets widespread attacks by third parties. These attacks find fertile ground as they exploit security vulnerabilities, resulting in data leaks [3]. The result is that both the security and the privacy of the data stored in cloud services are questioned [4, 5]. In addition, users' data is often used for exploitation purposes or given to third parties such as advertising companies. Another deterrent is that data providers usually store that information without encryption, making user data easily accessible [6].

An attractive solution that can give another approach to the issue is blockchain, which proposes a decentralized and highly secure solution for data storage [7]. Similarly, the blockchain can be used as an intermediary whose primary function is to maintain and validate actions within the chain. In general, blockchain implements a distributed global platform that runs smart contracts [8], utilizes proven technologies, and has an architecture that allows various additional functions to be implemented simply and transparently [9]. It enables the creation of different security levels and licenses only certified users to access specific services or resources. Due to the encryption [10, 11] of transactions and its operating environment, it is ideal for environments that require reliability without the mediation of third-party trusted entities, as it can achieve complete confidentiality of transactions and selective access between participants only to licensed information. This achieves the confidence of the participants in the sharing of information, combined with all the benefits of blockchain [12, 13].

This function is advanced further by using blockchain-provided smart contracts, which allow access to information under precise, strictly specified, and preagreed-upon conditions [8, 14]. When unavoidable circumstances are met, these contracts will close deals. It is simply a protocol designed to digitally facilitate, verify, or enforce the negotiation or execution. These contracts enable the performance of trustworthy transactions without the involvement of third parties, with the transactions being secure, monitored, and irreversible [15, 16]. They seek to protect the scope of contract law while also reducing the additional processing expenses involved with the award and implementation of

intermediary contracts [17]. Blockchain implementation is based on the Byzantine Fault Tolerance (BFT) consent algorithm [18, 19], which means that its command service must be jointly controlled by network members. Using the BFT algorithm, the standard guarantees coverage or the ability to reach consensus, even if there are rival nodes (malicious) or if the nodes are offline [20].

In this paper, we present an enhanced model that uses blockchain technology and smart contracts to guarantee the approaches and innovation achieved in music education. Given that, even in a progressive system like the one that is being proposed, the proposed approach is that the authentication items can be easily copied, guessed, or revealed by automated methods and technical means, the main contribution of this work is to offer a single and robust identification scheme, which is based on an innovative ZNP system [12], which enables one side of communication to convince the other side of its validity.

The following is the structure of the paper. In the following section, an overview of the several appropriate methods that have been identified in the relevant literature is presented. In Section 3, we will discuss the ZNP protocol that has been delivered. In Section 4, the scenarios and results prove that ZNP and complexity exist. In the final part of the report, Section 5, a summary of the findings and a list of possible following study directions are presented.

2. Related Literature

The literature on blockchain technologies, smart contracts, and zero-proof knowledge is becoming more and more frequent since these innovative technologies are gaining confidence in the community [21].

Hu et al. [14] gave a detailed study of smart contracts, emphasizing current applications and the problems they confront. They introduced the idea of a blockchain-based smart contract, a digital software meant to enable the settlement or contract terms immediately among users when specific circumstances are satisfied. With the improvement in blockchain technology, smart contracts are being utilized to fulfill a wide variety of objectives, from self-maintained accounts on public blockchains to automating corporate collaboration on blockchain systems.

On the other hand, Wang et al. [8] provided a systematic and extensive assessment of blockchain-enabled intelligent contracts to motivate more study in this developing research field. Following the introduction of the operational mechanism and mainstream platforms for blockchain-enabled smart contracts, they proposed a scientific framework for smart contracts based on a groundbreaking six-layer design, which was accepted by the scientific community. Second, the technological and legal difficulties, as well as current research advances, were detailed. Thirdly, they discussed some representative application cases. They concluded by debating the future development patterns for smart contracts.

Yang and Li [12], employing smart contracts and zero-knowledge proof methods to create identity unlikability, have successfully avoided the disclosure of attribute ownership in the present claim identification model on the blockchain. Aside from that, they created a system prototype known as BZDIMS, which features a challenge-response protocol that allows users to reveal their ownership of characteristics to service providers, thereby maintaining the privacy of their activities. Performance and security analyses demonstrated that their system provided good attribute privacy security and a broader application breadth than the previous paradigm.

Sankar et al. [17] examined and compared the viability and efficiency of blockchain consensus algorithms. The consensus protocol is at the heart of blockchain technology. Academics are eager to design a well-optimized Byzantine fault-tolerant consensus system in light of the advent of new possibilities in blockchain technology. Exciting options include developing a worldwide consensus protocol or creating a cross-platform plug-and-play software application to support a variety of consensus mechanisms. Incorporating the principles of quorum slices and federated Byzantine Fault Tolerance, the Stellar Consensus System is a global consensus protocol designed to be fault-tolerant and claims to be Byzantine Fault Tolerance. Additionally, the hyperledger is an open-source project led by the Linux Foundation that focuses on realizing the notion of realistic Byzantine Fault Tolerance and providing a framework for the plug-and-play deployment of many different consensus protocols and chain applications.

Finally, Buchman [18] developed Tendermint, a novel protocol for organizing events in a dispersed network under adversarial circumstances, as part of his examination of Byzantine Fault Tolerance. Known more frequently as unanimous agreement or atomic broadcast, the problem has gained significant attention in recent years because of the widespread growth of digital currencies such as Bitcoin and Ethereum, which effectively remedy the issue in public settings without the intervention of a central authority. Their concept modernized previous academic work in the field by providing a safe consensus mechanism with accountability requirements and functionality for creating arbitrary applications atop the consensus. Their idea is a high-performance blockchain, capable of processing several events per second over dozens of nodes scattered across the world, with a latency of less than one second and performance deteriorating very slightly in the face of hostile assaults.

3. Proposed ZNP Protocol

Entering a service electronically involves different authentication methods. It often requires repetition of the same information or distinct numbers and codes, which can be easily intercepted or revealed [22]. The service provider usually keeps a summary of each user's password. Each time the user wants to connect to the service, the password is given in the summary function, and the result is compared to the saved one [23]. This protocol may not allow the password to be saved in its original form, but the server temporarily learns it [24, 25]. This process could be replaced with a ZNP indicating that each customer owns the password [12].

Although it has offered us many benefits, including openness, immutability, and decentralization, blockchain

technology may not provide the necessary level of anonymity for certain types of transactions. However, integrating blockchain technology with ZNP has the potential to deliver to customers a potent combination of immutability and security. A ZNP is a sort of cryptography that allows one person (the prover) to demonstrate to another party (the verifier) that certain information is accurate without giving any extra information. When it comes to messaging applications, end-to-end encryption has been a significant factor in developing private message transmission. On the other hand, traditional messaging applications demand that users verify their identities on a central server. Individuals can demonstrate their identity using ZNPs without divulging any more personal information.

In the proposed ZNP, each calculation is performed by exchanging messages between an entity called prover (P) and an entity called verifier (V). Typically, P wants to convince V that a proposition is true (witness). P and V are probabilistic Turing machines, where P has unlimited computing power while V is limited to probabilistic calculations of polynomial complexity [26].

Zero knowledge is realized, given that V learns nothing more than the fact that P's claim is valid [27]. A key role in proving that an interactive system has the property of zero knowledge is played by the simulator (S), which simulates P but does not have access to the witness. His contribution is as follows [28]: V interacts with S. At some point, V will put S in the "difficult position" of not being able to answer a question as he does not have access to the witness. In this case, we return the V tape to a state before rewinding and running the protocol from that point on. If V (with continuous rewinds) finally accepts S's proof, the protocol holds the status of zero knowledge, as V cannot distinguish a P who knows the witness and an S who pretends. V cannot export any additional information from the protocol (since, in the second case, there is no information to ship) [29, 30].

Let an NP language L and M be a polynomial Turing machine such that [31, 32]

$$x \in L \Leftrightarrow \exists w \in \{0,1\}^{p(|x|)}: M(x,w) = 1, \tag{1}$$

where p is a polynomial. One proof of zero knowledge for L is two possible Turing Polynomial Time (TPT) machines P and V for which the following three properties apply [33]:

(1) Completeness: if $x \in L$ and w are a witness to this, that is,

$$M(x,w) = 1, \tag{2}$$

then

$$\Pr\left[\text{out}_{\mathscr{V}} < P(x,w), V(x) > (x)\right] = 1 \geq \frac{2}{3}, \tag{3}$$

where

(a) $P(x,w), V(x)$ is the interaction between P and V with standard (public input) x and private input of P at w.

(b) $\text{out}_{\mathscr{V}}$ is the output V at the end of the protocol.

(2) Correctness: if $x \notin L$, then

$$\forall (P^*, w) \Pr\left[\text{out}_{\mathscr{V}} < P^*(x,w), V(x) > (x)\right] = 1 \leq \frac{2}{3}.OP^*, \tag{4}$$

where P^* does not need to be TPT.

(3) Validity: V does not accept false statements (even if P tries to trick him).

The proposed model appears to be related to the NP complexity class in the above definition [34–37].

4. Evidence of ZNP and Complexity

To prove the proposed ZNP methodology, we will use three different examples which show its power as a computational and cryptographic model which can respond to the proposed implementation [15, 38].

4.1. Graph Isomorphism. The first example concerns the isomorphism of graphs. Specifically, two isomorphic graphs where the mapping from ABCD to CDAB corresponds to the first graph to the second, as shown in Figure 1.

Two graphs, $G1$ and $G2$, are said to be isomorphic if they have the same number of vertices. There is a shift, that is, function 1–1 and on, between their nodes such that two nodes of one are connected by acne if and only if the corresponding nodes of the other are connected by acne. Equivalently, there is a renaming of the nodes of a graph such that the charts are identical. The problem of graph isomorphism belongs to the NP class, but it is not known whether it is NP-complete or not [39]. Assume that both P and V know the graphs $G1$ and $G2$; that is, the latter is a common input of the protocol. In addition, P knows the isomorphism between them ϕ: $G1 \longrightarrow G2$ (private input of V or the witness mentioned above). Using a zero-knowledge protocol, he can prove that he knows the isomorphism without revealing it [40]:

(1) P randomly selects one of $G1$, $G2$, and Gi. By some permutation ψ of the vertices of Gi, P produces the graph $H = \psi(Gi)$, which is isomorphic with Gi. Because P knows the isomorphism ψ between H and Gi, he also knows the isomorphism $\psi\phi$ between H and $G3 - i$. Anyone else has as much difficulty finding an isomorphism between H and $G1$ or between H and $G2$ as finding an isomorphism between the initials $G1$ and $G2$.

(2) P binds to ψ, sending H to V.

(3) V randomly selects a graph from $G1$, $G2$, and Gj and sends his selection as a challenge to P, asking him to prove that H and Gj are isomorphic. That is, he asks for a permutation of Gj to produce H.

(4) P responds by doing the following:

$(\widehat{I} \pm I')$if $G_i = G_j$, send to V the permutation ψ.
$(\widehat{I}^2 I')$if $G_i \neq G_j$, then we have the following:

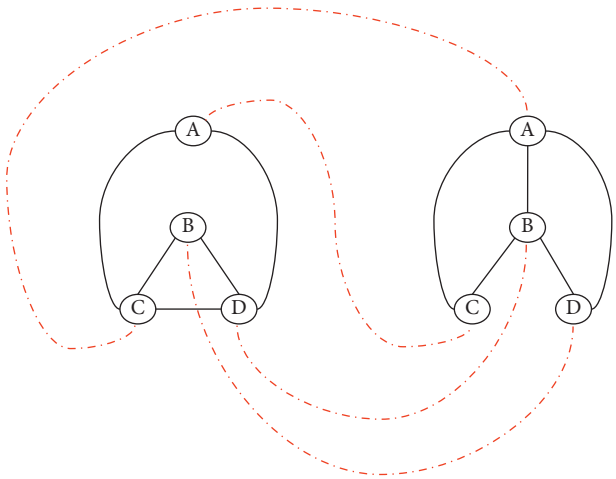


Figure 1: Two isomorphic graphs where the mapping from ABCD to CDAB corresponds to the first graph to the second.

(i) If $G1$ and $G2$ are isomorphic (then $\exists\rho$: $G_i = \rho(G_j)$), send to V the permutation $\psi\rho$
(ii) If $G1$ and $G2$ are not isomorphic (i.e., P is not honest), then it cannot find a suitable permutation and sends any random permutation

(5) If V receives a correct permutation, he continues (repeat steps 1–5); otherwise, he stops rejecting (i.e., he considers that the graphs are not isomorphic).

If V has not rejected after k repetitions of steps 1–5, he accepts (considers the graphs isomorphic). The above protocol fulfills the properties of the zero knowledge mentioned above. First, it is complete because if there is an isomorphism between $G1$ and $G2$, then P will convince V with a probability of 1 (V never rejects) [27, 41].

Regarding correctness, if there is no isomorphism, then P has a 1/2 chance at each step to deceive V (this will only happen if $Gi = Gj$). After k repetitions, this probability becomes $1/(2^k)$.

V does not get any additional information regarding the isomorphism between $G1$ and $G2$ regarding the zero knowledge. When interacting with S, his first step will be the same as P; that is, he will make a new random graph isomorphic to one of $G1$ and $G2$ each time. The probability of choosing either $G1$ or $G2$ is precisely 1/2. So, at this stage, V cannot separate them. Thus, the likelihood of cheating in k repetitions remains $1/(2^k)$. So, the expected execution time is polynomial as it results from the relation [15, 39, 42]:

$$T_{\mathscr{V}} \frac{\sum_{k=1}^{\infty} 1}{\left(2^k\right)} = T_{\mathscr{V}}, \tag{5}$$

where $T_{\mathscr{V}}$ is the execution time of V, which is polynomial.

4.2. 3-Coloring. A zero-knowledge protocol for an NP-complete problem would mean that all NP problems have zero-knowledge protocols [12, 26, 32, 39]. In the NP-complete problem of 3-Coloring, P knows a coloring c for a graph $G = (V, E)$ such that [43]

$$c\colon V \longrightarrow \{1, 2, 3\} \text{and}\, c(v_1) \neq c(v_2) \Leftrightarrow (v_1, v_2) \in E. \tag{6}$$

He wants to prove this knowledge to V without revealing c:

(1) P selects a random permutation π of {1, 2, 3}. From this, an alternative 3-$\pi \cdot c$ of G then uses a commitment scheme for $\pi.c$, that is, calculates values commit$((\pi.c)(v_i), r_i), \forall v_i \in V$ and sends them to V.
(2) V selects a random edge $(v_i, v_j) \in E$ and sends it to P.
(3) P releases the values $\pi \cdot c(v_i), \pi \cdot c(v_j)$ and sends them to V.
(4) V checks if $\pi \cdot c(v_i) \neq \pi \cdot c(v_j)$.

It is evident that the above protocol is complete. Regarding the correctness, we observe that if P does not have a valid 3-color, then V will choose an edge with the same peak colors with probability $1/|E|$. By repeating the protocol, we can make the probability that prover $1 - 1/|E|$ cheats him extremely small. About zero knowledge, even S does not have a valid coloring. If V chooses an edge with the same peak colors, then it rewinds to a previous state, and S selects a new random permutation that it uses in the new execution. It can be shown that the protocol with S does not have an expected execution time of a different order of magnitude than with P and V does not understand the difference. So, the protocol has the property of zero knowledge [15, 42].

4.3. Noninteractive Proof of Zero Knowledge. To make a noninteractive proof, we use a hash function [26, 39]:

$$H\colon \{0, 1\}^* \longrightarrow Z_q, \tag{7}$$

such that the discussion

$$(y, c, s) = g^t, H\left(g^t\right), t + H\left(g^t\right) w \bmod q. \tag{8}$$

Assume that H is a random oracle controlled by the simulator to demonstrate that ZNP holds. In the random oracle model, a nonhonest verifier V can ask questions of the random oracle and receive answers. In this case, c is forced to be selected after y, which is directly dependent on the characteristics of the hash function [12, 13]. Figure 2 shows how V^* interacts with H.

When the verifier asks for the proof of $h = g^w$, the simulator randomly selects c and s to compute $y = g^s h^{-c}$. Set (y, c) in History and return $\langle y, c, s\rangle$. The nonhonest verifier cannot separate an honest prover from an emulator unless $(y, c')\in$ History with $c \neq c'$. Then, V^* achieves with probability $(1/m)q_H$, where q_H is the number of questions in the random oracle. Then, we want to produce two discussions that end in acceptance with the same y but with different challenge values. Using these two discussions, we can extract a witness. Note that $c = H(y)$. If a dishonest prover P^* asks a unique question in the random oracle before producing $\langle y, c, s\rangle$, the resolution is the same as the interactive protocol. Problems arise when P^* asks more than one question [31, 38, 39].

Assume that, in the first round, P^* asks q_H questions before ending the discussion. The knowledge exporter then

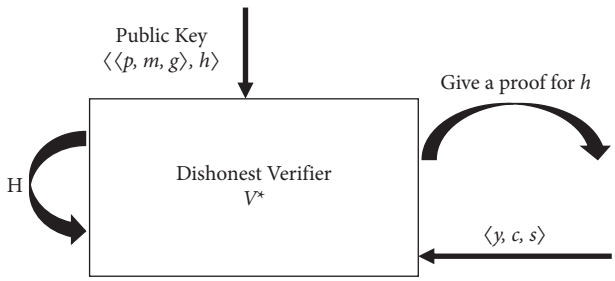


Figure 2: In the random oracle model, the simulation of dishonest verifier V^* is performed.

returns P^* to a previous step, with no guarantee that P^* will request q_H questions again. When P^* finishes, it will return y', c', s' with $c' = H(y')$ and possibly $y \neq y'$. This limits our capacity to compel a witness to testify, so we should modify the probability of having two acceptance discussions with the same y.

Assume that, after asking q_H questions, P^* selects a question he asked and uses the corresponding answer he got for it from the random oracle at his exit. Let $\text{Prob}[A] = \alpha$ be the probability that the discussion will end in acceptance. Let $\text{Prob}[Q_i] = \beta_i$ be the probability that the dishonest prover uses the i-th answer c_i, in which $1 \leq i \leq q_H$. We define $\text{Prob}[A \cap Q_i] = \alpha_i$. Respectively, for the repetition of the experiment, we write A', Q_j', c_i'. Then, it is valid [15, 29, 30, 42]:

$$\sum_{i=1}^{q_H} \alpha_i = \alpha \text{ and } \sum_{i=1}^{q_H} \beta_i = 1. \tag{9}$$

We define $\text{Prob}[E]$ as the probability of extracting a witness from P^*, and we have

$$\text{Prob}[E] = \text{Prob}\left[A \cap A' \cap (i = j) \cap \left(c_i \neq c_j'\right)\right]. \tag{10}$$

Similarly, we have

$$\text{Prob}[E] \geq \text{Prob}\left[A \cap A' \cap (i = j)\right] - \text{Prob}\left[\left(c_i = c_j'\right)\right]. \tag{11}$$

So,

$$\begin{aligned}\text{Prob}[E] \geq \text{Prob}\left[A \cap A'\right] - \frac{1}{q} &= \sum_{i=1}^{q_H} \text{Prob}\left[A \cap Q_i \cap A' \cap Q_i'\right] - \frac{1}{q} \\ &= \sum_{i=1}^{q_H} \text{Prob}\left[A_i \cap A_i'\right] - \frac{1}{q}.\end{aligned} \tag{12}$$

From the definition of exporter in our calculations, we know that

$$\text{Prob}\left[A_i \cap A_i'\right] \geq \frac{\text{Prob}\left[A_i\right]^2}{4} = \frac{\alpha_i^2}{4}. \tag{13}$$

The total probability is calculated as follows:

$$\text{Prob}[E] \geq \sum_{i=1}^{q_H} \text{Prob}\left[A_i \cap A_i'\right] - \frac{1}{q} = \frac{1}{4} \sum_{i=1}^{q_H} \alpha_i^2 - \frac{1}{q}. \tag{14}$$

From the statistics, we know that

$$\frac{\sum\left(\alpha_i^2\right)}{q_H} \geq \left(\frac{\alpha}{q_H}\right)^2. \tag{15}$$

And so,

$$\sum\left(\alpha_i^2\right) \geq \frac{\alpha}{q_H}. \tag{16}$$

And for any real α_i, they have an average:

$$\frac{\alpha}{q_H}. \tag{17}$$

As a result, we infer that we have a good chance of extracting a witness, given a persuasive prover:

$$\frac{\alpha^2}{4q_H} - \frac{1}{q}. \tag{18}$$

If it is necessary for the person who is proving a statement to possess certain confidential knowledge, then the person who is verifying the statement will not be able to prove the statement to anyone else unless they also possess the confidential information. The assertion that the prover possesses such information must be included in the statement that is being proven, but the knowledge itself cannot be included in the assertion, nor can it be transmitted with it. If this were not the case, the statement could not be proven using the zero-knowledge proof method since it would present the verifier with more information about the statement by the time the protocol was completed. A proof of knowledge is considered to be in the particular situation of zero knowledge when the assertion consists of nothing more than the fact that the prover is in possession of the confidential information.

As proved, concerning zero knowledge, even a node that does not hold a piece of valid information will rewind to a previous state and choose a new random permutation employed in the new execution. This is because zero knowledge prevents a node from storing any information at all. It is possible to demonstrate that the protocol with random nodes does not have an expected execution time of a different order of magnitude. Still, this protocol does not comprehend the distinction. Therefore, the protocol possesses the quality of not revealing any information.

5. Conclusions

This work proposed an innovative ZNP system [12] to ensure the methods and innovation produced in music education processes, using blockchain [9, 42] technology and smart contracts [14, 44]. The motivation for the development of this protocol is that, in the "conventional" authentication protocols [39, 45], at the end of their execution, the member who verifies the identity of his peer has messages and secrets that he can use for impersonation [37, 46]. Contrary to the proposed standard, the secret used to prove a member's identity depends on a specific time, so that, at another time, it is useless. In other words, the musical educational processes and the participants may know a secret, but without revealing any information about this secret. Three different examples were used to demonstrate

the capability of the template as both a computing and a cryptographic model, capable of responding to the suggested implementation and ensuring the authentication processes of blockchain technology.

Even though it may be possible to achieve a level of protection in musical educational processes that are practically acceptable, it is evident that a significant amount of research work is still required because the requirements are high and are continually increasing [47]. The sheer number of potential solutions and the associated expenses illustrate how challenging it is to ensure the safety of a comparable system in a safe setting. It is acceptable to conclude that, to secure it, specialized methods of issuing identities to the blockchain nodes, scattering the nodes, instant data copying, and an access mechanism that gives high possibilities of maintaining security and privacy are required [48].

A main future extension is a study of how the proposed methodology is improved when additional information is added to it, both from the network and from the music education content. This would apply to both of these sources of data. In addition to this, we intend to study the impact that the amount of the data has on the algorithm's scalability and evaluate how well our method performs in additional private databases. It would be interesting also to investigate new ways of encoding the available information with cryptotensioners to integrate this further information into the technique that has been proposed on a methodological level.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] H. N. Chua, J. S. Teh, and A. Herbland, "Identifying the effect of data breach publicity on information security awareness using hierarchical regression," *IEEE Access*, vol. 9, pp. 121759–121770, 2021.

[2] A. Bates and W. U. Hassan, "Can data provenance put an end to the data breach?" *IEEE Security & Privacy*, vol. 17, no. 4, pp. 88–93, 2019.

[3] N. Li, "Combination of blockchain and AI for music intellectual property protection," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–8, Article ID 4482217, 2022.

[4] A. S. Al-Ahmad and H. Kahtan, "Cloud computing review: features and issues," in *Proceedings of the 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, pp. 1–5, Singpore, Juy 2018.

[5] F. Wang, H. Wang, and L. Xue, "Research on data security in big data cloud computing environment," in *Proceedings of the 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1446–1450, China, Octomber 2021.

[6] M. Belotti, N. Bozic, G. Pujolle, and S. Secci, "A vademecum on blockchain technologies: when, which, and how," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3796–3838, 2019.

[7] R. Amelin, V. Arkhipov, S. Channov, M. Dobrobaba, and V. Naumov, "Prospects of blockchain-based information systems for the protection of intellectual property," *Communications in Computer and Information Science*, vol. 1038, pp. 327–337, 2019.

[8] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, "Blockchain-enabled smart contracts: architecture, applications, and future trends," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 11, pp. 2266–2277, 2019.

[9] C. G. Akcora, M. Kantarcioglu, and Y. R. Gel, "Blockchain data analytics," in *Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM)*, p. 6, Coimbatore, Chennai, November 2018.

[10] M. R. Ogiela and M. Oczko, "Comparison of selected homomorphic encryption techniques," in *Proceedings of the 2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, pp. 1110–1114, Zhengzhou, February 2018.

[11] R. Sendhil and A. Amuthan, "A descriptive study on homomorphic encryption schemes for enhancing security in fog computing," in *Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC)*, pp. 738–743, September 2020.

[12] X. Yang and W. Li, "A zero-knowledge-proof-based digital identity management scheme in blockchain," *Computers & Security*, vol. 99, Article ID 102050, 2020.

[13] S. Sahai, N. Singh, and P. Dayama, "Enabling privacy and traceability in supply chains using blockchain and zero knowledge proofs," in *Proceedings of the 2020 IEEE International Conference on Blockchain (Blockchain)*, pp. 134–143, Augest 2020.

[14] Y. Hu, M. Liyanage, A. Mansoor, K. Thilakarathna, G. Jourjon, and A. Seneviratne, "Blockchain-based Smart Contracts-Applications and Challenges," 2018, https://arxiv.org/abs/1810.04699.

[15] L. Cao and Z. Wan, "Anonymous scheme for blockchain atomic swap based on zero-knowledge proof," in *Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 371–374, June 2020.

[16] G. Hua, L. Zhu, J. Wu, C. Shen, L. Zhou, and Q. Lin, "Blockchain-based federated learning for intelligent control in heavy haul railway," *IEEE Access*, vol. 8, pp. 176830–176839, 2020.

[17] L. S. Sankar, M. Sindhu, and M. Sethumadhavan, "Survey of consensus protocols on blockchain applications," in *Proceedings of the 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pp. 1–5, Coimbatore, India, January 2017.

[18] E. Buchman, *Tendermint: Byzantine Fault Tolerance in the Age of Blockchains*, 109 pages, The University of Guelph, Guelph, Ontario, Canada, 2020.

[19] K. Lei, Q. Zhang, L. Xu, and Z. Qi, "Reputation-based byzantine fault-tolerance for consortium blockchain," in *Proceedings of the 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 604–611, Singapore, December 2018.

[20] G. Ra, T. Kim, and I. Lee, "VAIM: verifiable anonymous identity management for human-centric security and privacy in the internet of things," *IEEE Access*, vol. 9, pp. 75945–75960, 2021.

[21] K. Gao, F. Han, P. Dong, N. Xiong, and R. Du, "Connected vehicle as a mobile sensor for real time queue length at

signalized intersections," *Sensors*, vol. 19, no. 9, Article ID 2059, 2019.

[22] X. Wang, Q. Li, N. Xiong, and Y. Pan, "Ant colony optimization-based location-aware routing for wireless sensor networks," in *Wireless Algorithms, Systems, and Applications*pp. 109–120, Berlin, Heidelberg, 2008.

[23] L. Avigad and O. Goldreich, "Testing graph blow-up," in *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, D. Zuckerman and O. Goldreich, Eds., Springer, Berlin, Heidelberg, pp. 156–172, 2011.

[24] Y. Jiang, G. Tong, H. Yin, and N. Xiong, "A pedestrian detection method based on genetic algorithm for optimize XGBoost training parameters," *IEEE Access*, vol. 7, pp. 118310–118321, 2019.

[25] R. Wan, N. Xiong, and N. T. Loc, "An energy-efficient sleep scheduling mechanism with similarity measure for wireless sensor networks," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, p. 18, 2018.

[26] P. Mateus, F. Moura, and J. Rasga, "Transferring proofs of zero-knowledge systems with quantum correlations," in *Proceedings of the 2007 First International Conference on Quantum, Nano, and Micro Technologies (ICQNM'07)*, p. 9p. 9, January 2007.

[27] A. Broadbent, Z. Ji, F. Song, and J. Watrous, "Zero-knowledge proof systems for QMA," in *Proceedings of the 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 31–40, NY, Heidelberg, October 2016.

[28] *EUROCRYPT'92 Advances in Cryptology-*, 2020, https://www.bookdepository.com/Advances-Cryptology-EUROCRYPT-92-Rainer-Rueppel/9783540564133.

[29] M. Bellare and O. Goldreich, "On probabilistic versus deterministic provers in the definition of proofs of knowledge," in *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, D. Zuckerman and O. Goldreich, Eds., Springer, Berlin, Heidelberg, pp. 114–123, 2011.

[30] L. E. B. Salasar, J. G. Leite, and F. Louzada, "Likelihood-based inference for population size in a capture-recapture experiment with varying probabilities from occasion to occasion," *Brazilian Journal of Probability and Statistics*, vol. 30, no. 1, pp. 47–69, 2016.

[31] Z. Wan, Y. Zhou, and K. Ren, "Zk-AuthFeed: protecting data feed to smart contracts with authenticated zero knowledge proof," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. –1, p. 1, 2022.

[32] H. Ryu, D. Kang, and D. Won, "On a partially verifiable multiparty multi-argument zero-knowledge proof," in *Proceedings of the 2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, pp. 1–8, Heidelberg, China, January 2021.

[33] Y.-H. Chen, C.-Q. Ye, and P. Zhang, "Efficient group signature scheme based on RSA cryptosystem," in *Proceedings of the 2006 International Conference on Computing & Informatics*, pp. 1–3, NY, USA, June 2006.

[34] Y. Lu, S. Wu, Z. Fang, N. Xiong, S. Yoon, and D. S. Park, "Exploring finger vein based personal authentication for secure IoT," *Future Generation Computer Systems*, vol. 77, pp. 149–160, 2017.

[35] O. Baudron, P.-A. Fouque, D. Pointcheval, J. Stern, and G. Poupard, "Practical multi-candidate election system," in *Proceedings of the twentieth annual ACM symposium on Principles of distributed computing-PODC'01*, pp. 274–283, New York, NY, USA, December 2001.

[36] Z. Fei, K. Liu, B. Huang, Y. Zheng, and X. Xiang, "Dirichlet process mixture model based nonparametric Bayesian modeling and variational inference," in *Proceedings of the 2019 Chinese Automation Congress (CAC)*, pp. 3048–3051, Henan, China, Augest 2019.

[37] T. Shen, F. Wang, K. Chen, K. Wang, and B. Li, "Efficient leveled (multi) identity-based fully homomorphic encryption schemes," *IEEE Access*, vol. 7, pp. 79299–79310, 2019.

[38] T. Miyamae, F. Kozakura, M. Nakamura et al., "ZGridBC: zero-knowledge proof based scalable and private blockchain platform for smart grid," in *Proceedings of the 2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 1–3, China, February 2021.

[39] A. Pathak, T. Patil, S. Pawar, P. Raut, and S. Khairnar, "Secure authentication using zero knowledge proof," in *Proceedings of the 2021 Asian Conference on Innovation in Technology (ASIANCON)*, pp. 1–8, Heidelberg, China, December 2021.

[40] O. Goldreich, M. Sudan, and L. Trevisan, "From logarithmic advice to single-bit advice," in *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, L. Avigad, M. Bellare, Z. Brakerski et al., Eds., Springer, Berlin, Heidelberg, pp. 109–113, 2011.

[41] A. Broadbent and A. B. Grilo, "QMA-hardness of consistency of local density matrices with applications to quantum zero-knowledge," in *Proceedings of the 2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 196–205, Berlin, Heidelberg, Augest 2020.

[42] W. Lin, X. Zhang, Q. Cui, and Z. Zhang, "Blockchain based unified authentication with zero-knowledge proof in heterogeneous MEC," in *Proceedings of the 2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, Heidelberg, June 2021.

[43] O. Goldreich and D. Zuckerman, "Another Proof that $\mathcal{BPP}\subseteq \mathcal{PH}$ (and More)," in *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, L. Avigad, M. Bellare, Z. Brakerski et al., Eds., Springer, Berlin, Heidelberg, pp. 40–53, 2011.

[44] V. Aleksieva, H. Valchanov, and A. Huliyan, "Implementation of smart-contract, based on hyperledger fabric blockchain," in *Proceedings of the 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA)*, pp. 1–4, Berlin, Heidelberg, June 2020.

[45] J. Kim and A. Yun, "Secure fully homomorphic authenticated encryption," *IEEE Access*, vol. 9, pp. 107279–107297, 2021.

[46] M. Mohan, M. K. K. Devi, and V. J. Prakash, "Homomorphic encryption-state of the art," in *Proceedings of the 2017 International Conference on Intelligent Computing and Control (I2C2)*, pp. 1–6, NY, USA, June 2017.

[47] C. Jiang and C. Ru, "Application of blockchain technology in supply chain finance," in *Proceedings of the 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*, pp. 1342–1345, Zhengzhou, China, September 2020.

[48] J. Bringer, H. Chabanne, and A. Patey, "Privacy-preserving biometric identification using secure multiparty computation: an overview and recent trends," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 42–52, 2013.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 2266171, 9 pages
https://doi.org/10.1155/2022/2266171

Research Article

Applications of Game Theory and Advanced Machine Learning Methods for Adaptive Cyberdefense Strategies in the Digital Music Industry

Jing Jing

Music Teaching Department, Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Jing Jing; jing_jing26e@126.com

Received 3 May 2022; Revised 17 May 2022; Accepted 24 May 2022; Published 17 June 2022

Academic Editor: Konstantinos Demertzis

As the likelihood and impact of cyber-attacks continue to grow, organizations realize the need to invest in specialized methods to protect their digital data and the information they circulate or manage. Due to its broad use, game theory has evolved into a concept that can be applied practically while analyzing and modifying existing cyber protection methods to arrive at the best possible conclusions. This study presents an innovative hybrid model that combines game theory and advanced machine learning methods for adaptive cyber defense strategies. Specifically, a repetitive game methodology is implemented to analyze cyber-attacks and model behaviors and study how defenders and attackers make decisions in a competing field. Based on Bayesian inference, the proposed method can predict the next steps in the game to produce the appropriate countermeasures and implement the best cyber defense strategies that govern an organization. The suggested system introduced to the academic literature for the first time was successfully tested in a particular application scenario involving the digital music industry and coping with impending cyber-attacks.

1. Introduction

As the cyberspace landscape evolves rapidly, a dynamic framework emerges in which very delicate balances are observed to make optimal decisions. The asset, information, and data values that modern organizations must manage constantly increase. New cyber-attack techniques are continually developing while existing technological defense systems age, making active defense strategies resilient. Furthermore, economic changes, institutional reorganizations, consumer trends, and legal and regulatory compliance requirements all impact decision-making and the overall development of sound defense strategies for an organization or company. To maintain its competitive advantage, the organization in question will need to continuously improve its defense strategies, which will be based on ongoing knowledge and utilization of the cyber threat landscape [1]. An accurate inventory of all assets classified by their value to the organization is critical in assessing the severity of the risks they face and, as a result, the decisions that must be made about them.

Game theory [2] can be used in cybersecurity to create tangible solutions that will allow the maximum utilization of existing strategies and optimize them to create a robust and long-term security environment at the organization level [3–5]. Using game theory principles, cyber security professionals can implement a network of controls that specialize and, as a result, reduce the risk to their valuable assets. They can also apply areas with a low level of risk, maximizing their return on investment. As a result, using specialized scenarios based on game theory, it is possible to predict the attackers' strategy at each stage of the attack cycle, assisting in developing intelligent models to improve cyber security and the development of new intelligent systems to deter attackers.

This study presents applications of game theory and advanced machine learning methods for adaptive cyber defense strategies in the digital music industry. An

innovative application of a repetitive game methodology for cyber-attack analysis is proposed; using Bayesian inference [6], the next steps in the game can be predicted so that the best cyber defense strategies can be implemented to shield an organization from cyber-attacks.

2. Literature Review

The literature on game theory for adaptive cyber defense approaches is extensive, whether at the network level, where we must deal with massive amounts of raw data or at the strategy level.

In 2015, Laszka et al. [7] developed a strategy for reducing spear-phishing assaults by targeted per-user filtering criteria. They framed the task of screening harmful e-mails, both targeted and untargeted, as a security game. These defined optimum filtering techniques and demonstrated how they might be computed in practice. They put their theoretical hypotheses to the test by comparing them to two datasets taken from the actual world. Using two different sets of real data, they demonstrated that the recommended baselines result in less harm than the nonstrategic restrictions. In addition, they found that the improvement over nonstrategic criteria was more substantial for the comprehensive information. This improvement was unaffected by an increase in the number of targeted customers. This indicated that their technique scaled effectively, both analytically and in terms of how well it performed.

Schlenker et al. [8] explored the critical issue of allocating cyber alarms to a restricted number of professionals in cyber security activities. They proposed the cyber-alert allocation game to investigate this issue and demonstrated how to compute the defender's best options. They offered a unique technique for addressing implement ability concerns in determining the defender's best marginal tactic to resolve this game. Finally, they presented heuristics for solving big games like the one described and an objective assessment of the suggested framework and solution methods.

Nguyen et al. [9] conducted a study of deep reinforcement learning (DRL) techniques used in cyber protection. They discussed various critical topics, such as DRL-based security approaches for cyber-physical infrastructure, independent intrusion detection approaches, and multiagent DRL-based game theory simulations for cyber-attack defensive tactics. Additionally, extensive debates and potential study paths on cyber security focusing on DRLs are provided. They hoped that this exhaustive assessment would provide the groundwork for and support future research into the ability to develop DRL to address more sophisticated digital privacy concerns.

Alpcan and Basar [10], in 2010, in their book, about network security and game theoretic approaches, aimed to present a theoretical foundation for making resource allocation decisions that balance available capabilities and perceived security risks in a principled manner. They focused on analytical models based on game, information, communication, optimization, decision, and control theories applied to diverse security topics. At the same time, connections between theoretical models and real-world security problems are highlighted to establish the critical feedback loop between theory and practice.

Hemberg et al. [11] presented an architecture for adversarial AI called RIVALS that abstractly replicated the hostile, competing for a coevolutionary mechanism in security contexts. The purpose was to develop a system capable of pro-active cyber security against dynamic automated attackers. They reviewed its present uses and how it is used to develop defensive measures. Further work will involve expanding it to enable more cyber defense purposes, creating more effective or genuine reality methods, and applying other Nash equilibrium-finding algorithms to other cyber security challenges with established Nash equilibria [12] and analyzing efficiency.

3. Proposed Game Strategy

Many cyber-attacks follow a pattern built on repeating tactics or procedures over time. Infrastructure vulnerability control strategies, for example, can compete with current defensive systems and change their applications over time. In this sense, attackers and defenders engage regularly, and these interactions may be depicted using repetitive games, which are a type of dynamic game [13]. The concept described here is using a repeated game to evaluate cyber-attacks and simulating some cooperative behaviors without a clear endpoint. A repetitive game begins with a static game repeated infinitely or intermittently many times. A reward is given to each player who completes a specific action in this strategic stage game. The sum of each player's gains throughout the game constitutes their final reward. In addition, Bayesian inference [14] can predict the next steps in the game to produce the appropriate countermeasures and implement the best cyber defense strategies that govern a complex system with high uncertainty [15].

The starting point for modeling the proposed system is a static game of the following format:

$$\Gamma = \{N, (X_i, u_i)_{i \in N}\}, \tag{1}$$

where N = {1, 2,. .., n} is the set of players, X_i is the set of the player's í pure strategies, and u is its performance function. Assume that this stage game is repeated T times, where T is finite or infinite. Each repetition takes place over a period. A typical time (or stage) is denoted by í, where í = *1*, 2, ..., T. The interaction evolves as follows [16, 17]:

(1) In period 1, players simultaneously select actions, which we symbolize as $x^1 = (x_1^1, x_2^1, \ldots, x_n^1)$, where the pointer symbolizes the player and the exponent symbolizes the stage (time period). The action x_1^1 belongs to the set $X_i, i \in N$. Each player has been informed afterward of the choices of the other players. The performance of player i in this period is

$$u_i\left(x^1\right), \quad i \in N. \tag{2}$$

(2) In period 2, players choose at the same time actions $x^2 = (x_1^2, x_2^2, \ldots, x_n^2)$. Each player has been informed

afterward of the choices of the other players. Player's i odds are

$$u_i\left(x^2\right), \quad i \in N. \tag{3}$$

(3) In period T, players choose actions $x^T = (x_1^T, x_2^T, \ldots, x_n^T)$. The player's odds are

$$u_i\left(x^T\right), \quad i \in N. \tag{4}$$

(4) If T is finite, the interaction is complete (at the end of period T). Otherwise, the game continues in perpetuity

In the repetitive game, we suggest each player earns a sequence of payoffs (one payoff for each period). This yield sequence is assumed to be valued using the discounted sum of the sequence terms. The term discount expresses the assumption that a person does not value current and future returns equally. Parameter δ is the discount rate of an individual [18]. The closer δ is to 0, the less the individual values a future versus a present performance. In other words, the smaller the δ is, the less the person is interested in the future or the more impatient he is. Conversely, the higher the δ (the closer it is to 1) is, the more a person values a future performance or is more patient [19–21].

Let a finite terminal story $h^T = (x^1, x^2, \ldots, x^T)$. If the discount rate of player i is given by the parameter $\delta_i \in (0, 1)$, then the discounted sum of the payoffs of player i is

$$\begin{aligned} V_i &\equiv u_i\left(x^1\right) + \delta_i u_i\left(x^2\right) + \delta_i^2 u_i\left(x^3\right) + \cdots + \delta_i^{T-1} u_i\left(x^T\right) \\ &= \sum_{t=1}^{T} \delta_i^{t-1} u_i\left(x^t\right). \end{aligned} \tag{5}$$

Let now an infinite terminal story $h^\infty = (x^1, x^2, \ldots)$. In this case, the discounted sum of player's payoffs is

$$V_i \equiv u_i\left(x^1\right) + \delta_i u_i\left(x^2\right) + \delta_i^2 u_i\left(x^3\right) + \cdots = \sum_{t=1}^{\infty} \delta_i^{t-1} u_i\left(x^t\right). \tag{6}$$

In the case of infinity T, instead of the discounted sum of the odds, we use the discounted average of the player's odds as a valuation function:

$$u_i^t = u_i\left(x^t\right), \quad t = 1, 2, \ldots. \tag{7}$$

Based on the relation, player i obtains the sequence of odds $(u_i^1, u_i^2, \ldots)$. We define a number c such that i is indifferent between the series of yields $(u_i^1, u_i^2, \ldots)$ and the sequence $(c, c_1, ..., c_n)$. So, we take the following relation [22]:

$$\begin{aligned} V_i &= \sum_{t=1}^{\infty} \delta_i^{t-1} u_i^t \\ &= \sum_{t=1}^{\infty} \delta_i^{t-1} c. \end{aligned} \tag{8}$$

However, it is true that

$$\sum_{t=1}^{\infty} \delta_i^{t-1} c = \frac{c}{1 - \delta_i}. \tag{9}$$

Consequently, we have

$$\begin{aligned} V_i &= \sum_{t=1}^{\infty} \delta_i^{t-1} u_i^t \\ &= \frac{c}{1 - \delta_i}, \end{aligned} \tag{10}$$

so that

$$(1 - \delta_i) V_i = (1 - \delta_i) \sum_{t=1}^{\infty} \delta_i^{t-1} u_i^t = c. \tag{11}$$

The term $(1 - \delta_i)V_i$ is the discounted average of the yield sequence $(u_i^1, u_i^2, \ldots)$. The functions V_i and $(1 - \delta_i)V_i$ express the same preferences since one is a positive monotonic transformation. Thus, the set of all terminal stories is the set of all sequences [3, 23]:

$$h^\infty = \left(x^1, x^2, \ldots\right). \tag{12}$$

So, player i evaluates the terminal history h^∞ based on the function:

$$V_i = \sum_{t=1}^{\infty} \delta^{t-1} u_i\left(x^t\right), \tag{13}$$

or it is equivalent:

$$(1 - \delta) V_i = (1 - \delta) \sum_{t=1}^{\infty} \delta^{t-1} u_i\left(x^t\right). \tag{14}$$

So, we have the repetitive game $G^\infty(\delta)$.

A rule that dictates the player's behavior in response to any and all scenarios is an example of a strategy. Consequently, a single-player strategy is a guideline for picking actions at each stage (repetition) of the stage game as a function of previous decisions when playing a game that features observable actions and is repetitive. To be more exact, the energy that the player will pick at each narrative stage is determined by the strategy used in a game with just one player and repeated steps. The action must be feasible; that is, it must belong to all the available options of the gaming stage [24, 25].

Odds pairs corresponding to the four pairs of clear strategies are feasible (in the sense that there are strategies that generate specific returns). It should also be noted that all convex combinations of two or more yield pairs of pure methods constitute achievable yields (using mixed strategies) [5, 26].

In general, the total possible odds of the stage game are given by the quadrilateral in Figure 1, which has as vertices the four pairs of payoffs corresponding to the clear strategies described by the set {(5, 5), (0, 6), (6, 0), (1,1)}.

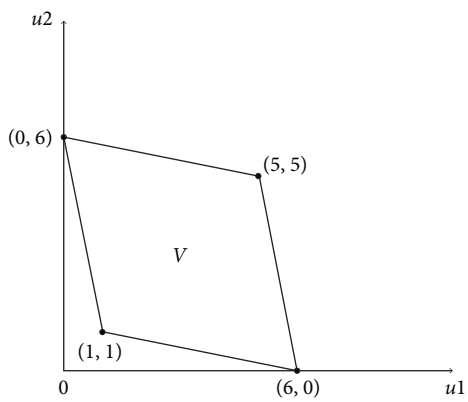


Figure 1: Possible payoffs in the form of a diagram.

4. Application Testing

In order to model the proposed system, a specialized threat scenario was implemented with an application study in the music industry. This was done because the large number of visitors combined with the enormous amounts of music content spent on a daily basis creates a new landscape of threats, in which the clear strategies for cybersecurity need to be rearranged on an ongoing basis. According to this logic, they frequently employ advanced techniques, including zero-day attacks, to launch attacks on music streaming platforms that modern cybercriminals have targeted [24, 27, 28].

For the implementation of the proposed system, we consider a static game $\Gamma = \{N, (X_i, u_i)_{i\in N}\}$, such that

(1) $(u_1^*, u_2^*, \ldots, u_n^*)$ is the payoff vector of a Nash equilibrium of Γ (the Nash equilibrium is a solution technique for non-cooperative games with two or more participants in game theory. We assume that each player is aware of the equilibrium strategies that the other players would employ and that no player stands to gain by simply changing his strategy [2] if each participant has decided on a strategy and no one has the option of modifying their strategy. On the contrary, the other players maintain their strategy constant; their set of plans and results are the Nash equilibrium.

(2) $(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_n)$ is a vector of possible returns of Γ such that $\hat{v}_i > u_i^*$ for each i.

(3) The discount rate δ is quite large.

Then, there is a perfect subgame balance for the game $G^{\infty}(\delta)$ with average odds $(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_n)$.

To prove the above scenario, we will assume that the vector $(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_n)$ can be achieved through a vector of pure strategies $\hat{x} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)$. That is, $u_i(\hat{x}) = \hat{u}_i, i \in N$. It should be noted that it is not necessary to assume clean strategies. If this is not the case, vector V is achieved through mixed strategies [5, 10].

Suppose that the equilibrium returns $(u_1^*, u_2^*, \ldots, u_n^*)$ of Γ are derived from the strategy vector $(x_1^*, x_2^*, \ldots, x_n^*)$.

We consider the following firing strategy:

(1) Stage $i = 1$:

(a) player i selects an action $\hat{x}_i$

(2) Stage $i > 1$:

(a) If the result in the previous stages 1, 2, ..., t-1 is $\hat{x}$, player i selects an action $\hat{x}_i$

(b) In all other cases, player i selects an action x_i^*

We will examine the players' motivations for adopting or not the above strategy.

Let us look at player i if all other players adopt the firing strategy. We must determine the optimal response of i in each period where it has observed a result $\hat{x}$ and its optimal response in each period where it has observed a result different from $\hat{x}$.

It should be recorded the actions that i will take if it observes in a period t that in the previous period t-1 a result other than $\hat{x}$ was outcome. Player i knows that, in period t, the other players will choose the actions x_{-i}^* and that this will be done in perpetuity (because they follow the firing strategy). Therefore, the optimal reaction of 1 is to choose x_i^* in perpetuity.

Let us now see what player i will do in those periods for which history contains only the result $\hat{x}$. Suppose that, in some of them, i chooses to deviate from the energy $\hat{x}_i$. The optimal deviation, which we denote by x_i^d d, is what solves the problem.

In particular,

$$\max_{x_i} u_i(x_i, \hat{x}_{-i}). \tag{15}$$

The yield of i during the deviation period is

$$u_i^d \equiv u_i\left(x_i^d, \hat{x}_{-i}\right), \tag{16}$$

where it is true that

$$u_i^d \geq \hat{u}_i. \tag{17}$$

From the next period, the other players will start choosing x_{-i}^*, after noticing the choice x_i^d of i. The punishment period of i will therefore begin. Given this, the optimal response of i is to select the energy x_i^*. Thus, from the period following the deviation to perpetuity, i will have a payoff of u_i^* per period. Its average discounted return from the breach is, therefore,

$$(1-\delta)\left(u_i^d + \delta u_i^* + \delta^2 u_i^* + \ldots\right) = (1-\delta)u_i^d + \delta u_i^*. \tag{18}$$

If, on the contrary, i, after each story that contains only $\hat{x}$, chooses the action $\hat{x}_i$; its return will be $\hat{u}_i$ because after such a story, the other players choose $\hat{x}_{-i}$. For each of the following periods, the same result will be repeated, and i will continuously gain $\hat{u}_i$ per period. The average payoff of i in this case is

$$(1-\delta)\left(\hat{u}_i + \delta\hat{u}_i + \delta^2\hat{u}_i + \ldots\right) = (1-\delta)\frac{1}{1-\delta}\hat{u}_i = \hat{u}_i. \tag{19}$$

Therefore, player i will select $\hat{x}_i$ after each story that contains only $\hat{x}$ if

$$\widehat{u}_i \geq (1-\delta)u_i^d + \delta u_i^* \Longleftrightarrow \delta\left(u_i^d - u_i^*\right) \geq u_i^d - \widehat{u}_i. \tag{20}$$

If the following inequality relations are valid,

$$u_i^d \geq \widehat{u}_i > u_i^*, \tag{21}$$

so it applies

$$\delta \geq \frac{u_i^d - \widehat{u}_i}{u_i^d - u_i^*}. \tag{22}$$

And so

$$\frac{u_i^d - \widehat{u}_i}{u_i^d - u_i^*} \leq 1. \tag{23}$$

The possible payoffs that exceed the Nash payoffs are shown in Figure 2.

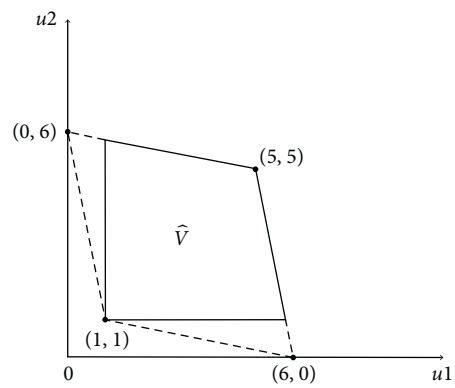


Figure 2: Possible payoffs that exceed Nash payoffs.

Next, to determine which value can best estimate the random variable, we apply Bayesian inference to create a loss function, which will measure how wrong the estimate is, that is, how different the estimation of the parameter is from its actual price. This means that the goal is for the value of the estimate to minimize the loss function. For example, in the absolute error loss function, underestimation and overestimation of the parameter are punished in the same way, depending on the deviation of the estimate. In contrast to the preceding statement, the linear loss function penalizes underestimation with a different weight than overestimation.

The evaluation of the decision rules is done through the risk function, which is defined as the average value of the loss function. If $L(\delta, \theta)$ is the loss function and $\delta = \delta(x)$ is the decision rule, the hazard function of the decision rule δ is mathematically expressed as

$$\begin{aligned} R(\delta, \theta) &= E_\theta L(\delta, \theta) \\ &= \int L(\delta(x), \theta) f(x|\theta) \mathrm{d}x. \end{aligned} \tag{24}$$

In the Bayesian approach, it is possible to balance the risk function of each decision rule, based on ex-ante personal opinion, through $u(\theta)$. So, the Bayesian risk is defined as [29–31]

$$\begin{aligned} BR(\delta) &= \int_\Theta R(\delta, \theta) u(\theta) \mathrm{d}\theta \\ &= \int_\Theta \int_X L(\delta(x), \theta) f(x|\theta) u(\theta) \mathrm{d}x\, \mathrm{d}\theta. \end{aligned} \tag{25}$$

The decision rule that minimizes Bayesian risk is the Bayesian rule. In particular, the Bayesian risk of decision rule δ, as a function of the loss of a square error, is equal to

$$\begin{aligned} E(\delta - \theta)^2 &= \int_\Theta \int_X (\delta(x) - \theta)^2 f(x|\theta) \mathrm{d}x u(\theta) \mathrm{d}\theta \\ &= \int_\Theta \int_X (\delta(x) - \theta)^2 f(x|\theta) u(\theta) \mathrm{d}x \mathrm{d}\theta \\ &= \int_\Theta \int_X (\delta(x) - \theta)^2 f(x, \theta) \mathrm{d}x \mathrm{d}\theta \\ &= \int_X \int_\Theta (\delta(x) - \theta)^2 p(\theta|x) g(x) \mathrm{d}\theta\, \mathrm{d}x \\ &= \int_X \int_\Theta (\delta(x) - \theta)^2 p(\theta|x) \mathrm{d}\theta g(x) \mathrm{d}x. \end{aligned} \tag{26}$$

Since the margin function g(x) is a nonnegative function, we are interested in minimizing the

$$I(x) = \int_\Theta (\delta(x) - \theta)^2 p(\theta|x) \mathrm{d}\theta, \tag{27}$$

which becomes

$$\begin{aligned} I(x) &= \int_\Theta \left(\delta(x)^2 - 2\theta\delta(x) + \theta^2\right) p(\theta|x) \mathrm{d}\theta \\ &= \delta(x)^2 \int_\Theta p(\theta|x) \mathrm{d}\theta - 2\delta(x) \int_\Theta \theta p(\theta|x) \mathrm{d}\theta + \int_\Theta \theta^2 p(\theta|x) \mathrm{d}\theta \\ &= \int_\Theta p(\theta|x) \mathrm{d}\theta \left(\delta(x)^2 - 2\delta(x) \frac{\int_\Theta \theta p(\theta|x) \mathrm{d}\theta}{\int_\Theta p(\theta|x) \mathrm{d}\theta} \right) + \int_\Theta \theta^2 p(\theta|x) \mathrm{d}\theta \\ &= \int_\Theta p(\theta|x) \mathrm{d}\theta \left(\delta(x) - \frac{\int_\Theta \theta p(\theta|x) \mathrm{d}\theta}{\int_\Theta p(\theta|x) \mathrm{d}\theta} \right)^2 + \left[\int_\Theta \theta^2 p(\theta|x) \mathrm{d}\theta - \frac{\left(\int_\Theta \theta p(\theta|x) \mathrm{d}\theta\right)^2}{\int_\Theta p(\theta|x) \mathrm{d}\theta} \right]. \end{aligned} \tag{28}$$

The above function is minimized for $\delta(x)$ when

$$\delta(x) = \frac{\int_{\Theta} \theta p(\theta|x) \mathrm{d}\theta}{\int_{\Theta} p(\theta|x) \mathrm{d}\theta} = E(\theta|x). \tag{29}$$

So, through a hypothesis check, we can determine the correctness of a hypothesis that concerns an action without dealing with estimating the possible values of the strategy. Based on this view, it is possible to predict the future prices of the process as, through the ex-post distribution, the forecasting process is immediate and accurate.

Assuming that, for a random sample of observations $x = x_1, x_2, \ldots, x_n$, it is desirable to predict the future observation y, and it is necessary to calculate the a posteriori distribution of prediction $\pi(y \mid x)$. For the mathematical calculation of the ex-post distribution, the mathematical analysis of the ex-prediction distribution $g(x)$ is required:

$$g(x) = \int_{\theta} f(x, \theta) \mathrm{d}\theta = \int_{\theta} f(x|\theta) u(\theta) \mathrm{d}\theta. \tag{30}$$

Observing the equation, it is easy to see that the integral contains the product of the probability function $f(x|\theta)$, with the ex-ante distribution $u(\theta)$. Thus, based on the above procedure, the mathematical definition of the ex-post prediction distribution $\pi(y|x)$ will be made:

$$\begin{aligned} \pi(y|x) &= \int_{\theta} f(y, \theta|x) \mathrm{d}\theta \\ &= \int_{\theta} f(y|\theta, x) p(\theta|x) \mathrm{d}\theta \\ &= \int_{\theta} f(y|\theta) p(\theta|x) \mathrm{d}\theta. \end{aligned} \tag{31}$$

The probability function of the following observations is equal to the integral of the joint probability function $f(y, \theta \mid x)$. It should be noted that this time the shared distribution function is bound to the observed data. Reapplying the Bayesian theorem, we conclude that the ex-post distribution of prediction is again equal to the product of probability $f(y|\theta)$, but this time, with the ex-post distribution $p(\theta|x)$, this is completed for the variable θ.

The effect of the ex-post information is therefore apparent. It should be noted that, from the probability function $f(y, \theta|x)$, we came to the probability function if all a posteriori information originates from the parameter θ.

To prove the validity of the methodology described above, we will make the assumption that we will count the number of privilege escalation attacks made against a randomly chosen account on the music streaming platform that is the focus of the current example.

Suppose that the number of specific attacks over a period follows the "Poisson" distribution $(\theta), \theta \in \Theta = (0, \infty)$, i.e.,

$$f(x_i|\theta) = e^{-\theta} \frac{\theta^{x_i}}{x_i!}, \quad i = 1, 2, \ldots, n. \tag{32}$$

For n observation time, the probability of the sample is

$$\begin{aligned} L(\theta|x) &= \prod_{i=1}^{n} f(x_1|\theta) \\ &= \prod_{i=1}^{n} e^{-\theta} \frac{\theta^{x_i}}{x_i!} \\ &= e^{-n\theta} \frac{\theta^{\sum x_i}}{\prod x_i!}. \end{aligned} \tag{33}$$

Suppose that the ex-ante distribution of θ follows the *Gamma* distribution $(a\ \beta)$, where $a, \beta > 0$; quantities are known as a function of probability density:

$$u(\theta) = \frac{\theta^{\alpha-1} e^{-(\theta/\beta)}}{\beta^{\alpha} \Gamma(\alpha)}. \tag{34}$$

Therefore, the distribution margin is

$$\begin{aligned} g(x) &= \int_0^{\infty} L(\theta|x) u(\theta) \mathrm{d}\theta \\ &= \int_0^{\infty} \left(e^{-n\theta} \frac{\theta^{\sum x_i}}{\prod x_i!} \right) \left(\frac{\theta^{\alpha-1} e^{-(\theta/\beta)}}{\beta^{\alpha} \Gamma(\alpha)} \right) \mathrm{d}\theta \\ &= \frac{1}{\beta^{\alpha} \Gamma(\alpha) \prod x_i!} \int_0^{\infty} e^{-n\theta - (\theta/\beta)} \theta^{\sum x_i + \alpha - 1} \mathrm{d}\theta \\ &= \frac{(\beta/n\beta + 1)^{\sum x_i + \alpha} \Gamma(\sum x_i + \alpha)}{\beta^{\alpha} \Gamma(\alpha) \prod x_i!} \\ &\quad \cdot \int_0^{\infty} \frac{e^{-\theta(n\beta + 1/\beta)} \theta^{(\sum x_i + \alpha) - 1}}{(\beta/n\beta + 1)^{\sum x_i + \alpha} \Gamma(\sum x_i + \alpha)} \mathrm{d}\theta \\ &= \frac{(\beta/n\beta + 1)^{\sum x_i + \alpha} \Gamma(\sum x_i + \alpha)}{\beta^{\alpha} \Gamma(\alpha) \prod x_i!}. \end{aligned} \tag{35}$$

Therefore, the a posteriori distribution of θ is

$$
\begin{aligned}
p(\theta|x) &= \frac{L(\theta|x)u(\theta)}{g(\mathbf{x})} \\
&= \frac{\left(e^{-n\theta}\left(\theta^{\sum x_i}/\prod x_i!\right)\right)\left(\theta^{\alpha-1}e^{-(\theta/\beta)}/\beta^{\alpha}\Gamma(\alpha)\right)}{g(x)} \\
&= \frac{\left\{e^{-\theta(n\beta+1/\beta)}\theta^{\sum x_i+\alpha-1}/\prod x_i!\beta^{\alpha}\Gamma(\alpha)\right\}}{\left\{(\beta/n\beta+1)^{\sum x_i+\alpha}\Gamma\left(\sum x_i+\alpha\right)/\beta^{\alpha}\Gamma(\alpha)\prod x_i!\right\}} \\
&= \frac{e(n\beta+1/\beta)\theta^{\left(\sum x_i+\alpha\right)-1}}{(\beta/n\beta+1)^{\sum x_i+\alpha}\Gamma\left(\sum x_i+\alpha\right)}.
\end{aligned}
\tag{36}
$$

From the above relation, it becomes clear that the a posteriori distribution of θ is [32]

$$
\text{Gamma}\left(\sum x_i+\alpha, \frac{\beta}{n\beta+1}\right). \tag{37}
$$

The ex-post distribution following the *Gamma* distribution was expected since the *Gamma* distribution family, where the ex-distribution belongs, is conjugated to the Poisson distribution.

The ex-post average value for the above *Gamma* distribution is [30, 32, 33]

$$
\begin{aligned}
E(\theta|x) &= \left(\sum_{i=1}^{n} x_i+\alpha\right)\left(\frac{\beta}{n\beta+1}\right) \\
&= \frac{\beta\sum x_i}{n\beta+1}+\frac{\alpha\beta}{n\beta+1} \\
&= \frac{n\beta}{n\beta+1}\overline{x}+\frac{1}{n\beta+1}\alpha\beta.
\end{aligned}
\tag{38}
$$

Let $r \sim \text{Poisson}(\theta)$ be a new independent observation. The equation gives the forecast distribution:

$$
\pi(y|x) = \int_{\Theta} f(y|\theta)p(\theta|x)\mathrm{d}\theta, \tag{39}
$$

where for the specific statistical model it is [34, 35]

$$
\begin{aligned}
\pi(y|x) &= \int_{\theta}\left(e^{-\theta}\frac{\theta^{y}}{y!}\right)\left(\frac{e^{-\theta(n\beta+1/\beta)}\theta^{\left(\sum x_i+\alpha\right)-1}}{(\beta/n\beta+1)^{\sum x_i+\alpha}\Gamma\left(\sum x_i+\alpha\right)}\right)\mathrm{d}\theta \\
&= \frac{1}{y!(\beta/n\beta+1)^{\sum x_i+\alpha}\Gamma\left(\sum x_i+\alpha\right)}\int_{\theta} e^{-\theta(n\beta+\beta+1/\beta)}\theta^{\left(\sum x_i+\alpha+y\right)-1}\mathrm{d}\theta \\
&= \frac{(\beta/n\beta+\beta+1)^{\sum x_i+\alpha+y}\Gamma\left(\sum x_i+\alpha+y\right)}{(\beta/n\beta+1)^{\sum x_i+\alpha}\Gamma\left(\sum x_i+\alpha\right)\Gamma(y+t1))^{y}} \\
&= \frac{\Gamma\left(\sum x_i+\alpha+y\right)}{\Gamma\left(\sum x_i+\alpha\right)\Gamma\left(y+t1\right)\left(\frac{n\beta+1}{n\beta+\beta+1}\right)^{\sum x_i+\alpha}\left(\frac{\beta}{n\beta+\beta+1}\right)},
\end{aligned}
\tag{40}
$$

which is a negative binomial with parameters [36–38]:

$$
\left(\sum x_i+\alpha, \frac{\beta}{n\beta+\beta+1}\right). \tag{41}
$$

5. Conclusions

This research offered an original and forward-thinking application of game theory and cutting-edge machine learning methods for adaptive cyber protection measures. It suggested a methodology based on game theory, which concerns the study of elements that characterize situations of competitive interdependence with an emphasis on the decision-making process of more than one decision maker. We make the assumption that every player is informed of the equilibrium tactics that the other players will use and that no player has anything to gain by merely altering the strategy that he uses. Therefore, in order to figure out which value can provide the most accurate estimation of the random variable, we use Bayesian inference to devise a loss function. This function will measure how inaccurate the estimate is, or more specifically, how far off the estimation of the parameter is from the actual price. This indicates that the objective is to have the value of the estimate be as low as possible concerning the loss function.

Specifically, the study of features that characterizes conditions of competitive interdependence includes the opponents. To be more specific, a methodology based on a repeated game is used to examine cyber-attacks and model behaviors and research how defenders and attackers make decisions in an environment where they compete. The unique approach developed can forecast the future moves in the game to generate the proper countermeasures and apply the most acceptable cyber defense tactics that govern a company. This ability is based on the use of Bayesian inference, which is a type of statistical inference. The suggested system was tested with great success in a particular application scenario in the digital music sector and how to cope

with upcoming cyber-attacks. The testing was successful on both fronts.

A crucial step in further developing the proposed model is the further investigation of how they can adapt the method parameters to processes of modern and asynchronous change of the initial parameters of the evaluators. This is one of the processes that will further the development of the proposed model. As a result, the expansion and empirical exploration of the characteristics of the method estimators in finite samples, which call for the use of Monte Carlo simulations, is also a vital component of the proposed system's evolutionary parameter.

Data Availability

The data used in this study are available from the author upon reasonable request.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] S. Musman and A. Turner, "A game theoretic approach to cyber security risk management," *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, vol. 15, no. 2, pp. 127–146, 2018.

[2] L. Grønbæk, M. Lindroos, G. Munro, and P. Pintassilgo, "Basic concepts in game theory," in *Game Theory and Fisheries Management Game Theory and Fisheries Management: Theory and Applications*, L. Grønbæk, M. Lindroos, G. Munro, and P. Pintassilgo, Eds., pp. 19–30, Springer International Publishing, Berlin, Germany, 2020.

[3] D. A. Akinwumi, G. B. Iwasokun, B. K. Alese, and S. A. Oluwadare, "A review of game theory approach to cyber security risk management," *Nigerian Journal of Technology*, vol. 36, no. 4, p. 1271, 2018.

[4] T. Alpcan, Y. Vorobeychik, J. S. Baras, and G. Dán, "Decision and game theory for security," in *Proceedings of the 10th international conference, GameSec 2019*, Stockholm, Sweden, November 2019.

[5] C. T. Do, N. H. Tran, C. Hong et al., "Game theory for cyber security and privacy," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–37, 2018.

[6] I. M. del Águila and J. del Sagrado, "Bayesian networks for enhancement of requirements engineering: a literature review," *Requirements Engineering*, vol. 21, no. 4, pp. 461–480, 2016.

[7] A. Laszka, Y. Vorobeychik, and X. Koutsoukos, "Optimal personalized filtering against spear-phishing attacks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Austin, TX, USA, February 2015.

[8] A. Schlenker, H. Xu, M. Guirguis et al., "Don't bury your head in warnings: a game-theoretic approach for intelligent allocation of cyber-security alerts," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 381–387, Melbourne, Australia, August 2017.

[9] T. T. Nguyen and V. J. Reddi, "Deep reinforcement learning for cyber security," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–17, 2021.

[10] T. Alpcan and T. Basar, "Network security," Cambridge University Press, Cambridge, NY, USA, 2010.

[11] E. Hemberg, L. Zhang, and U.-M. O'Reilly, "Exploring adversarial artificial intelligence for autonomous adaptive cyber defense," in *Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., pp. 41–61, Springer International Publishing, Berlin, Germany, 2020.

[12] S. Ganzfried, "Computing Nash equilibria in multiplayer DAG-structured stochastic games with persistent imperfect information," in *Decision and Game Theory for Security*-Springer, pp. 3–16, Berlin, Germany, 2021.

[13] L. Huang and Q. Zhu, "Strategic learning for active, adaptive, and autonomous cyber defense," in *Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., Springer International Publishing, pp. 205–230, Berlin, Germany, 2020.

[14] P. Peng Xie, J. H. Li, X. Xinming Ou, P. Peng Liu, and R. Levy, "Using Bayesian networks for cyber security analysis," in *Proceedings of the 2010 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN)*, pp. 211–220, Chicago, IL, USA, June 2010.

[15] Y. F. Dolgii and P. G. Surkov, "Asymptotics of regularized solutions of a linear nonautonomous system of advanced differential equations," *Differential Equations*, vol. 46, no. 4, pp. 470–488, 2010.

[16] A. R. Butler, T. H. Nguyen, and A. Sinha, "Countering attacker data manipulation in security games," in *Decision and Game Theory for Security*, pp. 59–79, Springer, Berlin, Germnay, 2021.

[17] T. Halabi, M. Bellaiche, and A. Abusitta, "A cooperative game for online cloud federation formation based on security risk assessment," in *Proceedings of the 2018 4th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 83–88, Shanghai, China, June 2018.

[18] A. Schlenker, O. Thakoor, H. Xu, F. Fang, M. Tambe, and P. Vayanos, "Game theoretic cyber deception to foil adversarial network reconnaissance," in *Adaptive Autonomous Secure Cyber Systems Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., pp. 183–204, Springer International Publishing, Berlin, Germany, 2020.

[19] H. Worthington, R. S. McCrea, R. King, and R. A. Griffiths, "Estimation of population size when capture probability depends on individual states," *Journal of Agricultural, Biological, and Environmental Statistics*, vol. 24, no. 1, pp. 154–172, 2019.

[20] Z. Ni, Q. Li, and G. Liu, "Game-model-based network security risk control," *Computer*, vol. 51, no. 4, pp. 28–38, 2018.

[21] M. Redmayne, "Objective probability and the assessment of evidence," *Law, Probability and Risk*, vol. 2, no. 4, pp. 275–294, 2003.

[22] M. N. Alsaleh and E. Al-Shaer, "Automated cyber risk mitigation: making informed cost-effective decisions," in *Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., Springer International Publishing, pp. 131–157, Berlin, Germany, 2020.

[23] S. Venkatesan, S. Sugrim, J. A. Youzwak, C.-Y. J. Chiang, and R. Chadha, "A framework for studying autonomic computing models in cyber deception," in *Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., pp. 89–107, Springer International Publishing, Berlin, Germany, 2020.

[24] J. Jia and N. Z. Gong, "Defending against machine learning based inference attacks via adversarial examples: opportunities and challenges," in *Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., Springer International Publishing, pp. 118–307, Berlin, Germany, 2020.

[25] A. K. Guts, "Models of flood-attacks, mathematical catastrophe theory, theory of differential games and security strategies," in *Proceedings of the 2020 Dynamics of Systems, Mechanisms and Machines (Dynamics)*, pp. 1–5, Omsk, Russia, August 2020.

[26] Y. Urakawa, "Application of limited Pole-placement method to state feedback system," in *Proceedings of the 2021 IEEE International Conference on Mechatronics (ICM)*, pp. 1–5, Kashiwa, Japan, March 2021.

[27] B. Nugraha, N. Kulkarni, and A. Gopikrishnan, "Detecting adversarial DDoS attacks in software- defined networking using deep learning techniques and adversarial training," in *Proceedings of the 2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, pp. 448–454, Rhodes, Greece, July 2021.

[28] M. Papastergiou, D. Kanaros, A. Papamichou, and N. Vernadakis, "Effects of a project based on mobile applications, exergames and a web 2.0 social learning platform on students' physical activity and nutritional criteria in the era of COVID 19," *Educational Media International*, vol. 58, no. 4, pp. 297–316, 2021.

[29] R. M. El-Sagheer, M. A. W. Mahmoud, and H. M. Hasaballah, "Bayesian estimations using MCMC approach under three-parameter burr-XII distribution based on unified hybrid censored scheme," *Journal of Statistical Theory and Practice*, vol. 13, no. 4, p. 65, 2019.

[30] A. J. M. Garrett, "Review: probability theory: the logic of science, by E. T. Jaynes," *Law, Probability and Risk*, vol. 3, no. 3-4, pp. 243–246, 2004.

[31] S. Frey, A. Rashid, P. Anthonysamy, M. Pinto-Albuquerque, and S. A. Naqvi, "The good, the bad and the ugly," in *Proceedings of the 2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*, p. 496, Gothenburg, Sweden, February 2018.

[32] C. A. Coelho, "The generalized integer gamma distribution-A basis for distributions in multivariate statistics," *Journal of Multivariate Analysis*, vol. 64, no. 1, pp. 86–102, 1998.

[33] C. Sangüesa, "Error bounds in approximations of random sums using gamma-type operators," *Insurance: Mathematics and Economics*, vol. 42, no. 2, pp. 484–491, 2008.

[34] T. W. Anderson, *An Introduction To Multivariate Statistical Analysis*, Wiley, Hoboken, NJ, USA, 2003.

[35] J. O. Berger, "Bayesian analysis," in *Statistical Decision Theory and Bayesian Analysis*, J. O. Berger, Ed., Springer, pp. 118–307, New York, NY, USA, 1985.

[36] J. O. Berger, "Basic concepts," in *Statistical Decision Theory and Bayesian Analysis*, J. O. Berger, Ed., Springer, pp. 1–45, New York, NY, USA, 1985.

[37] E.-H. Choi, T. Fujiwara, and O. Mizuno, "Weighting for combinatorial testing by bayesian inference," in *Proceedings of the 2017 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pp. 389–391, Tokyo, Japan, March 2017.

[38] Z. Yu, T. Huang, and J. K. Liu, "Implementation of bayesian inference in distributed neural networks," in *Proceedings of the 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, pp. 666–673, Valladolid, Spain, March 2018.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3762755, 9 pages
https://doi.org/10.1155/2022/3762755

Research Article

Optimization of Cyber Tactics in Sports Strategies Using Hybrid AI Decision-Making Technologies

Meiling Duan

Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Meiling Duan; duanmeiling1974@126.com

Received 4 May 2022; Revised 23 May 2022; Accepted 25 May 2022; Published 13 June 2022

Academic Editor: Konstantinos Demertzis

One of the main problems of modern research concerns the optimal design of solutions to address cybersecurity problems, with methods that provide the ability to choose an objective function different from that of the classic problem of more economical design while allowing the use of constraints on any possible variable during planning, including financial resources. The number of corresponding solutions concerns the issue of optimal design of security strategies with a simulation that lies in game theory, with players, the defender on one side defending the system managing the available options of strategic solutions, and the attacker, who chooses the way to strike the system, based on some attack scenarios that cannot be easily predicted. The inherent difficulty of implementing the proposed solutions lies in the combined explosion of all possible combinations that make up the solution space, the complete examination of which requires a lot of computational time and computing resources, to the point that their use becomes unprofitable. This weakness is attributed to even minor problems, and the possible strategies available to the defender are finite but at the same time numerous. To solve the abovementioned problem, the work proposes a hybrid system that aims to identify the best possible approach in the theoretically optimal solution in a short time and with minimal computing resources. Specifically, a heuristic optimization methodology is used with overlapping answers between two contiguous neighborhoods based on the Bloom Filters structure that supports fast listings and searches. This methodology, which is evaluated in optimizing safety strategies in the sports industry, brings about 40% optimization.

1. Introduction

Game theory is the branch that deals with the analysis and evaluation of games if players behave logically. This definition distinguishes the term "game," which is most often used in everyday life and has mainly a recreational meaning. A game is any situation in which two or more people, called players, are called upon to make one or more decisions, depending on which event will occur, which has a different value for each player [1].

The definition of *utility* is perhaps one of the most fundamental concepts in game theory [2], as it dramatically facilitates their handling and analysis. It was designed to determine a person's level of contentment as a consequence of a particular outcome [3, 4]. More specifically, utility is an arbitrary measure of satisfaction that aims to quantify the effect of an event on a person's happiness. Like any size, the utility has its unit of measurement, the *util*. Of course, *util* has no physical substance, but they serve as units of measurement of utility. The assignment of a quantity to the utility that a person derives is the function of the *individual's utility function*. According to Game theory, even when they are not consciously aware of it, people strive to maximize the utility function associated with themselves [3, 5]. This, in many cases, coincides with other objectives that are readily apparent [6].

Game theory is the ultimate modeling system for security systems and, more recently, cybersecurity. It allows the creation of tangible solutions that will enable the evaluation of existing strategies and their optimization to create a robust and long-term security environment at the organization level [7]. Using the principles of game theory [8], it is possible to develop cyber-threat scenarios where cyber security professionals can apply the strategies that govern the

organization and control or quantify the risk to their valuable assets [1]. They can also use areas with a low level of risk to maximize the return on their investments. As a result, using specialized scenarios based on game theory, it is possible to predict the attackers' strategy at each stage of the attack cycle, assisting in developing intelligent models to improve cybersecurity and creating new intelligent systems to defraud the attackers [9].

Nevertheless, the inherent difficulty of implementing the proposed solutions that may arise from performing tests—simulations based on game theory, lies in the combined explosion of all possible combinations that make up the solution space, the entire examination of which requires a considerable computer time and corresponding computing resources, to the point that their use becomes unprofitable [5]. This weakness is because, even for minor problems, the possible strategies available to the defender are finite but at the same time numerous. For this reason, the work proposes a hybrid system [10, 11], which aims to identify the best-possible solutions in the area of theoretically optimal solutions in a short time and with minimal computing resources. Specifically, a heuristic optimization [12, 13] methodology is used with overlapping solutions between two contiguous neighborhoods based on the Bloom Filters structure that supports fast listings and searches [14, 15]. This methodology, which has been tested in optimizing safety strategies in the sports industry, achieves up to 40% optimization compared to other methods.

2. Relevant Publications

Artificial intelligence in cybersecurity is a concept that is continuously evolving. The literature on decision-making, deep learning, and game theory focuses on utilizing different concepts to efficiently solve complex real-world cybersecurity problems [3, 16, 17].

Das and Sandhane [18] offered a concise summary of AI applications of major cybersecurity solutions and assessed the potential for boosting cybersecurity capabilities via defensive mechanism enhancements. To begin, neural networks are utilized for safeguarding the periphery and a variety of other protection domains. On the other hand, it was evident that some cybersecurity issues could be resolved effectively only via artificial intelligence technologies. For example, thorough intelligence is critical for strategic decision-making, and logical judgment support is one of the unresolved protection concerns. While neural networks were not the best technology for many applications, advanced cybersecurity measures remained necessary. These domains included decision support, situational awareness, and data access.

Johnson [19] examined the influence of AI on strategic stability, focusing on the dangers and trade-offs associated with predelegating military power (or automating aggression) to robots. He contended that AI-enabled decision-support tools—supplanting human analytical reasoning, compassion, inventiveness, and imagination in the strategic judgment method—would be profoundly disastrous if defense planners came to view AI support' function as a magic bullet human assessment and decision-cognitive making's inadequacies. Additionally, the article discussed the malicious use of artificial intelligence-enhanced fake reports, deepfakes, bots, and other forms of social media by nonstate actors and state proxy actors, which may produce states to overestimate a threat by unclear or exploited information, thereby growing destabilization.

Alpcan and Basar [20] systematically sought to provide a conceptual framework for making resource allotment choices that balance existing skills and perceived security issues in their literature about network security and game-theoretic techniques. They concentrated on applying game, data, interaction, efficiency, selection, and control theories to various security difficulties. Simultaneously, links between conceptual models and real-world security issues are emphasized to generate a key review loop between principles and application.

Nguyen and Reddi [21] investigated the application of Deep Reinforcement Learning (DRL) approaches in cyber warfare. They explored a variety of vital subjects, including DRL-based defense techniques for cyber-physical assets, autonomous intrusion discovery, and multiagent DRL-based game theory simulations for cyberattack defense measures. Furthermore, comprehensive arguments and possible research directions on Internet security emphasize DRLs are offered. They hoped that this extensive review would provide a framework for and motivate further research into the capability of developing DRL to deal with increasingly advanced digital confidentiality complications.

Schlenker et al. [8] investigated the fundamental inherent problem of assigning cyber warnings to a small number of security experts. They investigated this issue using the Cyber-alert Allocation Game and demonstrated how to compute the defender's best options. They proposed a novel method for dealing with concerns about implement ability when determining the defender's most acceptable marginal technique to overcome this game. Finally, they provided heuristics for resolving large games similar to those depicted, and an objective assessment of the methodology and treatment approaches proposed.

3. Definition of the Problem

Even for a system under study related to the sports industry, the security of which requires the design of a system of strategies, specific decisions need to be identified [7, 19]. Let $(a_1, a_2, \ldots, a_n)$ be the vector of the above decisions, which are also called design variables, and can be any system design security decisions. The security engineer is asked to decide the best possible system design [22], given the constraints imposed by the nature of the system, security policies applied, system functionality, users, etc. It should be noted that the variables can differ depending on their nature, the point of danger presented by change, and the problem that the security engineer is called to solve; for example, the problem can be size optimization, topology optimization, system optimization, financial optimization, and so on [23]. With the proposed method, it is possible to use game theory to

solve these problems and, at the same time, identify combinations thereof [9].

The problem can be analyzed in two substages: the construction of the player's earnings registers and the analysis of the game to select the best strategy [24] for the defending player (i.e., the security engineer). Specifically, the first step, in particular, forecasts what will occur in each case of travel planning (as travel is considered as the operating cost for each action, with a positive direction as the tolerable cost while a negative approach is the opposite) [1, 25, 26]. Figure 1 shows the travel benefit function.

The mathematical expression of the scenario examined, as shown in Figure 1 is [2, 4, 26]:

$$u_{d,i}(d_i) = \begin{cases} a^+_{d,i} \cdot d_i,\ 0 \le d_i \le d^+_{i,lim}, \\ a^-_{d,i} \cdot |d_i|,\ d_{i,lim} \le d_i < 0, \\ a^+_{d,i} \cdot d^+_{i,lim} + p_d + a^+_{d,i} \cdot \left(d_i - d^+_{i,lim}\right)^k,\ d_i > d^+_{i,lim}, \\ a^-_{d,i} \cdot \left|d^-_{i,lim}\right| + p_d + a^-_{d,i} \cdot \left(d^-_{i,lim} - d_i\right)^k,\ d_i < d^-_{i,lim}, \end{cases} \tag{1}$$

$u_{d,i}(d_i)$: is the utility function for a particular travel d_i, $d^+_{i,lim}$: is the limit of travel of a degree of freedom in a positive travel, which is desirable not to be exceeded; $d^-_{i,lim}$: is the limit of travel of any degree of freedom in the negative direction, which is desirable not to be exceeded; p_d: is the penalty imposed if one of the travel limits for this degree of freedom is surpassed. Its price is negative, $a^+_{d,i}$: a coefficient showing the linear change of the benefit for small values of the d_i travel when it has a positive direction. It is strictly negative and indicates the preference of player A for more minor travels than larger ones, $a^+_{d,i}$: coefficient showing the linear change of the benefit for small values of the d_i movement when it has a negative direction. It is strictly negative and indicates the preference of player A for smaller movements over larger ones, k: indicator showing how the utility changes for d_i movement values greater than the corresponding desired limit.

Respectively, the following function determines the benefit due to trends (as trends are considered the operating cost for each possible action, with a positive direction indicating the modest cost and a negative direction meaning the opposite) [27–29]:

$$u_{\sigma,i}(\sigma_i) = \begin{cases} a^+_{\sigma,i} \cdot \sigma_i,\ 0 \le \sigma_i \le \sigma^+_{i,lim}, \\ a^-_{\sigma,i} \cdot |\sigma_i|,\ \sigma^-_{i,lim} \le \sigma_i < 0, \\ a^+_{\sigma,i} \cdot \sigma^+_{i,lim} + p_\sigma + a^+_{\sigma,i} \cdot \left(\sigma_i - \sigma^+_{i,lim}\right)^k,\ \sigma_i > \sigma^+_{i,lim}, \\ a^-_{\sigma,i} \cdot \left|\sigma^-_{i,lim}\right| + p_\sigma + a^-_{\sigma,i} \cdot \left(\sigma^-_{i,lim} - \sigma_i\right)^k,\ \sigma_i < \sigma^-_{i,lim}. \end{cases} \tag{2}$$

The overall benefit of the results is summarized in the equation:

$$u_{res} = w_d \cdot u_d + w_\sigma \cdot u_\sigma, \tag{3}$$

where

$$\begin{aligned} u_d &= \sum w_{d,i} \cdot u_{d,i}, \\ u_\sigma &= \sum w_{\sigma,i} \cdot u_{\sigma,i}. \end{aligned} \tag{4}$$

The sizes w_d and w_σ express the importance of movements and trends, respectively, in shaping the usefulness of the results. If restrictions are placed on only one of the two types of outcomes of interest, then this takes on total weight of 1 while the other is 0.

Respectively, the quantities u_d and u_σ express the gravity coefficients of the individual movements and stresses, i.e., the importance of any node movement or trend that develops in a member in the final utility configuration [23, 30]. For these rates, we have

$$\begin{aligned} \sum w_{d,i} &= 1, \\ \sum w_{\sigma,i} &= 1. \end{aligned} \tag{5}$$

The value of the travel or trend factor is determined by whether the node is committed to the specific movement or whether the particular member has a limit on the trend that develops. Thus, if the total number of travel constraints is denoted by $n_{c,d}$ and the number of constraints on trends by $n_{c,\sigma}$, then the individual importance factors will take values:

$$w_{d,i} = \begin{cases} \dfrac{1}{n_{c,d}}, \\ 0, \end{cases} \quad w_{\sigma,i} = \begin{cases} \dfrac{1}{n_{c,\sigma}}, \\ 0. \end{cases} \tag{6}$$

The preferences of player A must also consider the fact that in addition to smaller transfers that are equivalent to the minimum operating cost, it must also seek the lowest possible financial cost. Therefore, it is necessary to determine the utility that it derives due to the financial cost. This is defined as [9, 17, 31, 32]:

$$u_{de\ s} = \begin{cases} a_{\text{cost}} \cdot \text{cost},\ \text{cost} \le \text{budget}, \\ a_{\text{cost}} \cdot \text{cost} + p_{\text{budget}},\ \text{cost} > \text{budget}, \end{cases} \tag{7}$$

where cost is the total cost of the security plan, a_{cost} is a negative factor that indicates the preference of player A for cheaper modes of action than more expensive, and p_{budget} is a penalty imposed if a permissible cost limit is exceeded.

The overall benefit enjoyed by player A will result from the simultaneous action of the benefits due to design and results. The degree to which each affects the result depends on the user and is what will largely determine the result of the optimal strategy:

$$u_A = w_{res} \cdot u_{res} + w_{de\ s} \cdot u_{de\ s}, \tag{8}$$

where $w_{res,} w_{de\ s}$ are the weights that the user assigns to the analysis and design results, respectively. Of course, they must satisfy the property:

$$w_{res} + w_{de\ s} = 1. \tag{9}$$

In the usual case, the importance and the results of the analysis are given equal importance, so the rule is

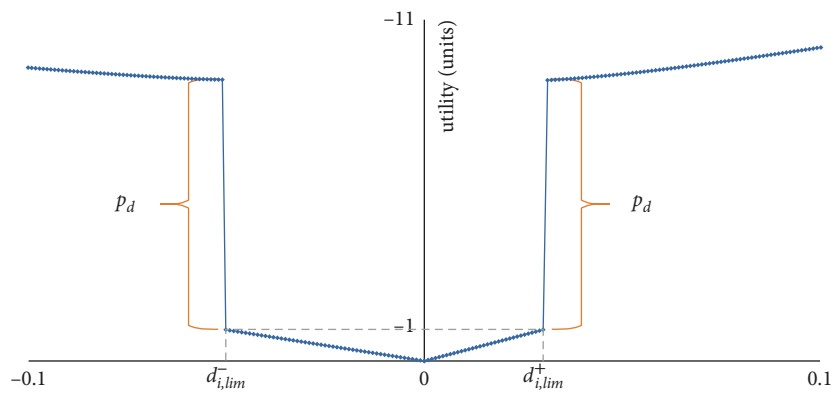


Figure 1: Travel scenario utility function.

$$w_{res} = w_{de\ s} = \frac{1}{2}. \tag{10}$$

The utility function of player B, which requires action as it creates security incidents, is taken equal to and opposite to the utility function of player A so that the benefit of one player is to the detriment of the opponent:

$$u_B = -u_A. \tag{11}$$

The earnings of player A for each mitigation system design and each decision form the register of the game's profits, from the solution of which will emerge the best-defense strategy of the system in each phase of the attack [9, 20, 33]. We assume that the players' utility functions are equal and opposite for this modeling, which places the game in the category of zero-sum games. In this type of game, each outcome is antagonistic for both players, in the sense that what one player "wins," the other "loses."

4. Optimization

In the previous section, we developed a methodology for solving security solution design problems. Mathematical optimization involves selecting the optimal component concerning a given criterion from a given group of potential solutions. Problems relating to optimization appear in each of the quantitative subfields, including computer science and engineering, as well as operations research and economics. An optimization problem can be simplified to its most basic form by stating that its solution is to maximize or minimize an objective function. This can be accomplished by methodically selecting input values from within an allowable set and computing the value of the process. A significant portion of applied mathematics uses optimization theory and methods for different formulations. In a broader sense, optimization refers to determining the "best available" values of a particular objective function given a specific domain (or input). This process might involve various objective functions and several different disciplines.

Solution extraction consists of finding the line of the usual form table of the corresponding game with the largest column minimum [7, 34]. To make this possible, the standard game board must be constructed. All possible design options the defender can choose (strategies) must be listed and analyzed for each possible decision case, and then the utility table and the game in its standard form must be developed to follow the resolution process [35, 36].

The above process presents significant computational issues [37–39]. When the construction has m groups of members, N^m possible choices for each member and L possible decision-making cases, the possible designs are N^m, and the static analyzes to be performed are $N^m \times \mathrm{L}$. This size is too large even for medium-sized problems. For example, if $m = 8$, $N = 20$, $L = 5$, then a total of $20^8 \times 5 = 1.28 \times 10^{11}$ static analyzes must be performed. When one static solution takes around 0.015 seconds to execute, the total analysis time is $1.28 \times 10^{11} \times 0.015 = 1.92 \cdot 10^9$ seconds, i.e., about 61 years.

On the other hand, even if the analysis time is reduced, the algorithm's complexity remains enormous. The number of member groups is the most decisive factor, as it exponentially increases the size of the problem. For example, if the member groups in the above scenario increase to 9 instead of 8, the estimated solution time will exceed 1200 years. However, a mediocre computer will face memory problems in addition to the problem of resolution time, as it is required to store logs with a large number of data, making it impossible to solve the problem in this manner. So, as it is understood, the need for a faster solution is imperative, so there is a shift to heuristic algorithms [7, 9, 40].

The heuristic algorithm presented in this paper is based on the observation that at each step, most of the bioinspired heuristic algorithms define an area in which, after their analysis, they find the best solution, with the center forming a new neighborhood. However, when examining the new neighborhood, previously rejected solutions are re-examined. At best, half of the solutions of the previous neighborhood are re-examined [41–44]. This overlap of solutions is typically illustrated in Figure 2.

The above shows that the algorithm is called upon to repeat calculations for cases it has already considered in the previous steps. In many cases, this extra work is less laborious, as it is enough to simply calculate the candidate solution's cost. In other cases, however, the cost passes the first checkpoint of the algorithm. It is followed by static

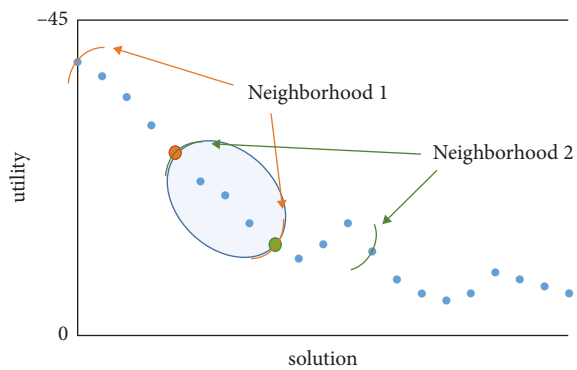


Figure 2: Overlap of solutions between two contiguous neighborhoods.

resolution and determination of the benefit, a more complicated process. In any case, the algorithm is procrastinating for no reason.

To address this flaw, it is reasonable to keep a record of the solutions that have been tested so that their analysis is not repeated in the subsequent steps. This record has three crucial features [7, 13, 37]:

(1) The number of solutions that will be recorded will be pretty large and therefore using the typical list will consume a significant part of the computer memory.

(2) The number of recordings cannot be determined in advance. This number can be estimated, but not accurately. This means that the number of positions in a list that includes them cannot be determined.

(3) The registration and search process for verification may not be fast enough.

This paper uses bloom filters to solve all of these problems.

5. Bloom Filters

Bloom Filters [45] are data structures that have an advantage over other data lists. They are more efficient in memory space, allowing for a chance of error when searching within logs, offering significant gains in complex applications [34, 46]. The structure of a Bloom Filter supports very fast listings and searches. Their most important advantage lies in their much better performance in a large volume of data and the minimal memory space used per registered object. Their use is ideal for applications in which one wants to check if a value is included in a list, but for bloom filters to make sense, it must first be demonstrated that there is a possibility of limiting the likelihood of error [12, 47, 48]. The more memory space available, the more this possibility will tend to be eliminated. Also, a key issue is the determination of the appropriate parameters to optimize the efficiency of the bloom filter, i.e., to achieve low-memory usage with an acceptable error rate at the same time [45, 49].

The following assumption is made for the proposed implementation: all matching mechanisms operate in a completely random manner. This means that the positions (in series of bits) identified by matching mechanisms acting on a specific object follow a uniform distribution and are entirely independent of the other entries. Given the use of n bits and the input of a set of data S, set s, it is desirable to determine the error probability. A question that arises is, what is the expected number of values of "1" after introducing all objects. To resolve this, a specific position in the string of bits will be examined. The probability that the value 1 will exist after the entries are in that position will be calculated [37, 50].

Initially, the probability that this bit will have the value 0 after the entries will be determined. The probability of getting the value 1 by the action of a matching mechanism is only $1/n$, so the probability of staying 0 is $(1-1/n)$. Therefore, for k matching mechanisms and s number of objects, the chance becomes $(1-1/n)^{k\cdot s}$ and finally, the probability for each bit to be set "1" after all entries will be [36, 44, 51, 52]

$$1-\left(1-\frac{1}{n}\right)^{k\cdot s}. \tag{12}$$

Observing Figure 3, it is evident that the function $y_1 = 1+x$ is upper blocked by $y_2 = e^x$ since $e^x \geq 1+x, \quad \forall x \in R$, while for $x \longrightarrow 0 \Rightarrow e^x \simeq 1+x$.

Therefore, the above probability, respectively, will apply:

$$1-\left(1-\frac{1}{n}\right)^{k\cdot s} \simeq 1-e^{-k\cdot s/n}, \tag{13}$$

and because $b = n/s$, where b are the bits per object, eventually the probability is configured as follows:

$$1-\left(1-\frac{1}{n}\right)^{k\cdot s} \simeq 1-e^{-(k/b)}. \tag{14}$$

From this relationship, it is now clear that as the number of bits per object increases, the probability of a particular bit becoming "1" tends to zero.

Now, for example, an object ε, $x \notin S$ is examined. To make a mistake, all k bits in the corresponding positions must have the value 1, so this probability is [9, 47, 53, 54]:

$$\varepsilon \simeq \left[1-e^{-(k/b)}\right]^k. \tag{15}$$

So in order to determine parameters that result in a small tolerable error while using a small memory space, the number of bits per object must be determined, i.e., b. For given b, the error ε turns to be minimized for a value of $k \simeq \ln 2 \cdot b$. k must be an integer, extracted after rounding the above number. This is how it turns out [47, 55, 56]:

$$\varepsilon \simeq \left(\frac{1}{2}\right)^{\ln 2\cdot b}. \tag{16}$$

This equation can be expressed in terms of b, so for a given error, the bits per entry are calculated:

$$b \simeq 1.44 \cdot \log_2\left(\frac{1}{\varepsilon}\right). \tag{17}$$

For example, for 8 bits per entry is calculated:

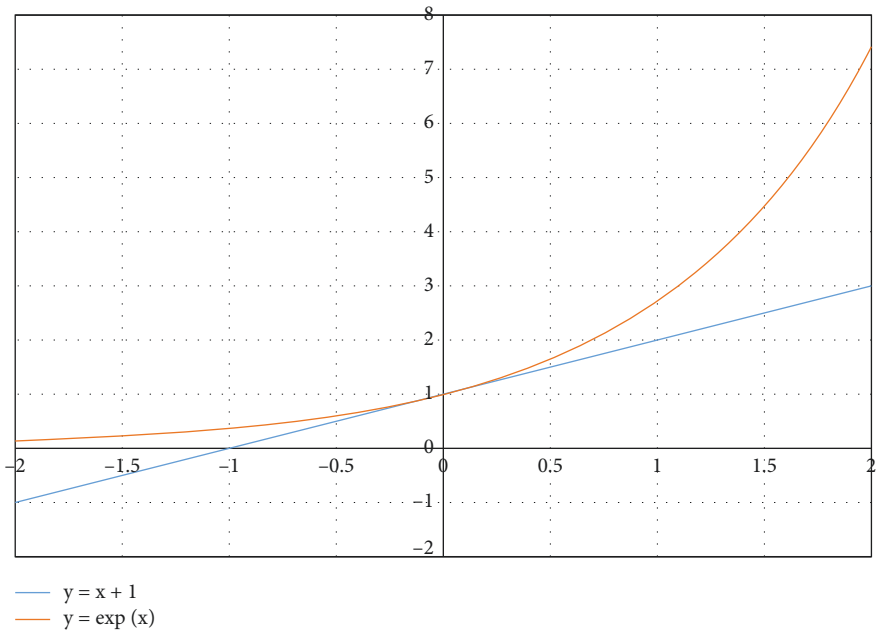


Figure 3: The functions $y_1 = 1 + x, y_2 = e^x$.

$$k \simeq \ln 2 \cdot 8 \simeq 5.54 \Rightarrow k = 5. \tag{18}$$

And it turns out:

$$\varepsilon \simeq 2\%. \tag{19}$$

Indicatively, it is stated that if the bits per entry are doubled to 16, then this probability becomes approximately:

$$\varepsilon \simeq 0.4‰. \tag{20}$$

6. Conclusions

The inherent difficulty of applying the proposed security solutions to cybersecurity problems lies in the situations characterized by the combined explosion of all possible combinations that make up the solution space. Developing and finding the optimal solution takes a significant amount of computational time and computing resources to the point where their use is frequently unprofitable, especially in high-risk rearranged environments. This weakness exists because, even for minor issues, the defender's available strategies are limited but numerous. For this reason, the paper proposed a hybrid system of heuristic intelligent algorithms, which aims to identify, in a short time and with minimal computing resources, the best-possible solutions in the area of the theoretically optimal solution. As it turned out experimentally, by utilizing bloom filters, the system significantly reduced computing time and the corresponding required resources.

The bloom filters are both in complex applications and in a simple use such as the one that the improved gradual impairment algorithm is called to do. For this problem, some remarks are made [34, 39, 57, 58]:

(1) It has already been stated that we insert values into a bloom filter rather than items. On the other hand, each potential answer is fundamentally a list, or more specifically, an object that includes the serial numbers of the individual member groups. The utilization of this solution's serial number enables the execution of the procedures mentioned earlier. Because of this, the bloom filter will enter the serial number of the resolution, which is going to be different for every key, and then based on this. The entries are going to be searched again.

(2) The little chance of making a mistake, which is already very low, is not a cause for concern because it is highly improbable that this will result in a significant error. In particular, when the bloom filter is looking for a solution, it may believe that it already exists in its list. As a result, the answer will not be subjected to further analysis. It would be a terrible turn of events if this (very uncommon) instance of wrongfully rejected evidence turned out to be the best.

(3) The bloom filter was applied to the issues that arose and provided conclusive evidence of its speed and general utility. To be more specific, its use prevents the study of 20–40% of the total solutions that were initially investigated, which results in a proportional reduction in the amount of time needed for computing.

There are some disadvantages to using bloom filters in specializing and evaluating the experimental process. Specifically, we cannot register objects, nor pointers to objects, only values. So, during the search, a check is made whether a value has been met or not in the registration phase. A second disadvantage is that deletions from the entries are not allowed. Finally, the most critical weakness is the possibility

of error. If a value has been entered, there is no way it can be mistakenly considered that it is not included in the entries. On the contrary, there is a particular possibility that a value has not been recorded, and the bloom filters during the search process claim that it has been found in the list. These drawbacks are also research questions that will be addressed in future extensions of this work.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] D. A. Akinwumi, G. B. Iwasokun, B. K. Alese, and S. A. Oluwadare, "A review of game theory approach to cyber security risk management," *Nigerian Journal of Technology*, vol. 36, no. 4, p. 1271, 2018.

[2] M. D. Mednikov, N. A. Sokolitsyna, A. S. Sokolitsyn, and V. P. Semenov, "Game theory model of forming enterprise development strategy in market environment uncertainty," in *Proceedings of the 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM)*, pp. 876–878, St. Petersburg, Russiadoi, May 2017.

[3] J. P. Hespanha, "Sensor manipulation games in cyber security," *Game Theory and Machine Learning for Cyber Security*, pp. 137–148, 2021.

[4] Manisha and N. P. Singh, "Efficient network selection using game theory in a heterogeneous wireless network," in *Proceedings of the 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1–4, IEEE, Madurai, India, March 2015.

[5] S. Frey, A. Rashid, P. Anthonysamy, M. Pinto-Albuquerque, and S. A. Naqvi, "The Good, the Bad and the Ugly: A Study of Security Decisions in a Cyber-Physical Systems Game," in *Proceedings of the 40th International Conference on Software Engineering*, p. 496, December 2018.

[6] S. Yang, "An approach on attack path prediction modeling based on game theory,"vol. 5, pp. 2604–2608, in *Proceedings of the 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 5, IEE, Chongqing China, April 2021.

[7] S. Roy, S. U. Kadir, Y. Vorobeychik, and A. Laszka, "Strategic remote attestation: testbed for internet-of-things devices and stackelberg security game for optimal strategies," *Lecture Notes in Computer Science*, vol. 13061, pp. 271–290, 2021.

[8] A. Schlenker, H. Xu, M. Guirguis et al., "Don't bury your head in warnings: a game-theoretic approach for intelligent allocation of cyber-security alerts," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 381–387, Melbourne, Australia, August 2017.

[9] A. R. Butler, T. H. Nguyen, and A. Sinha, "Countering attacker data manipulation in security games," *Lecture Notes in Computer Science*, vol. 13061, pp. 59–79, 2021.

[10] B. Huang, Y. Sun, Y.-M. Sun, and C.-X. Zhao, "A hybrid heuristic search algorithm for scheduling FMS based on Petri net model," *International Journal of Advanced Manufacturing Technology*, vol. 48, no. 9-12, pp. 925–933, 2010.

[11] S. Olyaee, R. Ebrahimpur, and S. Esfandeh, "A hybrid genetic algorithm-neural network for modeling of periodic nonlinearity in three-longitudinal-mode laser heterodyne interferometer," in *Proceedings of the 2013 21st Iranian Conference on Electrical Engineering (ICEE)*, pp. 1–5, IEE, Mashhad, Iran, May 2013.

[12] C. G. Zhai, X. B. Jiang, Y. X. Zhang, and N. Liu, "Research on the optimization of military supplies under big data background," in *Proceedings of the 2018 International Conference on Big Data and Artificial Intelligence (BDAI)*, pp. 18–23, IEEE, Beijing China, June 2018.

[13] P. Lin, "Research on optimization of distributed big data real-time management method," in *Proceedings of the 2018 3rd International Conference on Smart City and Systems Engineering (ICSCSE)*, pp. 626–630, IEEE, Xiamen China, September 2018.

[14] D.-C. Chang, C. Chen, and M. Thanavel, "Dynamic Reordering Bloom Filter," in *Proceedings of the 2017 19th Asia-Pacific Network Operations And Management Symposium (APNOMS)*, pp. 288–291, IEE, Seoul, Korea (South), September 2017.

[15] J. Lee and H. Lim, "A circled Bloom filter for the membership identification of multiple sets," in *Proceedings of the 2019 International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–3, IEEE, Auckland, New Zealand, January 2019.

[16] S. Cano-Berlanga, J.-M. Giménez-Gómez, and C. Vilella, "Enjoying cooperative games: the R package GameTheory," *Applied Mathematics and Computation*, vol. 305, pp. 381–393, 2017.

[17] A. Eldosouky and S. Sengupta, "Moving Target defense games for cyber security: theory and applications," in *Game Theory and Machine Learning for Cyber Security*, pp. 160–179, IEEE, 2021.

[18] R. Das and R. Sandhane, "Artificial intelligence in cyber security," *Journal of Physics: Conference Series*, vol. 1964, no. 4, Article ID 042072, 2021.

[19] J. Johnson, "Delegating strategic decision-making to machines: dr. Strangelove Redux?" *Journal of Strategic Studies*, vol. 45, no. 3, pp. 439–477, 2022.

[20] T. Alpcan and T. Basar, "Network security," *Network Security: A Decision and Game-Theoretic Approach*, Cambridge University Press, Cambridge NY, USA, 2010.

[21] T. T. Nguyen and V. J. Reddi, "Deep reinforcement learning for cyber security," *IEEE Transactions on Neural Networks and Learning Systems*, no. –17, pp. 1–17, 2021.

[22] A. K. Guts, "Models of Flood-attacks, mathematical catastrophe theory, theory of differential games and security strategies," in *Proceedings of the 2020 Dynamics of Systems, Mechanisms and Machines (Dynamics)*, pp. 1–5, Omsk, Russia, August 2020.

[23] L. Huang and Q. Zhu, "Strategic learning for active, adaptive, and autonomous cyber defense," in *Adaptive Autonomous Secure Cyber Systems*, S. Jajodia, G. Cybenko, V. S. Subrahmanian, V. Swarup, C. Wang, and M. Wellman, Eds., pp. 205–230, 2020.

[24] K. Jin, T. Yin, C. A. Kamhoua, and M. Liu, "Network games with strategic machine learning," *Lecture Notes in Computer Science*, pp. 118–137, 2021.

[25] Z. Ni, Q. Li, and G. Liu, "Game-Model-based network security risk control," *Computer*, vol. 51, no. 4, pp. 28–38, 2018.

[26] S. V. Dugani, S. Dixit, and M. Belur, "Automated adaptive sequential recommendation of travel route," in *Proceedings of the 2017 International Conference on Computing*

Methodologies and Communication (ICCMC), pp. 284–288, IEEE, Erode India, July. 2017.

[27] Z. Li, Y. Liu, D. Liu, N. Zhang, D. Lu, and X. Huang, "A security defense model for ubiquitous electric internet of things based on game theory," in *Proceedings of the 2020 IEEE 4th Conference on Energy Internet and Energy System Integration (EI2)*, pp. 3125–3128, IEEE, Wuhan, China, July 2020.

[28] V. Torra, "Privacy models and disclosure risk measures," in *Data Privacy: Foundations, New Developments and the Big Data Challenge*, V. Torra, Ed., Springer International Publishing, Cham, pp. 111–189, 2017.

[29] M. Iezzi, "Practical privacy-preserving data science with homomorphic encryption: an overview," in *Proceedings of the 2020 IEEE International Conference on Big Data (Big Data)*, pp. 3979–3988, GA, USA, September 2020.

[30] X. Z. Bao and X. F. Li, "Cost allocation of integrated supply based on shapely value method," in *Proceedings of the 2010 International Conference on Intelligent Computation Technology and Automation*, vol. 1, pp. 1054–1057, Changsha, China, May 2010.

[31] G. Singhal, R. R. Mainuddin, R. Rajesh, M. T. Beg, R. K. Tyagi, and A. L. Dawar, "Overview of optical techniques for characterization of high-power infrared gas lasers," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4165–4173, 2015.

[32] D. G. Bhoyar and U. Yadav, "Review of jamming attack using game theory," in *Proceedings of the 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1–4, Coimbatore India, March 2017.

[33] L. Luo, D. Guo, R. T. B. Ma, O. Rottenstreich, and X. Luo, "Optimizing bloom filter: challenges, solutions, and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1912–1949, 2019.

[34] M. Kwon, V. Shankar, S. Pontarelli, and P. Reviriego, "A fingerprint-based bloom filter with deletion capabilities," in *Proceedings of the 2019 European Conference on Networks and Communications (EuCNC)*, pp. 453–458, Valencia Spain, June 2019.

[35] M. Todorov Marinov, "A bloom filter application for processing big datasets through MapReduce framework," in *Proceedings of the 2021 International Conference on Information Technologies (InfoTech)*, pp. 1–5, IEE, Varna Bulgaria, Sep. 2021.

[36] H. Byun, S. Kim, C. Yim, and H. Lim, "Addition of a secondary functional bloom filter," *IEEE Communications Letters*, vol. 24, no. 10, pp. 2123–2127, 2020.

[37] R. Xie and M. Z. R. H. H. G. Li, "Hash adaptive bloom filter," in *Proceedings of the 2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pp. 636–647, Chania, Greece, April 2021.

[38] A. Bala, I. Ismail, R. Ibrahim, and S. M. Sait, "Applications of metaheuristics in reservoir computing techniques: a review," *IEEE Access*, vol. 6, pp. 58012–58029, 2018.

[39] T.-S. Chen and B.-H. Wu, "Gateway selection based on game theory in internet of things," in *Proceedings of the 2018 International Conference on Electronics Technology (ICET)*, pp. 403–406, IEE, Chengdu, Chinadoi, July 2018.

[40] C. Y. Tseung, K. P. Chow, and X. Zhang, "Extended abstract: anti-DDoS technique using self-learning bloom filter," in *Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, p. 204, Beijing, China, July 2017.

[41] S. Z. Kiss, E. Hosszu, J. Tapolcai, L. Ronyai, and O. Rottenstreich, "Bloom filter with a false positive free zone," in *Proceedings of the IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pp. 1412–1420, IEE, Honolulu, HI, USA, April 2018.

[42] A. Pagh, R. Pagh, and S. S. Rao, "An optimal bloom filter replacement," 2004, https://arxiv.org/abs/0804.1845.

[43] L. E. B. Salasar, J. G. Leite, and F. Louzada, "Likelihood-based inference for population size in a capture-recapture experiment with varying probabilities from occasion to occasion," *Brazilian Journal of Probability and Statistics*, vol. 30, no. 1, pp. 47–69, 2016.

[44] R. T. Kamurthi, S. R. Chopra, and R. Sharma, "Confrontation-Wi-Fi risks and data breach," in *Proceedings of the 2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, pp. 633–638, Pune, India, April 2021.

[45] A. Al-Badarneh, H. Najadat, and S. Rababah, "Performance evaluation of bloom filter size in map-side and reduce-side bloom joins," in *Proceedings of the 2017 8th International Conference on Information and Communication Systems (ICICS)*, pp. 165–170, IEE, Irbid Jordandoi, April 2017.

[46] H. Lim, J. Lee, and C. Yim, "Complement bloom filter for identifying true positiveness of a bloom filter," *IEEE Communications Letters*, vol. 19, no. 11, pp. 1905–1908, 2015.

[47] P. Liu, C. Jiang, X. Zhang, and W. Yu, "Compressed bloom filter method of dds middleware based on FPGA," in *Proceedings of the 2021 7th International Conference on Computer and Communications (ICCC)*, pp. 1143–1147, IEEE, Chengdu, China, December 2021.

[48] T. Varshney and K. Verma, "Rectifying flow of duplicacy using Bloom-filter," in *Proceedings of the 2017 International Conference on Computer, Communications and Electronics (Comptelix)*, pp. 300–304, IEEE, Jaipur India, August 2017.

[49] M. Burgin and P. Rocchi, "Ample probability in cognition," in *Proceedings of the 2019 IEEE 18th International Conference on Cognitive Informatics & Cognitive Computing (ICCI∗CC)*, pp. 62–65, New York City, NY, USA, July 2019.

[50] A. J. M. Garrett, "Review: probability theory: the logic of science," in *Law, Probability and Risk*, E. T. Jaynes, Ed., vol. 3, no. 4, pp. 243–246, 2004.

[51] A. R. Hota and S. Sundaram, "Interdependent security games on networks under behavioral probability weighting," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 262–273, 2018.

[52] Z. Sun, J.-S. Pan, C.-H. Chen, and T.-Y. Wu, "A probability-based analytical model based on deep learning for traffic information estimation," in *Proceedings of the 2020 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan)*, pp. 1-2, IEEE, Taoyuan, Taiwandoi, November 2020.

[53] P. M. Bala, S. Usharani, and M. Aswin, "IDS based fake content detection on social network using bloom filtering," in *Proceedings of the 2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, pp. 1–6, IEEE, Pondicherry,India, November 2020.

[54] Y. Wu and J. S. J. T. O. G. B. He, "Elastic bloom filter: deletable and expandable filter using elastic fingerprints," *IEEE Transactions on Computers*, vol. 71, no. 4, pp. 984–991, 2022.

[55] S. Li Fuyan and L. Fuyan, "Risk-considered Shapley profit allocation of innovative supply chain," in *Proceedings of the 2010 IEEE International Conference on Emergency Management and Management Sciences*, pp. 238–241, IEEE, Beijing, China, September 2010.

[56] S. Sengupta and A. Rana, "Role of bloom filter in analysis of big data," in *Proceedings of the 2020 8th International Conference on Reliability, Infocom Technologies and Optimization*

(Trends and Future Directions) (ICRITO), pp. 6–9, Noida, India, June 2020.

[57] K. Nandhini and R. Balasubramaniam, "Malicious website detection using probabilistic data structure bloom filter," in *Proceedings of the 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 311–316, IEEE, Erode India, March 2019.

[58] R. Patgiri, "HFil: a high accuracy bloom filter," in *Proceedings of the 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2019 HFil: A High Accuracy Bloom Filter; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 2169–2174, Zhangjiajie,China, December 2019.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 2061769, 9 pages
https://doi.org/10.1155/2022/2061769

Research Article

Using Advanced Analytic Techniques to Optimize Cyber-Physical Defensive Plans in Sports Infrastructures and Facilities

Rui Wang

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Rui Wang; wangrui_197910@163.com

Received 5 May 2022; Revised 18 May 2022; Accepted 24 May 2022; Published 11 June 2022

Academic Editor: Konstantinos Demertzis

The technical projects for securing a network of infrastructures and processes are designed, financed, carried out, maintained, and operated within a general infrastructure system, which can gather activities and funds of the private or public sector. The importance of such projects for modern society is enormous, and there is a positive correlation between the size of infrastructure projects and the strength of the national economy. At the same time, it falls within the critical infrastructure sector most of the time. This work, taking into account the massive importance of investments made in the field of sports and the corresponding significance of the design and implementation of robust cybersecurity systems, presents an innovative optimization system for the design, performance, and adaptation of safeguards of technical projects, which require a high level of security standards. A realistic optimization system of low computational complexity is proposed and tested, dividing the problem into a series of subproblems of one-step optimization, which can be solved with great ease and without requirements on computational resources. The great innovation of the proposed system is that the separation is done so that the solution results from the optimal individual solutions of the subproblems without affecting the final result.

1. Introduction

The primary objective of programs to construct sports infrastructure is to be of service to, further advance, and generally better society. These activities include conceiving of, designing, constructing, and operating facilities necessary for contemporary sports culture and organizing and staging the relevant sporting events. These projects are an essential component of a nation's infrastructure and have significant repercussions on social and economic fronts [1]. They require significant capital investment, provide public services, and, in most cases, are considered to fall within the area of responsibility of the public sector. The necessity and feasibility of most of these projects are usually assessed by general methods of determining their economic characteristics (costs and benefits) [2, 3].

Professionals involved in sports infrastructure projects recognize the interdisciplinary nature of their design. In addition to the operational effectiveness of these projects and their impact on public cohesion, health, and security, planners are called upon to consider their beneficial and adverse environmental, social, and economic effects. They must also consider other factors (e.g., institutional, aesthetic, legal, and financial) to determine whether a particular project is safe and successfully implemented. Designers today are staffed with employees specialized in scientific fields, such as engineering, computer science, economics, and law, who, in addition to technical specialization, have a basic understanding of other sciences and the ability to work with other professionals because the implementation of these projects requires a high degree of interdisciplinarity, especially in the field of ensuring the infrastructure grid they cover [4].

In conclusion, the natural, environmental, social, and most importantly the security framework within which the design takes place varies from that of a patchwork of space-time and topographic elements. Therefore, while trying to adopt a broad strategy and comprehensive techniques, planners need to note that any unique athletic project provides a range of features and constraints. This is something that must be taken into consideration. Techniques that have been effectively implemented in the design

process in the past for specific sports projects of varying sorts might serve as a reference for developing similar projects in the future. On the other hand, individuals in charge of the majority of the projects will probably be required to make modifications in response to the changing circumstances and the general security considerations that have to be taken into account.

This work presents an innovative optimization system for designing, performing, and adapting safeguards for technical projects requiring high-security standards. Taking into account the enormous importance of investments made in the field of sports and the corresponding significance of the design and implementation of robust cybersecurity systems, this work presents an innovative optimization system for the design, performance, and adaptation of safeguards. In particular, a cutting-edge and extremely realistic optimization system is proposed to be used to develop, implement, and maintain technical projects that call for a high level of security standards. The problem is broken down into a series of individual one-step optimization subproblems when using an analytical optimization system. These subproblems are much simpler and easier to solve than the original problem.

1.1. Related Literature Review. This section introduces the essential academic searches related to a practical approach to defending a cyber-physical system on technical and policy levels.

Mehrdad et al. [5] reviewed the publications on industrial cyber-physical security. They aimed to tackle the power transmission systems' protection strengths and vulnerabilities in the face of malicious assaults. They stated that to obtain a greater sense of protection for energy systems. When tackling energy network security issues, researchers should take a systematic approach and examine all phases of the holistic resilience cycle. To enhance the cyber-physical integrity of electrical power networks [6, 7], the idea of the Holistic Resilience Cycle was presented. This is a structured method to power system security that is defined by four steps (prevention and planning, detection, mitigation and reaction, and system recovery) as being inextricably linked and comprehensible only in context.

The study of Cai et al. [8] is characterized by attack modeling, security assessment, attack identification, and mitigation. Existing approaches were evaluated to ascertain the true nature of the cyber-physical power system security issue. Based on these technologies' features and evolution tendencies, the limits of the present research were identified, and solutions were proposed to further this field's study [9]. According to their security study, the future focal areas can be stated as follows: the field of theory to the offensive penetration method of systems should be examined in conjunction with the actual communication infrastructure and security prevention mechanisms. With the cyber and physical worlds inextricably linked, the power system cascade failure induced by cyberattacks was investigated, and a fusion analysis and quantitative risk assessment approach has been provided. Finally, attack modeling and defensive detection were accomplished using a cyber-physical fusion model. Then, following attack route prediction, a qualitative distinction of common defects and intense assaults, real-time assessment of protection stability, and digital aid decision-making could be accomplished.

Lai et al. [6] introduced a trilevel optimization model for constructing a coordinated assault scenario and determining the ideal defense strategy, which is novel in resource management to resist an attack. Additionally, considerable reductions in unsaved energy were seen when the suggested optimization technique was used to distribute defense resources. The numerical findings indicate that assault and defense methods vary according to offensive budgets, defense budgets, and restoration periods. Intruders are likely to conduct assaults that result in compounding failures, and the ideal defense approach would prevent such failures. Additionally, the formulation might be enhanced by factoring in the uncertainty associated with the attacker and restoration processes, dynamic difficulties, and grid storage incorporation.

He and Yan [2] conducted a thorough and systematic evaluation of significant smart grid attack risks and security measures. They began their assessment by providing an overview of smart grid security from a cyber-physical viewpoint before focusing on attack strategies that substantially influence the functioning of the power grid and the accompanying response measures. They then examined the potential problems associated with smart grid security after an in-depth examination of the threats and responses. They concentrated on assaults and defenses in the smart grid by conducting a complete and systematic analysis of the state-of-the-art in the sector, including everything from protection frameworks to attack methods and defensive techniques and a variety of possibilities and problems. They believed that their publication would inform people of attack dangers and mitigation techniques in complex cyber-infrastructure facilities such as the smart grid and would motivate researchers to work on developing secure and resilient networks.

Hao et al. [10] developed a strategy for effectively computing an execution plan that optimizes the number of engineered code iterations to achieve maximum protection impact while ensuring the guarded task system's real-time controllability through a novel reaction time analysis. They demonstrated how to incorporate protection mechanisms into practical cases. The suggested approach can determine a suboptimal plan for executing a designed operating security inspection code for shielded tasks or programs to obtain the maximum protection impact while ensuring the system's stimulability. Both simulation-based testing and an application of the suggested approach on a prototype self-driving vehicle demonstrate that the proposed method may be utilized to secure real-time systems.

Hasan et al. [11] provided a technique for prioritizing cyber risk remediation plans in cyber systems that are both efficient and cost-effective (safety implications). These researchers developed a framework for estimating how cyberattacks and random system failures could affect their security and cause catastrophic harm. We undertook an operational impact assessment to determine the magnitude

of the damage caused by CPS threats. They advised constructing a model based on a data-driven attack and fault graph. In the end, they suggested building a strategic response decision capacity that comprises mitigation measures and policies that balance functional robustness and risk. The exploratory study in a real-world testbed showed that allocating resources based on node importance significantly decreased system-level risk. In the future, they were intending to expand the system and include all accessible system-level remediation measures, patch management, and system resilience.

1.2. Modeling the Cyber-Physical Defensive Plan. In modern reality, the complexity of systems and the evolution of technology require integrated defense security planning through an optimal economic approach [12, 13]. Existing experience shows that the anticipated benefits of such design are significant, especially in developing countries, in terms of the quality of the final result, the economy achieved, and the speed of implementation. This is the case even though the degree to which such methodologies are used varies depending on the type of project being undertaken. However, the rigorous application of the systemic concept in manufacturing technical works is a challenging endeavor. This is because it necessitates the detailed characteristics of the system, several technical-economic studies, the application of behavior calculation models, sensitivity analysis, and the formulation of optimal strategies regarding the objectives that have been established. Even though rigorous systematic studies could be beneficial, the abovementioned requirements render their implementation in small-scale projects impractical and prohibitive. However, decision-making is a necessary and ongoing process [6]. The development of systematic analysis techniques with simplified requirements, which most scholars accept and able to improve the effectiveness of promoted measures, is of great practical importance.

It should be noted that the mathematical models of the system in the method of systematic analysis play a central role in the modeling of the systems in question, as it is an essential tool in the modern design and management of projects and strategies [14, 15]. In general, such a model consists of one or more statements, expressed in mathematical terms, that describe relationships between dependent and independent variables, as shown in Figure 1.

The cyber-physical defensive plan is described with a mathematical model. This model is comprised of equations, logical assertions, and other instructions for processing the data that is currently available, as well as for creating and analyzing data that has been artificially generated [16]. The relationships that describe the system, correlating the input and output variables, are expressed by parameters, which are typically required to be determined by observations and measurements of the output variables and which can be constant or variable in a predetermined manner. In general, the parameters must be determined by the observations and measurements of the variables that are output from the system. Exogenous variables are those that the person experimenting does not have any influence over, and probability functions are used to characterize them. On the other hand, variables whose values can be entirely or partially determined are referred to as choice variables. These are the variables that are discussed further below. Restrictions or prohibitions that are applied to the model can include physical, economic, or any number of other factors that mathematical models can express [17, 18].

The analysis of mathematical models includes tables, graphs, mathematical equations, logical statements, and verbal descriptions, which are means of describing system boundaries, system input, and output elements and their relationships, and any feedback between output and input variables to achieve the desired result of the relevant modeling [19]. Specifically, the different types of mathematical models of the system depending on the types of mathematical functions used in this modeling are presented below [20, 21]:

(1) Algebraic equation: It can be obtained by adjusting a curve in empirical measurements; for example,

$$y = f(x) = a_0 + a_1 x + a_2 x^2. \tag{1}$$

(2) Equation of differences: They can describe time-varying systems with delay, memory, multiple variables, and so on; for example,

$$\frac{y_{k+1} = a_k y_k + b_k x_k}{z_{k+1} = \gamma_1 y_{k+1}^{\gamma_2}}. \tag{2}$$

(3) Normal differential equation: It can be obtained from processes of reduction or increase of the examined variable state; for example,

$$\frac{\mathrm{d}y}{\mathrm{d}t} = a y(t) + b x(t), \tag{3}$$

where a, b are the system parameters.

(4) Integral equation (an equation in which an unknown function appears under an integral sign): Known relationships that can be captured in the form of integral; for example,

$$y(t) = \int_{t_0}^{t} g(t, \tau) x(\tau) \mathrm{d}\tau. \tag{4}$$

(5) Differential equation with some derivatives; for example,

$$S \frac{\partial h}{\partial t} + \nabla (T \nabla h) = R - P, \tag{5}$$

where S, t, h, R, P are the system parameters.

This particular equation is a differential equation, which means that it establishes a connection between one or more unknown functions and the derivatives of those functions. The function stands in for the system quantities; the results stand in for the rates of change those quantities are subject to, and the differential equation defines the connection between the two. To put it another way, the status variables

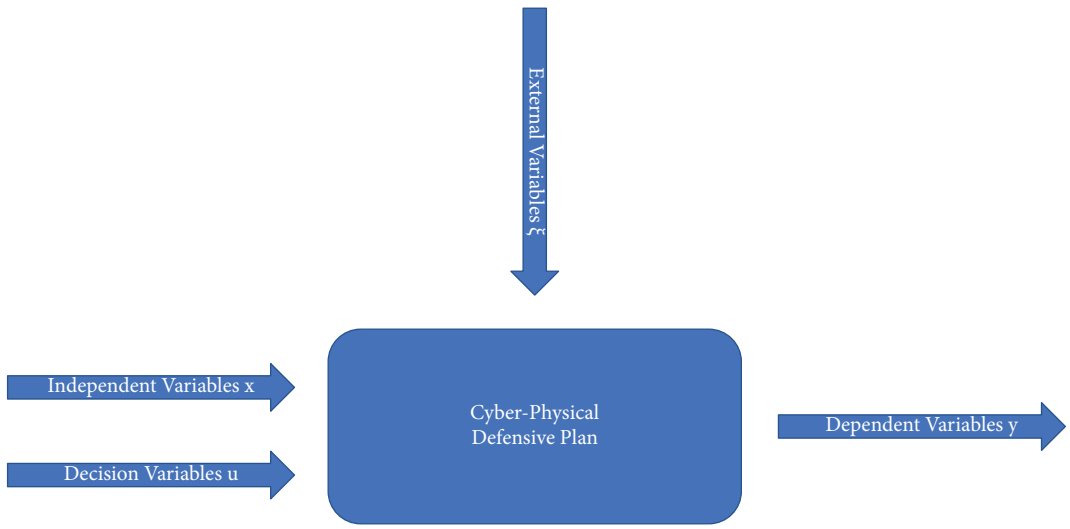


Figure 1: Schematic description of the mathematical model.

represent the bare minimum of variables needed to describe the conditions of the system at any given time or location. Each possible configuration of the decision variables gives rise to a distinct policy or group of decisions. It is possible to implement a method if doing so does not violate any restrictions, and the area of viable approaches is the set of all possible policies taken together.

The objective function is an all-encompassing way of expressing various concepts related to optimality or the most desirable outcome. In a broader sense, the objective function is a performance indicator that we can use to evaluate the implications or derivatives produced by the system. For instance, the goal function can be utilized to calculate the cost of various amounts of resources generated or used in the context of the sports projects that are the topic of this conversation.

Summarizing the above, the methodology proposed for modeling the cyber-physical defensive plan requires [2, 11]:

(1) Model of the system in the general form (as described in Figure 1):

$$\underline{y} = fn\left(\underline{x}, \underline{u}, \underline{\xi}\right), \tag{6}$$

with

$$\begin{aligned} \underline{y} &= \left[y_1, y_2, \ldots y_n\right]^T, \\ \underline{x} &= \left[x_1, x_2, \ldots x_n\right]^T, \\ \underline{u} &= \left[u_1, u_2, \ldots u_n\right]^T, \\ \underline{\xi} &= \left[\xi_1, \xi_2, \ldots \xi_n\right]^T. \end{aligned} \tag{7}$$

(2) Performance indicator (objective function) related to the outcome of a specific policy applied to the problem:

$$\min J = J\left(\underline{y}, \underline{u}\right). \tag{8}$$

(3) Set of restrictions:

$$\frac{F\left(y, \underline{u}\right) = 0}{G\left(y, \underline{u}\right) \geq 0}. \tag{9}$$

It should also be emphasized that the system in question is thought of as a model of distributed parameters. This means that it takes into account the behavioral deviations from point to point throughout the system. This contributes to the system's overall complexity and the very realistic modeling that is followed. In addition, the primary modeling techniques can be broken down into four categories: statistical methods, research simulation using sampling techniques, probabilistic models and techniques, and modeling techniques based on probabilities.

1.3. Analytic Technique to Optimize Cyber-Physical Defensive Plan. After the cyber-physical defensive plan parametric system has been modeled, the general optimization problem is formulated as follows: we want to determine the values of the decision variables u that minimize the objective function J with known J, F, G functions under a set of constraints [22].

Specifically,

$$\begin{aligned} \min J &= J\left(\underline{y}, \underline{u}\right), \\ F\left(\underline{y}, \underline{u}\right) &= 0, G\left(\underline{y}, \underline{u}\right) \geq 0. \end{aligned} \tag{10}$$

Decision theory is divided into two broad categories, based on whether the decision-maker is a single body or multiple bodies. So far, similar problems have been

encountered in the first category of methods, which can be divided into static or single-stage and serial or multistage, where time can be discrete or continuous. The static problem concerns minimizing the cost function, which is a function of the decision variables vector. In the serial problem, the vector of system state variables evolves in time or space according to a method of equations in which decision variables are also involved. The cost function is the sum of the transition costs at each stage and ultimately depends on the (known) initial situation and values of the decision variables at each stage [14, 23]. The solution to the above problem for optimal control is developed using classical optimization methods where the functions are continuous and derivable without restrictions.

The proposed method is an advanced method that separates the multistage optimization problem into a series of single-stage optimization problems [24]. Even multistage systems of increased complexity, such as the one under consideration, can be solved with particular ease. The great innovation of the proposed solution is that the separation is done in such a way that the optimal solution of the initial problem results from the optimal solutions of the individual issues so that the method of solving does not affect the final result.

To be more specific, the following policy holds [25–27]:

$$\pi = \{\mu_0, \ldots, \mu_{N-1}\}, \tag{11}$$

which is a set of functions that determines the values of the decision variables $\underline{u}_k$ from the values of the state variables $\underline{x}_k$; that is,

$$\underline{u}_k = \mu_k(\underline{x}_k). \tag{12}$$

The problem is to minimize costs for all possible policies π so that

$$\min_\pi J_\pi(\underline{X}_0) = J^*(\underline{X}_0). \tag{13}$$

Let S_k be the set of all possible states at time k so that

$$(S_k \subset R^n). \tag{14}$$

And let C_k be the set of all possible decisions at time k so that

$$(C_k \subset R^m). \tag{15}$$

For every

$$\underline{x}_k \in S_k \Longleftrightarrow \underline{u}_k = \mu_k(\underline{X}_k) \in C_k. \tag{16}$$

Then, $U_k(\underline{X}_k)$ is the sum of all possible decisions at time k, if the state is $\underline{x}_k$ (i.e., takes into account the constraints of the problem):

$$U_k(\underline{X}_k) \subset C_k). \tag{17}$$

So, there is an acceptable policy:

$$\pi = \{\mu_0, \ldots, \mu_{N-1}\}, \tag{18}$$

which is a set of functions that have a value field of the set:

$$U_k(\underline{x}_k) or \mu_k\colon S_k \longrightarrow C_k. \tag{19}$$

such that

$$\mu_k(\underline{X}_k) \in U_k(\underline{X}_k) \forall \underline{X}_k \in S_k. \tag{20}$$

So, the intermediate cost depends only on the current situation at time k and takes the form

$$J_k(x_k) = g_k(x_k, u_k) + J_{k+1}(f_k(x_k, u_k)). \tag{21}$$

In this equation, the problem of the existence of higher derivatives of the proposed function exists, where the sum of all possible decisions at time k is a function of the actual decision. An acceptable policy is a self-adjoint operator π, and U is a bounded self-adjoint operator. The proposed approach gives a new direction without multiple operator integrals to improve earlier results. It is a method that uses only unitary operators. This fact is proved as follows:

$$\begin{aligned}
J_N &= g_N(x_N) = J_N(x_N),\\
J_{N-1} &= g_N(x_N) + g_{N-1}(x_{N-1}, u_{N-1}) = g_{N-1}(x_{N-1}, u_{N-1}) + J_N(x_N)\\
&= g_{N-1}(x_{N-1}, \mu_{N-1}(x_{N-1})) + J_N(f_N(x_{N-1}, \mu_{N-1}(x_{N-1})) = J_{N-1}(x_{N-1}),\\
J_{N-2} &= g_N(x_N) + g_{N-1}(x_{N-1}, u_{N-1}) + g_{N-2}(x_{N-2}, u_{N-2}) = g_{N-2}(x_{N-2}, u_{N-2}) + J_{N-1}(x_{N-1})\\
&= g_{N-2}(x_{N-2}, \mu_{N-2}(x_{N-2})) + J_{N-1}(f_{N-1}(x_{N-2}, \mu_{N-2}(x_{N-2}))) = J_{N-2}(x_{N-2}).
\end{aligned} \tag{22}$$

And the problem of minimization at each stage is expressed as follows:

$$\begin{aligned} J_N^*(x_N) &= g_N(x_N), \\ J_k^*(x_k) &= \min_{u_k \in U_k(x_k)} \left[g_k(x_k, u_k) + J_{k+1}^*(f_k(x_k, u_k)) \right], \\ & \qquad k = N-1, N-2, \ldots, 0, \end{aligned} \tag{23}$$

where $J_k^*(X_k) = J^*(X_k)$ is the optimal cost for the problem starting from the state x_k at time k and thus $J_0^*(X_0) = J^*(x_0)$ is the minimum cost of transition from x_0 to x_N:

Consequently, if

$$\exists \pi^* = \{\mu_0^*, \ldots, \mu_{N-1}^*\}, \tag{24}$$

such that

$$\mu_k^*(X_k), \tag{25}$$

achieves the minimum for each X_k; then, μ^* is the optimal policy because

$$J_N^*(x_N) = g_N(x_N), \tag{26}$$

which is calculated from

$$J_k(x_k) = g_k(x_k, u_k) + J_{k+1}^*(x_{k+1}). \tag{27}$$

So, based on the principle of optimality, if $\{\mu_0^*, \ldots, \mu_{N-1}^*\}$ is optimal for the initial problem, then $\{\mu_k^*, \ldots, \mu_{N-1}^*\}$ is optimal for the problem starting at time k.

Finding the most effective approach to solving many important practical problems requires one to investigate many approaches. In many cases, this requires determining either the highest or lowest value that can be returned by a function. The majority of these issues can be resolved by first locating the right function and then applying the principles of calculus in order to ascertain whether the needed maximum or minimum value should be found. In this problem, this means that the best policy will yield the minimum cost of moving to the final state from any intermediate stage k. The proof of this is based on the following two propositions [28]:

$$\min_{x,y} \left[h_1(x) + h_2(x, y) \right] = \min_x \left[h_1(x) + \min_y h_2(x, y) \right]. \tag{28}$$

$$\min_\mu \left[h(x, \mu(x) \right] = \min_\mu \left[h(x, u) \right]. \tag{29}$$

As for $i+1$,

$$\begin{aligned} j_{i+1}^*(x_{i+1}) = j^*(x_{i+1}) = \min_{\{\mu_{i+1}, \ldots, \mu_{N-1}\}} \Bigg[& g_N(x_N) \\ & + \sum_{k=i+1}^{N-1} g_k(x_k, \mu_k(x_k)) \Bigg]. \end{aligned} \tag{30}$$

Then, for i, we have

$$J^*(x_i) = \min_{\{\mu_i, \ldots, \mu_{N-1}\}} \left[g_N(x_N) + \sum_{k=i}^{N-1} g_k(x_k, \mu_k(x_k)) \right]. \tag{31}$$

So,

$$\begin{aligned} &= \min_{\mu_i} g_i(x_i, \mu_i(x_i)) + \min_{\{\mu_{+1}, \ldots, \mu_{k+1}\}} \Bigg[g_N(x_N) \\ &\quad + \sum_{k=i+1}^{N-1} g_k(x_k, \mu_k(x_k)) \Bigg] \\ &= \min_{\mu_i} g_i(x_i, \mu_i(x_i)) + J^*(x_{i+1}) \\ &= \min_{\mu_i} g_i(x_i, \mu_i(x_i)) + J^*(x_{i+1}). \end{aligned} \tag{32}$$

And so,

$$= \min_{u_i \in U_i(x_i)} g_i(x_i, u_i) + J_{i+1}^*(f_i(x_i, u_i)) = J_i^*(x_i). \tag{33}$$

This fact verifies the request and solves the initial optimization problem.

Finally, although the computational load can be huge for a large number of decision variables and many stages, the proposed algorithm $|C| \times |S| \times N$ is much faster than the simple solutions. As shown above, no exhaustive calculation is applied for each k, provided that

$$S_k \longrightarrow C_k |Cl^{|S|}. \tag{34}$$

Since for N stages, the policy $\pi = \{\mu_0, \ldots, \mu_{N-1}\}$ is optimized based on

$$\left\{ |C|^{|S|} \right\}^N = |Cl|^{|S|N}. \tag{35}$$

Lattice models are a method that can be utilized in the valuation of economic derivatives. In a lattice model, the shortest route is searched to make the best conclusion on finding the most inexpensive bid in the search for CCTV equipment for the physical perimeter security of the sports project under consideration. It is necessary to use a discrete-time model to model the potential correlation aspects of complicated issues. The preceding technique is shown using a concrete case described in detail [29–31]. Due to the dependence on multiple paths, the Monte Carlo methods fail to make optimal decisions. The Monte Carlo methods are a comprehensive class of computer algorithms based on the concept of repeatedly taking a sample from a random pool to acquire numerical results. The fundamental idea is to use a chance to find solutions to problems that, in theory, might be solved using deterministic methods. When the issue in question involves a large number of variables that are each constrained uniquely, these methods are computationally inefficient. This indicates that approximating a solution using these methods consumes a significant amount of both time and computational effort. In addition, the model will

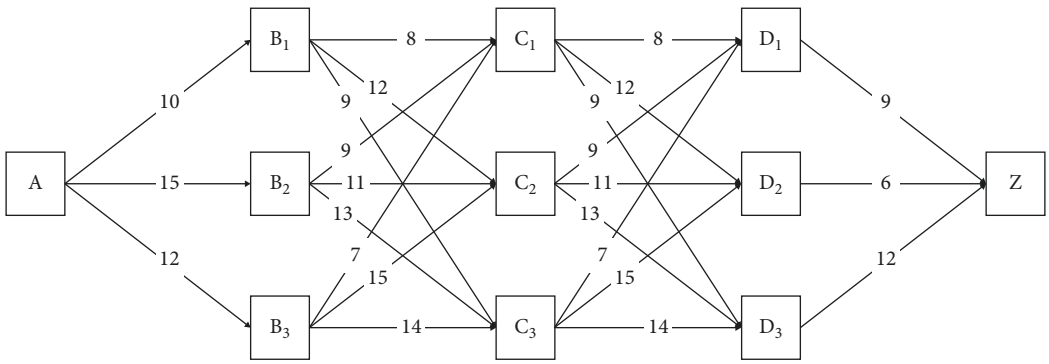


Figure 2: Financial lattice model.

produce unsatisfactory outputs if the parameters and constraints that are fed into it are of low quality. In this example, the solution with the minimum cost is requested within the A–Z, where the transition costs are as shown in Figure 2.

As shown in the figure, no loop has a negative cost. If the connection ij does not exist we set c_{ij} = infinity, while c_{ii} is taken as 0. Also, x_k is taken as the state where the node is in k stage (stage is the transition between nodes, and control is the decision of the next situation). $X_{k+1} = u_k$ is taken as a dynamic equation, and $g_k(X_k, u_k) = c_{X_k} u_k = c_{ij}$ as a cost function.

So based on the proposed algorithm, we have

$$J_N(x_N) = g_N(x_N) = \left\{ \frac{\infty \, if \, x_N = A}{0 \, if \, x_N = Z} \right. . \tag{36}$$

The optimal cost to get to node j starting from i is calculated as follows:

$$J_k(x_k) = \min_{u_k}\left[c_{x_k u_k} + J_{k+1}(u_k)\right] \eta J_k(i) = \min_j\left[c_{ij} + J_{k+1}(j)\right]. \tag{37}$$

Implementing the corresponding table of statements, without using exhaustive calculation, the statements are calculated as follows [32–34]:

$$3^3 = n^{N-1} \text{where } N \geq 3, \tag{38}$$

where N is the number of stages and n is the number of nodes in each intermediate stage.

The solution will occur in 3 stages, wherein each stage a decision will be made for an offer. The benefits are

$$a_j\left(1 - e^{-b_j x_j}\right). \tag{39}$$

The costs are

$$c_j x_j^{d_j}. \tag{40}$$

The state equation is

$$S_{j+1} = S_j - X_j. \tag{41}$$

The objective function is

$$g_j(x_j) = a_j\left(1 - e^{-b_j x_j}\right) - c_j x_j^{d_j}. \tag{42}$$

So,

$$\frac{J_N^*(S_N) = g_N(S_N) = 0}{J_j^*(X_j) = \max_{x_j}\left[g_j(x_j) + J_{j+1}^*(S_j - X_j)\right] j = N-1, N-2, \ldots, 0}. \tag{43}$$

Therefore, the best policy is

$$\frac{A[10] \longrightarrow B_1[8] \longrightarrow C_1[8] \longrightarrow D_1[9] \longrightarrow Zf^* = 35}{A[10] \longrightarrow B_1[9] \longrightarrow C_3[7] \longrightarrow D_1[9] \longrightarrow Zf^* = 35}. \tag{44}$$

2. Conclusions

In this work, we proposed an innovative optimization system for designing, implementing, and updating technical security projects, which require a high level of security standards. It is an analytical way of optimizing that divides the multistage optimization problem into a series of one-step optimization problems. Even multistage systems of increased complexity can be solved with particular ease. The great innovation of the proposed solution is that the separation is done in such a way that the optimal solution of the initial problem results from the optimal solutions of the individual issues so that the method of solving does not affect the result.

A clear example of the proposed procedure was presented descriptively. Specifically, the shortest route was sought in a lattice model to find the best decision on finding the most economical bid in the search for CCTV equipment for the physical perimeter security of the sports project in question. The solution with the minimum cost was achieved based on the proposed approach.

The extensive comparison with probabilistic methodologies that represent parts of stochastic systems through appropriate statistical parameters is an essential aspect that they should expand upon in the next stage of this research project. Additionally, queuing methods and inventory theory may give models for more extensive optimization and perhaps more efficient local decision-making systems.

Data Availability

The data used in this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

References

[1] M. PatéCornell, M. Kuypers, M. Smith, and P. Keller, "Cyber risk management for critical infrastructure: a risk analysis model and three case studies," *Risk Analysis*, vol. 38, no. 2, pp. 226–241, 2018.

[2] H. He and J. Yan, "Cyberphysical attacks and defences in the smart grid: a survey," *IET Cyber-Physical Systems: Theory & Applications*, vol. 1, no. 1, pp. 13–27, 2016.

[3] D. L. Marino, C. S. Wickramasinghe, K. Amarasinghe et al., "Cyber and physical anomaly detection in smart-grids," in *Proceedings of the 2019 Resilience Week (RWS)*, pp. 187–193, San Antonio, TX, USA, November 2019.

[4] V. Borgiani, P. Moratori, J. F. Kazienko, E. R. Tubino, and S. E. Quincozes, "Towards a Distributed Approach for Detection and Mitigation of Denial of Service Attacks within Industrial Internet of Things," *IEEE Internet Things Journal*, vol. 8, no. 6, pp. 4569–4578, 2021.

[5] S. Mehrdad, S. Mousavian, G. Madraki, and Y. Dvorkin, "Cyber-physical resilience of electrical power systems against malicious attacks: a review," *Current Sustainable/Renewable Energy Reports*, vol. 5, no. 1, pp. 14–22, 2018.

[6] K. Lai, M. Illindala, and K. Subramaniam, "A tri-level optimization model to mitigate coordinated attacks on electric power systems in a cyber-physical environment," *Applied Energy*, vol. 235, pp. 204–218, 2019.

[7] P. Akubathini, S. Chouksey, and H. S. Satheesh, "Evaluation of Machine Learning approaches for resource constrained IIoT devices," in *Proceedings of the 2021 13th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pp. 74–79, Chiang Mai, Thailand, October 2021.

[8] X. Cai, Q. Wang, Y. Tang, and L. Zhu, "Review of cyber-attacks and defense research on cyber physical power system," in *Proceedings of the2019 IEEE Sustainable Power and Energy Conference (iSPEC)*, pp. 487–492, Beijing, China, November 2019.

[9] P. Boström-Rost, "On Informative Path Planning for Tracking and Surveillance," 2019, http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-157026.

[10] X. Hao, M. Lv, J. Zheng, Z. Zhang, and W. Yi, "Integrating cyber-attack defense techniques into real-time cyber-physical systems," in *Proceedings of the 2019 IEEE 37th International Conference on Computer Design (ICCD)*, pp. 237–245, Abu Dhabi, UAE, November 2019.

[11] K. Hasan, S. Shetty, A. Hassanzadeh, and S. Ullah, "Towards Optimal Cyber Defense Remediation in Cyber Physical Systems by Balancing Operational Resilience and Strategic Risk," in *Proceedings of the MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM)*, pp. 1–8, Norfolk, VA, USA, November 2019.

[12] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.

[13] M. T. Amron, R. Ibrahim, and S. Chuprat, "A review on cloud computing acceptance factors," *Procedia Computer Science*, vol. 124, pp. 639–646, 2017.

[14] S. Roy, S. U. Kadir, Y. Vorobeychik, and A. Laszka, "Strategic remote attestation: testbed for internet-of-things devices and stackelberg security game for optimal strategies," in *Decision and Game Theory for Security*, pp. 271–290, Springer, Berlin, Germany, 2021.

[15] S. Varshney, D. Munjal, O. Bhattacharya, S. Saboo, and N. Aggarwal, "Big data privacy breach prevention strategies," in *Proceedings of the 2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)*, pp. 1–6, Gunupur Odisha, India, December 2020.

[16] R. Cavazos-Cadena and R. Montes-de-Oca, "The value iteration algorithm in risk-sensitive average markov decision chains with finite state space," *Mathematics of Operations Research*, vol. 28, no. 4, pp. 752–776, 2003.

[17] J. Qian, J. P. Lu, S. L. Hui, Y. J. Ma, and D. Y. Li, "Dynamic analysis and CFD numerical simulation on backpressure filling system," *Mathematical Problems in Engineering*, vol. 2015, Article ID e160641, 8 pages, 2015.

[18] B. Bordel, R. Alcarria, and T. Robles, "Recognizing human activities in Industry 4.0 scenarios through an analysis-modeling- recognition algorithm and context labels," *Integrated Computer-Aided Engineering*, vol. 29, no. 1, pp. 83–103, 2021.

[19] M. P. Deisenroth, T. Ohtsuka, F. Weissel, D. Brunn, and U. D. Hanebeck, "Finite-horizon optimal state-feedback control of nonlinear stochastic systems based on a minimum principle," in *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 371–376, Heidelberg, Germany, September 2006.

[20] K. Jin, T. Yin, C. A. Kamhoua, and M. Liu, "Network Games with Strategic Machine Learning," in *Lecture Notes in Computer Science, Decision and Game Theory for Security*, pp. 118–137, Springer, Berlin, Germany, 2021.

[21] C. Sangüesa, "Error bounds in approximations of random sums using gamma-type operators," *Insurance: Mathematics and Economics*, vol. 42, no. 2, pp. 484–491, 2008.

[22] O. Sundström, L. Guzzella, and P. Soltic, "Optimal hybridization in two parallel hybrid electric vehicles using dynamic programming," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 4642–4647, 2008.

[23] A. R. Butler, T. H. Nguyen, and A. Sinha, "Countering attacker data manipulation in security games," in *Decision and Game Theory for Security*, pp. 59–79, Springer, Berlin, Germany, 2021.

[24] X. Wang, Y. Ji, J. Wang, Y. Wang, and L. Qi, "Optimal energy management of microgrid based on multi-parameter dynamic programming," *International Journal of Distributed Sensor Networks*, vol. 16, no. 6, Article ID 155014772093714, 2020.

[25] A. R. Horowitz, "Loss functions and public policy," *Journal of Macroeconomics*, vol. 9, no. 4, pp. 489–504, 1987.

[26] C. Segovia and K. Smith-Miles, "Integrating Game Theory and Data Mining for Dynamic Distribution of Police to Combat Crime," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 780–783, Santiago, Chile, September 2018.

[27] M. Bellare and O. Goldreich, *On probabilistic versus deterministic provers in the definition of proofs of knowledge Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pp. 114–123, Springer, Berlin, Germany, 2011.

[28] S. Gupta, S. Al-Obaidi, and L. Ferrara, "Meta-analysis and machine learning models to optimize the efficiency of self-healing capacity of cementitious," *Material," Materials*, vol. 14, no. 16, 2021.

[29] A. Grabowski and T. Pfau, "A lattice of magneto-optical and magnetic traps for cold atoms," in *Proceedings of the 2003 European Quantum Electronics Conference. EQEC 2003 (IEEE Cat No.03TH8665)*, p. 274, June 2003.

[30] T. Reisz, "A power counting theorem for Feynman integrals on the lattice," *Communications in Mathematical Physics*, vol. 116, no. 1, pp. 81–126, 1988.

[31] Y. Zhang, Y. Wang, and J. Yang, "Lattice LSTM for Chinese sentence representation," *IEEE Transactions on Audio Speech and Language Processing*, vol. 28, pp. 1506–1519, 2020.

[32] J. Xie, X. Bai, D. Feng, and D. Gan, "Peaking cost compensation in northwest China power system," *European Transactions on Electrical Power*, vol. 19, no. 7, pp. 1016–1032, 2009.

[33] Y. Azan Basallo, V. Estrada Senti, and N. Martinez Sanchez, "Artificial intelligence techniques for information security risk assessment," *IEEE Latin America Transactions*, vol. 16, no. 3, pp. 897–901, 2018.

[34] X. Liang, T. Qi, Z. Jin, S. Qin, and P. Chen, "Risk assessment system based on fuzzy composite evaluation and a back-propagation neural network for a shield tunnel crossing under a river," *Advances in Civil Engineering*, vol. 2020, Article ID 8840200, 14 pages, 2020.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 5027256, 7 pages
https://doi.org/10.1155/2022/5027256

Research Article

Privacy Leaks Protection in Music Streaming Services Using an Intelligent Permissions Management System

Qian Wang

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Qian Wang; wangqian197902@126.com

Received 27 April 2022; Revised 16 May 2022; Accepted 24 May 2022; Published 8 June 2022

Academic Editor: Konstantinos Demertzis

A security violation is referred to as a personal data breach when it leads to unintentional and unlawful destruction, loss, alteration, unauthorized disclosure, or access to personal data that has been communicated, stored, or otherwise processed in some other manner. Based on the principles of information security, we can define a breach of confidentiality as the unauthorized or accidental disclosure or access to personal data, a breach of integrity as the unauthorized or accidental alteration of personal data, and a breach of availability as the unauthorized or accidental loss of access to or destruction of personal data. This paper suggests designing an intelligent consensus policy management system based on the Markov chain approach. It is a novel system that would analyze the present status of the consensus elements for future development and anticipates the possibility of possible breaches of sensitive personal data. The evaluation of the proposed strategy is based on a policy scenario that involves a hypothetical consensus and a data breach of sensitive information to music streaming services.

1. Introduction

Online streaming replaces the purchase of Compact Disc-CDs, and podcasts come and take the place of radio. In recent years, more and more people are changing their habits in the music industry because we are in the middle of rapid development [1]. The ability to listen to music directly on the mobile via the Internet, wherever we are, surpasses in ease any other method. However, the sound and listening quality are not the same as traditional. For this very reason, streaming services were created from which we can legally listen to music [2]. Most of them are designed to help the user discover new music, with the software itself suggesting what new we will listen to based on our musical preferences. Of course, a prerequisite for these services is payment. After a short trial period that requires a credit card, a small monthly fee gives access to many songs.

However, the streaming era brings new issues, the most basic of which is the acquisition of sensitive personal data and its "processing" via automated means. Personal data are information about a living person who can be identified (e.g., name, home address, e-mail address, location data, and usage data) [3]. If put together, different information can identify a particular person as personal data. The term "processing," by automated means as well as nonautomated processing includes the following activities: data collection and registration; organization and structure; storage; adaptation and modification; retrieval; retrieval of information; use; disclosure by transmission, dissemination, or any other form of disposal; correlation or combination; and restriction; deletion or destruction of personal data [4].

To be valid, personal data processing consent must be provided clearly and concisely, in language that is easy to understand and different from other information, such as terms and conditions. In addition, consent must be freely given, specific, aware, and unequivocal to be valid [5]. The request must state the purpose for which personal data will be used [6]. In certain circumstances, data subjects have the right to refuse to be subjected to a decision exclusively based on automated processing. Although this norm is generally

followed, there are a few exceptions, such as when the data subject has explicitly consented to an automated decision-making mechanism. A data breach occurs when personal data are disclosed, accidently or illegally, to unauthorized recipients, made temporarily unavailable, or altered [7, 8].

So, it is essential to have an intelligent mechanism that can manage the consent granted to have a case for creating a profile, which can regulate the use, and therefore, there is a case of data leakage. Based on the adoption of the Markov chain methodology [9], it is proposed to implement an intelligent consent management system in Music Streaming Services, which considers the current state of consent for the future development and forecast possible leaks of sensitive personal data [3, 10].

2. Related Literature

This section explores relevant work on Markov chains implementation, privacy concerns, risk perception, and user behavior in response to security recommendations [11, 12]. Wiering et al. [13] investigated multiobjective Markov decision systems in 2007 by substituting a cross benefit vector for the conventional linear reward signal. This multi-objective Markov procedure may be transferred when the weighting factor for the various reward elements is known in advance. They anticipated that the weighting function might be arbitrarily chosen and given by the actor or user after the algorithm addressed the issue. They maintained track of Pareto's optimum stationary policies to cope with it.

Feinberg and Rothblum [14] investigated a Markov selection procedure with a different reward system and a specified beginning state dispersion. Suppose the habitation measure of a stationary policy can be described as a convex sum of the habitation evaluations of other stationary policies. In that case, the policy may be divided into states. There are requisite circumstances for dividing a stationary policy in a single state and adequate criteria for splitting it throughout the entire state area. The findings were used to limited issues to compute an optimum policy by calculating and dividing an ideal stationary policy.

O'Connor et al. [15] investigated the optimum transport issue for couples of stationary constraint Markov chains, focusing on calculating ideal transitional connections, a limited family of transfer plans that encapsulate the characteristics of Markov chains. The optimum changeover connectivity issue is solved by aligning the two chains to minimize the overall long-term price. They produced a stable conclusion for both normalized and nonregularized methods and, consequently, a probabilistic coherence result. They tested their theoretical predictions through a mock trial, indicating that the approximation technique has a shorter total runtime and a low error rate. Finally, they expanded their approaches to hidden Markov structures and demonstrated the suggested algorithms' practical use via the implementation of computer-generated music.

Concerning privacy leaks, Yixin et al. [7] conducted semistructured surveys with customers to ascertain their perceptions of the dangers associated with data breaches, their willingness to take preventive actions, and their motivations for inactivity. They discovered that users' mind maps of credit agencies were inadequate and erroneous to a certain extent. They discovered that this conduct is motivated by the expenditures of preventive methods, a positivity bias in evaluating one's chance of persecution, sources of guidance, and a general inclination to wait for response until damage occurs. They reviewed the legal, technological, and pedagogical ramifications and possible approaches for improving consumer protection in the credit reporting system. Finally, they suggest future research options.

Gwebu et al. [8] investigated the relative usefulness of corporate credibility and post-breach reaction techniques in light of the considerable monetary losses related to data intrusions. The findings suggested that a firm's brand is a critical asset for preserving the firm's worth. Nevertheless, only particular reaction tactics are shown to lessen the fiscal effect on low-reputation organizations. At the same time, reply techniques are less critical for high firms. These results provide operators with proof counsel for preserving business assets after privacy violations and emphasize the demand for establishing more sophisticated breach management techniques. The generated theoretical justifications provided a cognitive foundation for evaluating the effectiveness of different data breach response tactics.

3. Methodology

The proposed methodology concerns implementing a policy consensus rights management system executed for each policy with a stationary Markov policy [9]. The Markov policy is considered to be a Markov chain $\{Xn(R)\}n \in \mathbb{N}0$ which is nondegradable or, more generally, has a unique closed communication class (meaning that transient states outside it are allowed), which is (genuinely) finite repetitive [16]. The idea of modeling is that Markov processes are appropriate stochastic models for describing and studying stochastic systems; the future evolution of which depends solely on their present state each time and not on their specific history. An example of a Markov chain and its mathematical modeling in predicting future situations is shown in Figure 1 [17–19].

Based on the adoption of the methodology of the Markov chains, it is proposed to implement an intelligent consensus policy management system, which considers the current state of the consensus data for the future development and forecasting of possible leaks of sensitive personal data [10]. A stochastic or Markov process [20–22] (or evolution) from T to S is a collection of random variables $\{X(t,\omega)\}_{t\in T,\omega\in\Omega}$ defined in a probability space (Ω, F, P), where $T = \mathbb{N}_0 = \{0, 1, 2, \ldots\}$ the time horizon (usually T is the set of times). S is the state space of the process, i.e., the value field of (t, ω) for each $t \in T$ and $\omega \in \Omega$. Because S is countable or finite, we are talking about a stochastic chain [23]. Ω is the sample space, i.e., the set of all possible results of the luck experiment under study. F: σ-algebra of contingencies is the field of definition of the contingencies of Ω (there are cases of sample space of Ω where we cannot consider each of its

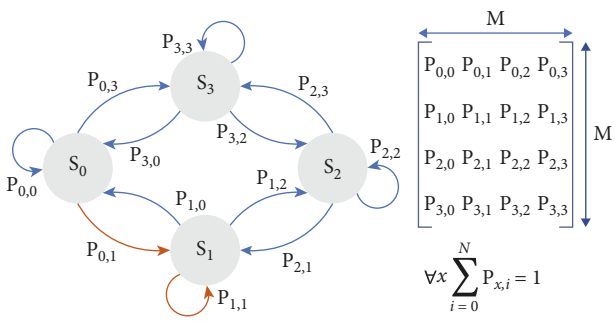


Figure 1: Markov chain.

subsets as a possibility) and P is the probability measure, i.e., a function P: $F \longrightarrow \mathbb{R}$ such that [21, 22]

(1) $P(\Omega) = 1$

(2) $P(A) \geq 0, \forall A \in \mathscr{F}$

(3) $P(U_{V=1}^{\infty} A_v) = \sum_{v=1}^{\infty} P(A_v), \forall \{A_v\}_{v\in\mathbb{N}}$ which $(A_i \cap A_j = \varnothing \forall i \neq j)$

The proposed process focuses on a system of Markov chains, i.e., processes from T = $\mathbb{N}$ 0 in a countable (or simply finite) space S, which have the Markov property [24–26]

$$P(X_{n+1} = j|X_0, X_1, \ldots, X_n) = P(X_{n+1} = j|X_n), \quad n \in \mathbb{N}_0, j \in S. \tag{1}$$

The above relation states that given the value of the random variable X_n (present), the random variable X_{n+1} (future) is stochastically independent of the variables $X_0, X_1, \ldots, X_{n-1}$ (past state of the process) [27]. The explanation of the Markov property of the proposed system is that the future of the Markov chain depends on the past only through the present [28]. So, the probability [22, 24]

$$p_{ij}(n, n+1) := P(X_{n+1} = j|X_n = i), \tag{2}$$

is the probability of passing 1st order from state i to state j by the $(n+1)$-th step. These probabilities do not depend on the time step n (stationary), i.e., [29, 30]

$$p_{ij} = P(X_{n+1} = j|X_n = i) = P(X_1 = j|X_0 = i), \tag{3}$$

so we have homogeneous chains $\{Xn\}$.

Respectively, considering the 1st order transition probability table of the chain [31–33],

$$\mathbb{P} = \{p_{ij}\}_{i,j\in S}, \tag{4}$$

we have a stochastic table, so it is easy to find the chain in the states $i_0, i_1, \ldots, i_{n-1}, i_n \in S$ in succession; the probability is as follows:

$$P(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}, X_n = i_n) = \pi_0(j) \cdot p_{i_0 i_1} \cdots p_{i_{n-1} i_n}. \tag{5}$$

Thus, for the distribution πn of the state of the chain, it is valid that [28, 32]

$$\pi_n(j) = P(X_n = j) \forall j \in S. \tag{6}$$

So,

$$\pi_{n+1}(j) = \sum_{i\in S} P(X_n = i) P(X_{n+1} = j|X_n = i) = \sum_{i\in S} \pi_n(i) \cdot p_{ij}, \tag{7}$$

or equivalents

$$\pi_{n+1} = \pi_n \mathbb{P}, \quad \forall n \in \mathbb{N}_0, \tag{8}$$

and so

$$\pi_n = \pi_0 \mathbb{P}^n, \tag{9}$$

where

$$\mathbb{P}^n = \{p_{ij}^{(n)}\}_{i,j\in S}, \tag{10}$$

the n-th order transition table (or n steps), i.e.,

$$p_{ij}^{(n)} = P(X_n = j|X_0 = i). \tag{11}$$

If we choose $\pi 0 = \pi$ such that $\pi = \pi\, \mathbb{P}$, we obtain $\pi n = \pi\ \forall\ n \in \mathbb{N}$, i.e., the state distribution of the chain is stationary (independent of n). Then, the chain has an unchanged or stationary distribution, and it is in statistical equilibrium. In queuing theory, systems are said to be in "statistical equilibrium" when the number of customers or objects waiting in the line oscillates so that the mean and distribution remain the same over a prolonged period.

Therefore, to find the stationary distribution π, it is enough to solve the system [21, 28, 34, 35]:

$$\begin{cases} \sum_{i\in S}^{\pi=\pi\cdot P} \pi(i) = 1, \\ \pi(i) \geq 0, \forall i \in S. \end{cases} \tag{12}$$

A chain will be nondegradable if the state space S is an entire closed class, i.e.,

$$C(X_n = i, i \in C). \tag{13}$$

The basic premise of the methodology is that the Markov chain

$$\{X_n\}_{n\in\mathbb{N}_0}, \tag{14}$$

is nondegradable, genuinely repetitive, and aperiodic. Then, regardless of its initial distribution is valid,

$$\lim_{n\longrightarrow\infty} \pi_n = \pi, \tag{15}$$

and

$$\lim_{n\longrightarrow\infty} p_{ij}^{(n)} = \pi(j), \quad \forall i \in S, \tag{16}$$

the percentage of time spent by the chain in each $i \in S$ situation is $\pi(i)$, so according to the employer theorem, we have

$$P\left[\frac{V_n(i)}{n} \longrightarrow \pi(i), \quad \forall i \in S\right] = 1. \tag{17}$$

So, the transition from one state to another implies a fee or some cost (negative fee) of the form as follows:

$$R_n = R_n(X_{n-1}, X_n), \quad n \in \mathbb{N}, \tag{18}$$

for the n-th step, so the total reward in the first n steps is

$$C(n) = \sum_{s=1}^{n} R_s(X_{s-1}, X_s), \quad n \in \mathbb{N}. \tag{19}$$

The average fee for going $i \longrightarrow j$ is

$$r_{ij} = E[R_n(i,j)] < \infty, \quad i, j \in S. \tag{20}$$

Ultimately, it applies to the pay rate (average pay per step) as follows:

$$\lim_{n \longrightarrow \infty} \frac{C(n)}{n} = \sum_j r_j \cdot \pi_j. \tag{21}$$

A fascinating question in the modeling of the proposed methodology concerns how we will choose the stationary policy or equivalent by what criteria this choice will be made. Since $C(i, \alpha)$ expresses sensitive data, we will deal with the standard of minimizing their propagation rate, which proves to be the most suitable in many applications.

Of course, if $C(i, \alpha)$ represents nonconsent of rights, the criterion will be maximizing the consent rate, which is equivalent to the previous one [36]. A stationary policy is $X_0 = i, i \in \mathrm{S}$. Also $p_{ij}^{(k)}(R)$ are the probabilities of passing k-th class [37]. Then, the average (or expected) total amount of consonants in the first n time points (or steps) when $X_0 = i$ and R is applied [18, 32, 38]:

$$V_n(i, R) = E_R\left[\sum_{k=0}^{n-1} C(X_k, a_k) | X_0 = i\right], \tag{22}$$

or by the definition and linearity of the (bound) average value:

$$V_n(i, R) = \sum_{k=0}^{n-1} \sum_{j \in S} p_{ij}^{(k)}(R) \cdot C(j, R_j), \tag{23}$$

or else

$$V_n(i, R) = E_R\left[\sum_{k=0}^{n-1} \lambda^k C(X_k, a_k) | X_0 = i\right]. \tag{24}$$

Our objective is to study the behavior of the rate of the average rate of consensus in the long run, i.e., the limit as follows:

$$\lim_{n \longrightarrow \infty} \frac{V_n(i, R)}{n}, \tag{25}$$

for each $i \in \mathrm{S}$.

However, because the Markov chain has a unique closed communication class (positively repetitive), the limit is unique and independent of the initial state i [39]:

$$\begin{aligned} g(R) &= \lim_{n \longrightarrow \infty} \frac{v_n(i, R)}{n} \\ &= \sum_{j \in S} \pi_j(R) \cdot C(j, R_j), \end{aligned} \tag{26}$$

with $\pi_j(R)$, $j \in S$ is the stationary distribution of the chain below the policy R and thus the rate of change of average rate of consent or, in the long run, average rate per unit time. Therefore, a stationary policy $R*$ will be optimal for the average consent rate if each stationary policy R applies [40]

$$g(R^*) \leq g(R). \tag{27}$$

So, since the time horizon is unlimited and the space of situations is finite, it turns out that there will always be a stationary policy that will be optimal in terms of the above criterion. Considering this and using the proposed methodology for each possible stationary policy, we can calculate the stationary distribution and the corresponding rate in each case. The above result is significant because it secures the right to seek an optimal approach for the stationary.

4. Privacy Leaks Protection Scenario

To protect individuals' privacy and help restore trust and transparency in the activities between people and entities that process their data, streaming services provide consent. The data subject's permission is any freely given, precise, informed, and unequivocal expression of the data subject's desires by which he, by a statement or by an obvious action, accepts the processing of personal data about them. Consents are divided into "Free," "Specific," "Informed," and "Indisputably indicated." In the scenario tested by the proposed system, consents are implemented to process users' data. The policy concerns consent if at least all four of the following conditions apply [6, 41, 42]:

(1) Free means that the data subjects are selected and controlled. It is invalid consent if the data subject does not have a natural option, is obliged to consent, or will suffer an undesirable consequence if they do not consent. Unless the consent is accompanied by a nonnegotiable provision of the terms and conditions, it has been begrudged.

(2) Specifically, it aims to ensure user control and transparency for the data subject.

(3) Informed aims to provide information to the persons to whom the data refer before their consent and is necessary to be able to make informed decisions, to understand what they agree on, and for example, to exercise their right to withdraw their consent.

(4) Indisputably indicates that there should be no doubt that the data subject has agreed to the data processing [29, 43].

With the policy improvement method starting from any R policy, we check if it is optimal or not. If not, we find a policy R' with $g(R') \leq g(R)$ and check if it is optimal. Keeping in mind that the number of stationary policies is limited (on account of the fact that both the situation space and the decision space are limited), we can proceed with the procedure described above until we find the most effective policy.

The benefit of using this method is that rather than controlling all of the available stable policies, we only have to maintain a typically small fraction of them and progress from one strategy to another policy improvement.

The equation makes the beginning [44–46]:

$$g(R) = \lim_{n \longrightarrow \infty} \frac{V_n(i,R)}{n}, \quad \forall i \in S, \tag{28}$$

which implies that asymptotically (i.e., for $n \longrightarrow \infty$: large n) holds

$$V_n(i,R) \cong ng(R) + u_i(R), \quad \forall i \in S. \tag{29}$$

The quantities $u_i(R)$ are relative values of the states i when a stationary policy R is applied, and their difference is equal to

$$u_i(R) - u_j(R) \cong V_n(i,R) - V_n(j,R), \quad \forall i, j \in S. \tag{30}$$

The relative values express the transient effect of the initial states on the expected total rate under the application of policy R.

Then, the quantity $h\ u_i(R) - u_j(R)$ expresses the difference in the average total rate, if the process starts from the state i compared with whether it began to from state j, when the policy R.

Equivalently applied, this difference is essentially the maximum rate of consensus so that the system (the chain) starts from state j rather than i below state policy R.

If we assume an aperiodic chain $Xn(R)$, then the limit exists [9, 14, 21]:

$$\lim_{n \longrightarrow \infty} p_{ij}^{(k)}(R) = \pi(j). \tag{31}$$

So, there is also

$$\lim_{n \longrightarrow \infty} V_n(i,R) = \lim_{n \longrightarrow \infty} \sum_{k=0}^{n-1} \sum_{j \in S} p_{ij}^{(k)}(R) \cdot C\left(j, R_j\right). \tag{32}$$

So,

$$u_i(R) - u_j(R) = \lim_{n \longrightarrow \infty} \left[V_n(i,R) - V_n(j,R)\right], \tag{33}$$

which expresses the long-term difference in the mean total rate if the process starts from state i rather than state j, under policy R.

Therefore [29],

$$V_n(i,R) = C\left(i, R_i\right) + \sum_{j \in S} p_{ij}\left(R_i\right) \cdot V_{n-1}(j,R), \tag{34}$$

and so

$$u_i(R) + g(R) \cong C\left(i, R_i\right) + \sum_{j \in S} p_{ij}\left(R_i\right) \cdot u_j(R), \tag{35}$$

or, respectively,

$$u_i(R) \cong C\left(i, R_i\right) - g(R) + \sum_{j \in S} p_{ij}\left(R_i\right) \cdot u_j(R). \tag{36}$$

So, for finding a rhythm and relative values based on the genuine iterative state $r \in S$, it holds [47]:

$$T_i(R) = 1 + \sum_{j \neq r} p_{ij}\left(R_i\right) \cdot T_j(R), \tag{37}$$

where $T_i(R)$ is the expected time of the first visit to r since the chain started from state i under policy R.

For the average rate $K_i(R)$, the following applies:

$$K_i(R) = C\left(i, R_i\right) + \sum_{j \neq r} p_{ij}\left(R_i\right) \cdot K_j(R). \tag{38}$$

Combining the above two equations, we take

$$\begin{aligned} K_i(R) - g(R) \cdot T_i(R) &= C\left(i, R_i\right) - g(R) + \sum_{j \neq r} p_{ij}\left(R_i\right)\left[K_j(R) - g(R) \cdot T_j(R)\right], \\ u_i(R) &= C\left(i, R_i\right) - g(R) + \sum_{j \neq r} p_{ij}\left(R_i\right)\left[K_j(R) - g(R) \cdot T_j(R)\right] + p_{ir}\left(R_i\right) \cdot u_r, \\ u_i(R) &= C\left(i, R_i\right) - g(R) + \sum_{j \neq r} p_{ij}\left(R_i\right) \cdot \left[K_j(R) - g(R) \cdot T_j(R)\right] + \\ &\quad + p_{ir}\left(R_i\right) \cdot \left[K_r(R) - g(R) \cdot T_r(R)\right], u_r = 0. \end{aligned} \tag{39}$$

So, it finally applies that

$$u_i(R) = C\left(i, R_i\right) - g(R) + \sum_{j \in S} p_{ij}\left(R_i\right) \cdot \left[K_j(R) - g(R) \cdot T_j(R)\right], \tag{40}$$

or else

$$u_i(R) = C\left(i, R_i\right) - g(R) + \sum_{j \in S} p_{ij}\left(R_i\right) \cdot u_j, \quad i \in S, \tag{41}$$

namely,

$$\left(g, \underline{u_i}\right) = \left(g(R), \underline{u_i(R)}\right), \tag{42}$$

are a solution of the original system, and therefore the request was proved.

5. Conclusions

Inside the scope of this study, we suggested a novel policy-based consensus management system intending to prevent data breaches within music streaming services. The methodology that has been proposed is solely founded on an advanced Markov chain system. This creates an intelligent consensus policy management framework that considers the current state of the consensus data to predict potential leaks of sensitive personal data and prepare for their development

in the future. To be more specific, we employ Markov processes with a discrete (limited or countable) state space and a distinct parametric space. We examine this way of improvement to see whether or not a policy is optimum. It is a forward-thinking and clever system that can model challenging scenarios, making it virtually more straightforward to discover answers to questions regarding dynamic circumstances of ambiguity.

The provision of consent by streaming services helps to reestablish confidence and transparency in the interactions between individuals and the organizations responsible for processing their data. This safeguards the privacy of individuals. The data subject's permission is any freely given, precise, informed, and unambiguous expression of the data subject's desires by which he, by a statement or by an obvious action, accepts the processing of personal data about them. This expression of the data subject's desires can take the form of a statement or an apparent effort. The terms "Free," "Specific," "Informed," and "Indisputably indicated" are used to classify different types of consent. In the hypothetical situation examined by the suggested system, permissions are successfully applied to handle users' data successfully.

The proposed tactic, in addition to the apparent advantages, lags because of the increased complexity even for a small space of solutions S. For example, for a set of N with two possible decisions (the same for each situation), from the simple multiplication principle, we have $2N$ different stationary policies. For $N = 10$, we have a total of 1024 stationary policies. We can overcome this obstacle with optimization methods and selecting a predefined solution space. Therefore, significant future development of the proposed system is the investigation of optimization methods using biologically inspired methods to find the optimal solution spaces that could significantly simplify the proposed methodology.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] C. Egbert, F. Alhenaki, and D. Johnson, "Leveraging a music streaming platform in establishing a novel storage covert channel," in *Proceedings of the 2020 IEEE 45th Conference on Local Computer Networks (LCN)*, pp. 437–440, IEEE, Sydney, Australia, November 2020.

[2] Z.-Y. Zhao, C.-D. Wang, P.-J. Zheng, Q. Gong, K.-W. Huang, and J.-H. Lai, "Music sharing platform based on sina app engine," in *Proceedings of the 2015 9th International Conference on Frontier of Computer Science and Technology*, pp. 298–303, IEEE, Dalian, China, December 2015.

[3] M. Liu, X. Liu, A. Yan, X. Li, G. Xie, and X. Tang, "An explanatory strategy for reducing the risk of privacy leaks," *Journal of Information Hiding and Privacy Protection*, vol. 3, no. 4, pp. 181–192, 2021.

[4] G. D'Acquisto, J. Domingo-Ferrer, P. Kikiras, V. Torra, Y.-A. de Montjoye, and A. Bourka, "Privacy by design in big data: an overview of privacy enhancing technologies in the era of big data analytics," 2015. Avaliable at: https://arxiv.org/abs/1512.06000.

[5] S. Sanjay, *Data Privacy and GDPR Handbook*, Wiley, Hoboken, NJ, USA, 2022, https://www.wiley.com/en-us/Data+Privacy+and+GDPR+Handbook-p-9781119594192.

[6] B. Mehta, U. P. Rao, R. Gupta, and M. Conti, "Towards privacy preserving unstructured big data publishing," *Journal of Intelligent and Fuzzy Systems*, vol. 36, no. 4, pp. 3471–3482, 2019.

[7] Y. Zou, A. H. Mhaidli, A. McCall, and F. Schaub, "I've got nothing to lose': consumers' risk perceptions and protective actions after the equifax data breach," pp. 197–216, 2018, Accessed: Apr. 21, 2022. [Online]. Available: https://www.usenix.org/conference/soups2018/presentation/zou.

[8] K. L. Gwebu, J. Wang, and L. Wang, "The role of corporate reputation and crisis response strategies in data breach management," *Journal of Management Information Systems*, vol. 35, no. 2, pp. 683–714, 2018.

[9] F. J. Beutler and K. W. Ross, "Optimal policies for controlled Markov chains with a constraint," *Journal of Mathematical Analysis and Applications*, vol. 112, no. 1, pp. 236–252, 1985.

[10] X. Shu, J. Zhang, D. Daphne Yao, and W.-C. Feng, "Fast detection of transformed data leaks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 3, pp. 528–542, 2016.

[11] C. C. Aggarwal, "Neighborhood-based collaborative filtering," in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed., Springer International Publishing, Berlin, Germany, pp. 29–70, 2016.

[12] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., Springer, Berlin, Germany, pp. 291–324, 2007.

[13] M. A. Wiering and E. D. de Jong, "Computing optimal stationary policies for multi-objective Markov decision processes," in *Proceedings of the 2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*, pp. 158–165, Honolulu, HI, USA, April 2007.

[14] E. A. Feinberg and U. G. Rothblum, "Splitting randomized stationary policies in total-reward Markov decision processes," *Mathematics of Operations Research*, vol. 37, no. 1, pp. 129–153, 2012, Feb. 2012.

[15] K. O'Connor, K. McGoff, and A. B. Nobel, "Optimal transport for stationary Markov chains via policy iteration," *Journal of Machine Learning Research*, vol. 23, no. 45, pp. 1–52, 2022.

[16] O. Lee, "Probabilistic properties of a nonlinear ARMA process with Markov switching," *Communications in Statistics - Theory and Methods*, vol. 34, no. 1, pp. 193–204, 2005, Feb. 2005.

[17] H. Igarashi and K. Watanabe, "Complex adjoint variable method for finite-element analysis of eddy current problems," *IEEE Transactions on Magnetics*, vol. 46, no. 8, pp. 2739–2742, 2010.

[18] A. J. M. Garrett, "Review: probability theory: the logic of science," in *Probab. Risk*, E. T. Jaynes and Law, Eds., vol. 3, no. 3–4, pp. 243–246, 2004.

[19] S. Salamat, B. Khaleghi, M. Imani, and T. Rosing, "Workload-aware opportunistic energy efficiency in multi-FPGA platforms," 2019, https://arxiv.org/abs/1908.06519.

[20] X. Hong, "Study of intergenerational mobility and urbanization based on OLS method and ordered probit mode," in *Proceedings of the 2020 Management Science Informatization and Economic Innovation Development Conference (MSIEID)*, pp. 435–447, IEEE, Guangzhou, China, September 2020.

[21] J. L. Pollock, "Reasoning and probability," *Law Probab. Risk*, vol. 6, no. 1–4, pp. 43–58, 2007.

[22] F. Taroni and A. Biedermann, "Inadequacies of posterior probabilities for the assessment of scientific evidence," *Law, Probability and Risk*, vol. 4, no. 1–2, pp. 89–114, 2005, Mar. 2005.

[23] I. V. Lyuboshenko, H. Maître, and A. Maruani, "Least-mean-squares phase unwrapping by use of an incomplete set of residue branch cuts," *Applied Optics*, vol. 41, no. 11, pp. 2129–2148, 2002, Apr. 2002.

[24] H. J. Cha and M. L. Ahn, "Development of design guidelines for tools to promote differentiated instruction in classroom teaching," *Asia Pacific Education Review*, vol. 15, no. 4, pp. 511–523, 2014.

[25] M. Redmayne, "Objective probability and the assessment of evidence," *Law, Probability and Risk*, vol. 2, no. 4, pp. 275–294, 2003.

[26] D. Hamer, "Probability, anti-resilience, and the weight of expectation," *Law, Probability and Risk*, vol. 11, no. 2–3, pp. 135–158, 2012.

[27] M. Khodas and A. M. Finkel'stein, "Hall coefficient in an interacting electron gas," *Physical Review B: Condensed Matter*, vol. 68, no. 15, Article ID 155114, 2003.

[28] B. H. H. Gade, C. N. Vooren, and M. Kloster, "Probability distribution for association of maneuvering vehicles," in *Proceedings of the 2019 22th International Conference on Information Fusion (FUSION)*, pp. 1–7, Ottawa, Canada, July 2019.

[29] A. M. Mathai and H. J. Haubold, "Applications to stochastic process and time series," in *Special Functions for Applied Scientists*, pp. 247–295, Springer, Berlin, Germany, 2008.

[30] G. Li, X. Ma, and H. Yang, "A hybrid model for forecasting sunspots time series based on variational mode decomposition and backpropagation neural network improved by firefly algorithm," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 3713410, 9 pages, 2018.

[31] U. Yenal and D. Jimenez, "Life data analysis with a joint probability density function," in *Proceedings of the 2020 Annual Reliability and Maintainability Symposium (RAMS)*, pp. 1–6, IEEE, Ottawa, Canada, January 2020.

[32] T. M. F. Alves, R. O. J. Soeiro, and A. V. T. Cartaxo, "Probability distribution of intercore crosstalk in weakly coupled MCFs with multiple interferers," in *Proceedings of the 2019 IEEE Photonics Conference (IPC)*, pp. 1–4, IEEE, San Antonio, TX, USA, September 2019.

[33] S. Guopan, "The effect of probability on risk perception and risk preference in decision making," in *Proceedings of the 2010 International Conference on Education and Management Technology*, pp. 690–693, Cairo, Egypt, November 2010.

[34] M. Deschênes, "Recommender systems to support learners' agency in a learning context: a systematic review," *Int. J. Educ. Technol. High. Educ.*vol. 17, no. 1, p. 50, 2020.

[35] S. M. M. Seyednezhad, K. N. Cozart, J. A. Bowllan, and A. O. Smith, "A review on recommendation systems: context-aware to social-based," [Online]. Available: http://arxiv.org/abs/1811.11866, 2018.

[36] C. A. Coelho, "The generalized integer gamma distribution A basis for distributions in multivariate statistics," *Journal of Multivariate Analysis*, vol. 64, no. 1, pp. 86–102, 1998.

[37] P. G, "Prediction of airline delays using k-nearest neighbor algorithm," SSRN Scholarly Paper ID 3340771, Aug. 2018. Accessed: Jan. 29, [Online]. Available: https://papers.ssrn.com/abstract=3340771, 2022.

[38] M. Srifi, A. Oussous, A. Ait Lahcen, and S. Mouline, "Recommender systems based on collaborative filtering using review texts—a survey," *Information*, vol. 116 pages, 2020.

[39] A. Platis, N. Limnios, and M. Le Du, "Dependability analysis of systems modeled by non-homogeneous Markov chains," *Reliability Engineering & System Safety*, vol. 61, no. 3, pp. 235–249, 1998.

[40] K. Siu, *Operational Modal Analysis*, Springer, Accessed: Apr. 27, 2022. [Online]. Available: https://link.springer.com/book/10.1007/978-981-10-4118-1, Berlin, Germany.

[41] A. Bates and W. U. Hassan, "Can data provenance put an end to the data breach?" *IEEE Security & Privacy*, vol. 17, no. 4, pp. 88–93, 2019.

[42] Z. Hassanzadeh, R. Biddle, and S. Marsen, "User perception of data breaches," *IEEE Transactions on Professional Communications*, vol. 64, no. 4, pp. 374–389, 2021, Sep. 2021.

[43] B. He, Y. Cui, J. Chen, and P. Xie, "A spatial data mining method for mineral resources potential assessment," in *Proceedings of the 2011 IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services*, pp. 96–99, IEEE, Fuzhou, China, June 2011.

[44] Z. Fang, M. Xu, S. Xu, and T. Hu, "A framework for predicting data breach risk: leveraging dependence to cope with sparsity," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2186–2201, 2021.

[45] R. R. Subramanian, R. Avula, P. S. Surya, and B. Pranay, "Modeling and predicting cyber hacking breaches," in *Proceedings of the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 288–293, IEEE, Madurai, India, February 2021.

[46] N. W. J. Yan and H. N. Chua, "A path analysis model to identify the effects of social media, news media and data breach on data protection regulation awareness," in *Proceedings of the 2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, pp. 1–6, IEEE, Kota Kinabalu, Malaysia, September 2020.

[47] K. Achim, *Probability Theory*, Springer, Accessed: Apr. 27, 2022. [Online]. Available: https://link.springer.com/book/10.1007/978-1-4471-5361-0, Berlin, Germany.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 8634784, 20 pages
https://doi.org/10.1155/2022/8634784

Research Article

A Modified ResNeXt for Android Malware Identification and Classification

Marwan Ali Albahar,[1] **Mahmoud Said ElSayed**,[2] **and Anca Jurcut**[2]

[1]*School of Computer Science, Umm Al-Qura University, Mecca, Saudi Arabia*
[2]*School of Computer Science, University College Dublin, Belfield, Dublin, Ireland*

Correspondence should be addressed to Marwan Ali Albahar; marwanalialbahar@gmail.com

Received 23 February 2022; Revised 15 March 2022; Accepted 28 April 2022; Published 20 May 2022

Academic Editor: Konstantinos Demertzis

It is critical to successfully identify, mitigate, and fight against Android malware assaults, since Android malware has long been a significant threat to the security of Android applications. Identifying and categorizing dangerous applications into categories that are similar to one another are especially important in the development of a safe Android app ecosystem. The categorization of malware families may be used to improve the efficiency of the malware detection process as well as to systematically identify malicious trends. In this study, we proposed a modified ResNeXt model by embedding a new regularization technique to improve the classification task. In addition, we present a comprehensive evaluation of the Android malware classification and detection using our modified ResNeXt. The nonintuitive malware's features are converted into fingerprint images in order to extract the rich information from the input data. In addition, we applied fine-tuned deep learning (DL) based on the convolutional neural network (CNN) on the visualized malware samples to automatically obtain the discriminatory features that separate normal from malicious data. Using DL techniques not only avoids the domain expert costs but also eliminates the frequent need for the feature engineering methods. Furthermore, we evaluated the effectiveness of the modified ResNeXt model in the classification process by testing a total of fifteen different combinations of the Android malware image sections on the Drebin dataset. In this study, we only use grayscale malware images from a modified ResNeXt to analyze the malware samples. The experimental results show that the modified ResNeXt successfully achieved an accuracy of 98.25% using Android certificates only. Furthermore, we undertook extensive trials on the dataset in order to confirm the efficacy of our methodology, and we compared our approach with several existing methods. Finally, this article reveals the evaluation of different models and a much more precise option for malware identification.

1. Introduction

Malware has many different definitions specified by different scholars and researchers depending on the attack vector deployed or harm caused. However, all researchers agreed on the same meaning in general, that is, malware applications have an evil intent [1]. Malicious software (malware) is any software with malicious intent. Malicious code is designed to disrupt normal functioning, display unwanted advertising, control the user's device without their awareness or knowledge, steal or gather sensitive information, and delete or encrypt important data [2]. Unintentionally, harmful software and malware are collectively referred to as "bad ware." The newly developed malware, which is very sophisticated, can obstruct emulators and elude deep static analysis. Malware can also be spread through metamorphic techniques such as instruction permutation, registry modification, encryption, antidebugging, multipacket, virtual machines, and code transformation. It has the capability to launch the payload intelligently to evade detection techniques [3–7]. Many new variants of malware can be generated using automation and reused development modules [8–10]. Like computer systems, malware systems have evolved enormously to be smarter, more intelligent, and more decisive. The main categories of malware are botnets, ransomware, viruses, rootkits, worms, and Trojans [10].

Malware can avoid detection by using polymorphic and metamorphic techniques [3, 11–13]. Malware developers frequently modify minor portions of the original source code in order to create new variants and avoid detection [9, 14, 15]. This makes it extremely difficult to distinguish malware variants from the same family [16, 17].

Malicious Android applications can infiltrate smartphones to be able to do anything without the user's knowledge, such as stealing information, blocking access to critical information from the device, or even mine cryptocurrency. Currently, the rate shows an incredibly high increase in Android malware samples (malicious Android apps) and their variants keep proliferating. The McAfee mobile threat report in Q1 2020 [18] announced that the size of new detected malware attacks reached 800,000 in the 4th quarter of 2019, exceeding the previous quarter of 35 million malware attacks.

Some of these attacks are not easy to detect since the attacker mimic the same normal behavior. For example, the attackers develop a new malware, named MalBus, to avoid any detection by using the original developer's Google Play account.

Moreover, the attacker can collect sensitive military and political information using scannable devices as well as target Google account login information using a phishing-style fake login page. Besides, a new Android malware family (i.e., LeifAccess), which is also known as Shopper, works by exploiting accessibility advantages to create accounts and post fake reviews on the Play Store. After installing it, it promotes click fraud without displaying an icon or a shortcut. As a result of the rising threat from an ever-increasing number of mobile malware instances and new malware families emerging, the Android ecosystem's security will be affected. In order to combat this threat and protect mobile users and systems [19, 20], many studies have been conducted to look for ways to detect and classify Android malware samples. As follows, there are many problems related to the classification and detection of malware: the problems of binary classification in which an app's malignancy is attempted to be determined, multiclass classification problems that include attempting to classify detected malicious apps into a known or unknown family, which is related to identifying malware families, and many others. So, malware researchers should be focusing their attention on the most dangerous families instead of focusing on individual malware samples or lower-risk families if samples are correctly classified and accurately characterized [19]. As a result, an effective malware family classification can assist malware analysts in identifying more malware samples by recognizing and understanding the characteristics of other malware samples in a family. On the other hand, there is a more challenging task than malware detection, which is the classification of malware families. The reason for that is that the numbers of malware samples vary extremely between different families [19–22].

Windows malware families have been the subject of several malware classification studies [9, 17, 23–25]. Due to the different structures and characteristics between Windows malware and Android malware, it is not applicable to utilize the same methodologies to categorize Windows malware families for Android malware families [25, 26]. Consequently, Android malware samples and variations have recently received a lot of attention from academics and the industry.

A recent publication [27] sheds light on the advancement of malware detection through the use of the ResNeXt model. This advancement is due to the architecture of the model, which combines the features of the ResNet and InceptionNet architectures. In addition, it requires low flops and applies the skip concept from ResNet architecture. Motivated by this fact, we proposed a modified ResNeXt model for Android malware classification. The proposed model operates on raw bytes, obviating the need for decryption, disassembly execution of code, and reverse engineering in order to identify malware. To extract the quality information, we converted the malware's nonintuitive features into fingerprint images. Seeing through malware binary, we evaluated the performance and generation of the proposed model to view the capability of discovering and extracting insights necessary for malware analysis and to pave the path for the development of effective malware classification systems.

The main contributions of this work are enumerated as follows:

(i) Proposes an effective modified ResNext classification network for automatically classifying Android malware families from raw malware samples. The Drebin dataset was used to test and validate the proposed system. This dataset contains 5560 applications from 179 different malware families.

(ii) Investigates fifteen different combinations of Android malware file structures in order to classify and generate malware images. In addition, we observed that CR combination of malware image is the most suited feature for malware identification and classification.

(iii) We extract composite features by designing a modified ResNext with a new regularization technique. In particular, we used the standard deviation of the weight matrix to create an adaptive weight decay form in order to prevent the model from taking values.

The rest of this study is organised as follows: Section 2 provides a brief background about the structure of the API files, CNN, deep residual networks, and regularization techniques. Section 3 provides an in-depth view of the existing promising countermeasures that have been produced to monitor and detect Android malware categories. The dataset analysis, methodology, and proposed model are described in Section 4. Section 5 discusses the experimental results and analysis, while an overall discussion of this research and the limitations are introduced in Section 6. Finally, the study' conclusion is discussed in the last section.

2. Background Theory

This section provides a brief background to introduce the structure of the API files and other classifiers and regularization techniques, which are applied for our classification problem.

2.1. Structure of APK Files. Android has become one of the most popular operating systems for smartphones. Since the Android operating system is open source, cybercriminals are attracted to using it. This section goes over the Android application, common Android malware families, Android malware analysis, and Android malware detection techniques. Generally, Android applications are mainly written using the Java programming language and then gather data and source files into an archive file called the Android application package (APK) (Android package) [28]. The APK is shared in the application market and is used for the installation of applications. However, the APK is a ZIP file consisting of multiple files, making it necessary to unzip it before use [29]. The Android APK structure is shown in Figure 1, and the details of each file and folder component are given in Table 1.

Malware families are groups of malwares with similar characteristics, behavior, and capabilities, such as stealing information from a location or a remote server, sending paid or malicious SMS messages, and so on. Malicious behavior uses the same package names as the attack for injecting a payload. In addition, the identity (signature) of a group of malwares (family) can be determined by repeating the use of package names (or other common characteristics) [19]. The most common malware families are given in Table 2 [18].

A significant amount of time is required to manually create features throughout the Android package (APK) structure for Android malware family classifications [3, 4, 30, 34, 35]. These safety mechanisms require significant computer resources, and their deployment in a restricted smartphone environment is challenging [31]. Android malware traces have been studied through classes.dex (CL), resource (RS), manifest, and Android application certificate (CR) files. Malware detection technology, as well as malicious code, have both been developed over time. It is necessary to analyze malware in order to be able to detect it. There are several methods for analyzing and classifying malware, including static, dynamic, hybrid, visualization (image), and audio [32, 33]. Static analysis and classification are the most common methods. Anti-malware signatures and behavioral techniques such as static and dynamic analyses are the most important techniques for identifying malware. Intelligent malware, on the other hand, employs dynamic analysis in conjunction with antiemulation technology [28, 31, 36]. In order to use dynamic and static techniques on such files, a significant amount of manual effort or human intervention is required. To reverse engineer or analyze an application, it is necessary to have prior knowledge of the domain [29, 37–40]. Table 3 provides a comparison between two types of analysis, which are static and dynamic analyses.

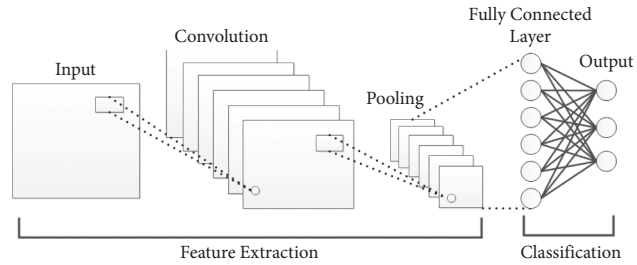


Figure 1: A typical architecture of the CNN.

However, the dynamic approaches experienced with false-positive rates can hinder their wild deployment in real applications. In general, dynamic malware detection is resilient to metamorphic and polymorphic malwares. However, they are slow, resource-consuming, and vulnerable due to the limitation of code reachability. Hence, they may be including a false positive rate. In practice, the weight value for each feature of an application indicates how significant the feature contributes to the classification result in the model pool that participates in the weighted voting that derives the classification result for the application. So, the abnormal feature values in terms of sensitive access might cause some benign apps to be falsely classified as malware. Thus, we can see the dynamic methods are not sufficient and scalable to trace many malicious apps. Moreover, since malware coders have more experience using traditional detection and classification techniques, it will be easier for them to create new malware that can circumvent the current security mechanism. For this reason, researchers have been working for the past few years to develop a new, faster method of automatically detecting, visualizing, and classifying malware. In 2011, Nataraj et al. [24] proposed a completely new approach to malware visualization and analysis using the image processing concept. This is done by visualizing the malware as binary images, extracting its features, and then classifying it. However, the system classifies malware into different families based on whether it has the same visual features, similar texture, and similar structure as malware belonging to another family [2]. This technique overcomes several limitations inherent in static and dynamic analyses. This motivates us to focus on visual-based analysis as it provided a new direction for deploying convolutional neural network (CNN) algorithms for the purpose of detecting malicious software effectively. The images generated by the visualization approach have a variety of layouts, styles, and forms. Thus, malware images have distinct visual similarities and characteristics that set them apart from benign images, which are distinguished by a variety of distinct stripes. These striking distinctions in the visual characteristics of acquired benign and malware images help us classify them according to their families.

2.2. Convolutional Neural Network (CNN). CNN is one of the neural network (NN) algorithms widely used for computer vision. However, in the traditional NN, the hidden layers are fully connected with each other, which

Table 1: Structure of APK files.

Reference	APK folders/files	Responsibilities
[30–33]	AndroidManifest.xml	It is one of the most important files in the Android application, which stores the basic information for the applications and includes the app components, such as activities, services, broadcast receivers, content providers, and others, in addition to package information, such as permissions, package name, and app ID. It also reveals the SDK version.
	Assets/folder	The assets include the assets of an application directory, like images and files, which can be put in this folder and accessed by the asset manager object to retrieve the application assess detailed in the assets folder.
	Lib/folder	This folder contains the native code libraries. The software layer of a processor relates to a specific type of gather inside in this folder.
	META-INF/folder	This includes three main files, which are the signatures certifications, and manifest files for the APK such as MANIFEST.MF, SF, and ∗. RSA.
	Res/folder	This folder includes a description of the resources such as icons, music, images, string, resources, and layouts. These resources are not compiled in resources, arsc folder.
	Classes.dex	Dex code represents bytecode for Android applications which is generated after the compilation of the Java code. which contains multiple constructs for all classes composed like file header, string table, local variable list, class definition table, and method list and can be understandable by the Dalvik virtual machine. Any change in the dex file will affect the APK.
	Resources.arsc	This includes an application's resources in a binary format, like strings, styles, and the paths of images or layouts files, which are a part of this content. However, the data can only be processed in an XML format.

Table 2: Common android malware families.

Common android malware families					
Accu Track	Counterclank	FakeTaoBao	Kidlogger	Placms	SpyOO
Ackposts	Crusewind	FakeTimer	KMIN	Plankton	Ssucl
Acnetdoor	Dogowar	FakeUpdate/Apkqug	Ksapp	Podec	Steek/Fatakr
Adsms	Dougalek	Fakevertu	LeNa	PoisonCake	Tascudap
Airpush/StopSMS	DroidDeluxe	Find and Call/Fidall	Lien/	ProxyTrojan/ NotCompatible/NioServ	TapSnake/ Droisnake
Anserver/Answerbot	DroidDream	Finspy	Locker/SLocker Ransomware	Qicsomos	TGloader/ Stiniter
Antares/	DroidJack/ SandoRAT	Fjcon	Loicdos	Raden	TigerBot
Antammi	DroidDreamLight	Flexispy	Loozfon	Repane	Tetus
Arspam	DroidKungfu	Foncy	Lovetrap/Luvrtrap	Roidsec/Sinpon	Titan
AVpass	Droidsheep	Fokange/Fokonge	Luckycat	RootSmart/ Bmaster	Tonclank
BackFlash/Crosate					
Badaccents	DSEncrypt	Fonefee/Feejar	Maistealer	RuFraud	Tracer
Badnews	Extension/Monad	Gamex	Malap	Saiva	TypStu
BankBot	FaceNiff	Gazon	Mania	Samsapo	UpdtBot
Basebridge	FakeAngry	Geinimi	MMarketPay	Scavir	UpdtKiller
BeanBot	FakeApp.AL	GGTracker	MobiDash	Scipiex	Uracto
			MobileSpy/Godwon	SaveMe/SocialPath	
Beita	FakeAV	GingerBreak	MobileTx	Sndapps/Snadapps	USBcleaver
Binv	FakeDaum/Vmwol	GingerMaster/ GingerBreaker	Mobinauten	SMSsniffer	Uten
BgServ	FakeBank	Godwon	Moghava	SpamBot	Uxipp
			Nandrobox	SeaWeth	
			Netisend	Selfmite	
Biige	FakeDefender	GoldenEagle/GlodEagl	Nickispy	Skullkey	Vdloader
			Obad	Smack	
			Oldboot/MouaBad	SMSpacem	
			OpFake	SMSilence/SMSCatcher	

Table 2: Continued.

Common android malware families					
Bosster Boxer Cajino Carberp Cawitt Code4hk/xRAT	FakeDoc FakeFlash FakeInst FakeJobOffer FakeMarket FakeMart	GoneIn60seconds GPspy HeHe Hidelcon HippoSMS HongTouTou/Adrd	PDAspy Penetho Photsy/Phopsy Pincer	SMSCatcher SMSreg Spitmo SMSspy SPPush	Walkinwat/ Pirater Waps/Simhosy Wroba/ HijackRAT YZHC Zeahache Zitmo/Citmo
Chulli Cellspy Coogos CopyCat Cosha	FakeNefix FakeNotify FakePlay FakePlayer FakeRegSMS	Iconosys Imlog Jifake JollyServ Jsmshider/Xsider	Pjapps	SpyBubble	ZergRush ZertSecurity Zsone

Table 3: Compare between static and dynamic analyses.

	Static	Dynamic
How it works	The suspected code is analyzed without the application being run during static analysis. This method involves disassembly of source code and analyzing it to check the presence of malware without executing the source code and depend only on malware abstraction characteristics and application byte code. Mostly, reengineering is applied [41–43].	The suspected code is analyzed during the runtime execution. It focuses on the characteristics and traces of suspicious use during implementation [24–26, 44–46]and also focuses on system calls and application programs. It is a cybernetic environment used for the execution of code.
Advantage	Harmful applications are not needed to be installed on the device. Do not execute or run the malware code. Applications are in format APK or archive in a zip package [41–43].	It can detect dependencies that are impossible to detect in the static method. Collects temporal instructions. Deals with real data, whereas. in the static analysis, you cannot know input files to be passed for analysis. It can overcome string detection issues, such as malware fitting and pleomorphism [41–43].
Disadvantage	This technique does not take into consideration the analysis of unknown malware. The source codes used are not directly available, and it must be disassembled to extract the features [16–21]. Harmful applications cannot appear until the code has been run. Suffers from code obfuscation [9, 17, 22, 23].	Can have a negative performance impact on the application. Requires better mobile security at critical monitoring stages. It can give incorrect results for similar behavior of the malicious applications with staring applications. It is a complex and time-consuming technique that requires high resource usage and storage capacity [22, 47].

significantly increases the number of training parameters, and this can increase the complexity of the classifier. The CNN produces the concept of parameter sharing in each layer to solve the limitations of the traditional NN and reduce the explosion of weight vectors. The weight sharing concept of the CNN can reduce the computational cost and the training time of the model classifier compared to the other DL models. The CNN is constructed from three core layers, i.e., convolution, pooling, and fully connected layers [48]. The simple architecture of the CNN is shown in Figure 1. The convolution layer is produced as a result of the linear operation of the kernel or filter with the previous output layer. The Relue activation function is widely used in the CNN to increase the degree of nonlinearity and to remove also all negative values of feature maps. The CNN can include more than a convolutional layer. The first layer is used to capture the simple features such as corners or edges. While, the higher layers are mainly used to learn the high-level features. The CNN does not only have the capability to extract the discriminatory features from the input data but can also reduce the spatial size of the convolved feature through the pooling layers. The CNN is implemented in several available architectures, such as ZFNet [49], ResNet [50], GoogLeNet [51], VGGNet [52], AlexNet [53], and LeNet [54]. Motivated by the success of the CNN in various application domains, we utilize the CNN for Android malware detection. The parameters' sharing and the concept of dimensionality reduction are the factors that inspired us to use the CNN for the detection problem.

2.3. Deep Residual Networks. In 2016, Microsoft Research Lab [55] released the deep residual network to solve the inherent problems of traditional deeper networks. The traditional models are more difficult to train and exposed to the degradation problem (of training accuracy). Adding new

layers to the deeper networks not necessarily improves the training accuracy. In most cases, the accuracy becomes saturated and will degrade rapidly with increasing of the network depth, and this can lead to high training errors. On the other hand, the deep residual network works to overcome the aforementioned limitations by using the residual blocks. The simple structure of residual building block is shown in Figure 2.

The potential of this type of networks lies on the concept of "skip connections" to improve the accuracy of the models. Instead of the consecutive connections of the layers in the neural network, some of the layers are skipped and feed the output of one layer as the input to the next layers. The residual blocks are stacked together in a sequential way to consist the residual. The identity mapping of deep residual is descried according to the following formula:

$$X_{i+1} = X_i + F(X_i, W_i), \tag{1}$$

where F is the residual function, W_i is the wight parameters of the block, and X_i and X_{i+1} represent the input and output of the ith unit in the network. In recent days, the deep residual approved its robustness in several computer vision applications. Motivated by its successfully achieved results in different tasks, we also utilized deep residual in this article for Android Malware detection.

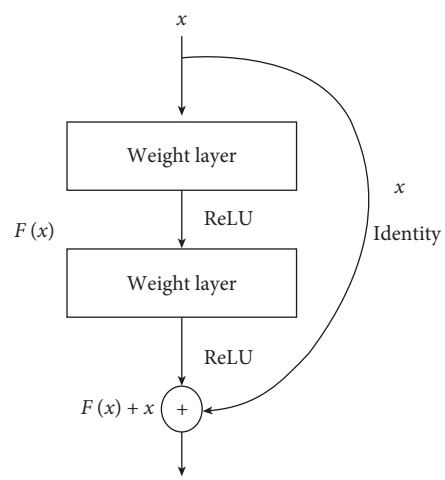


Figure 2: Residual learning: a building block.

2.4. Regularizer Technique

2.4.1. Regularization: A Method of Controlling the Model from Complexity. One of the big challenges in machine learning is how to build a more robust model that can perform effectively in the training data and the new testing data as well. However, the overfitting is one of the significant problems that can hinder the normal operation of machine learning techniques. The model can perform very well during the training, but unfortunately, it performs very poor in the new testing samples, which eliminates its wild implementation for zero-day attacks. The reason behind that returns to the complexity of the training model, resulting from large number of training parameters. As a result, the model can learn the noise in the input data as specific features to discriminate between different attack classes. To eliminate this problem and reduce the error of the prediction model, we can use regularization techniques. The key idea behind the regularization methods is to penalize the model by dropping some of its weight parameters, and this can increase the model's performance on unseen data detection. There are many several techniques widely used for the regularization process. L1 and L2 are the most two popular regularizer methods and comprehensively used in the domain area of machine learning [56]. In the following subsections, we will discuss these two methods in detail.

2.4.2. L1 or Lasso Regularizer. In the L1 regularizer, the absolute value of the magnitude is added to the new loss function. Penalizing the model with the absolute value will make the weight parameters of the insignificant features to reach to zero. As a result, these features will be ignored totally from the model training, i.e., are not contributing any more for the classifier's boundaries. So, we can find the L1 regularizer is widely used for feature selection purposes to selectively reduce or eliminate unnecessarily features [56]. The L1 regularizer is denoted mathematically as follows.

$$\lambda \sum_{i=1}^{n} |w_i|, \tag{2}$$

where λ is the new hyperparameter used for regularization, n represents the number of features in dataset, and w gives the weight values of each feature.

2.4.3. L2 or Ridge Regularizer. In the L2 regularizer [56], we added the square value of the magnitude to the new loss function instead of the absolute value like in L1. Thus, the weight of less important features will reach close to zero but never reach digit zero itself, i.e., the weight matrix will remain greater than zero. L2 regularizer provides better performance with low loss compared to L1. Since, it considers all features during the training process [57]. The mathematical notation of the L2 regularizer is discussed in the following equation:

$$\lambda \sum_{i=1}^{n} w_i^2. \tag{3}$$

The λ is the hyperparameter parameter and used to impose an additional penalty on the corresponding weight values and n and w represent the number of features and the coefficient value of each feature, respectively.

2.4.4. New Regularizer. As we explained in the aforementioned section above, the L1 regularizer is widely used for feature selection, while the L2 regularizer gives less importance to less significant features. However, both regularizers have substantial drawbacks and inherit some limitations, which hinder their broad use to detect zero-day threats. Unfortunately, L1 and L2 regularizers do not take into account the relationship between entries in a weight

matrix. In other words, they only deal with individual weight values. However, any change in the feature attributes, even a small change, can cause a big difference in the model performance. To solve this problem, we developed a new regularizer method (SD-Reg) based on the standard deviation to efficiently deal with the weight values' dispersion, i.e., the SD-Reg regularizer restrains the learning model of using a diapered range of weight space. It works to create a weight-decay adaptive form by considering the standard deviation of a weight matrix and multiplies it by λ parameter to obtain its regularization term.

It is shown in equations (3)–(5) how to formulate the new regularizer.

$$\lambda\sigma(w), \tag{4}$$

where σ indicates the standard deviation of the following weight values:

$$\sigma(w) = \sqrt{\frac{1}{nk}\left\{\sum_{i=1}^{nk} w_i^2 - \frac{1}{nk}\left(\sum_{i=1}^{nk} w_i\right)^2\right\}}. \tag{5}$$

For each row of the weight matrix, there are k rows; each row corresponds to an individual weight. Whereas, σ stands for the standard deviation's weighted averages. n is the number of columns in each ith row of the weight matrix, which is controlled by the parameter λ. The weight vector has a length of n. In our case, the loss function is

$$\min_w\{f(X, y\colon w) + \lambda\sigma(w)\}. \tag{6}$$

Thus, the standard deviation of w is used to minimize the loss function for w in order to select a specific range of values.

3. Literature Review

Recent years have witnessed a significant increase in the number of Android malware cyberattacks. On the other hand, there are substantial efforts by malware researchers and cybersecurity scholars to develop new reliable techniques in order to identify and mitigate the frequent development of these attacks. As a result, the automated Android malware detection (AMD) to deal with this critical cybersecurity challenge has increased too and cannot be neglected.

On the other hand, comprehensive AMD approaches have witnessed a significant increase in the use of ML and DL techniques to ensure the security of the Android ecosystem. This section represents the most widely employed ML and DL approaches for Android malware detection.

3.1. Malware Detection Based on ML. Several ML algorithms such as logistic regression (LR), random forest (RF), k-nearest neighbor (KNN), and support vector machine (SVM) have been used for Malware attack detection.

In [58], three ML classifiers, namely, KNN, RF, and DT were used for AMD on APK samples, which consisted of 300 benign samples and 183 malwares. The Dalvik opcode was extracted from the classes.dex file and then converted into an 8-bit grayscale image. The GIST descriptor was used later to extract the input attributes from the created images. The experimental results showed that RF provided better accuracy of 84.14%.

Li et al. [59] produced a lightweight model for AMD using the SVM algorithm on a dataset downloaded from Google Play store. The authors used less number of permissions for malware classification instead of using all the requested permissions, i.e., selected the top permissions that are more specific for benign and the top permissions for malware to build the malware detection system.

Wang and Li [60] utilized three different ML algorithms, including NB, DT and K-nearest neighbors, for AMD. The feature selection and reduction techniques were used at the first stage to reduce the number of training features and to create a lightweight model with less number of features.

In the study by Massarelli et al. [20], the dynamic analysis was used to find the resource consumption metrics, including CPU, memory, and network usage, as input features of the training classifier. The SVM was utilized to identify several Android malware families on the Drebin dataset.

Chen et al. [61] proposed TinyDroid for AMD. The authors extracted the opcode sequence from Dalvik Executable files and then used the n-gram to extract the trained features from the opcode data. Four ML algorithms, RF, KNN, SVM, and NB, were used for the classification process to identify the malicious malware from normal APKs.

However, most of the aforementioned methods mainly used feature engineering in order to find the best features of the input data. However, selecting the best features is not an easy task, since the features that can work efficiently for one malware class are not necessarily important for other classes. Besides, the input data almost has a high degree of nonlinearity, and shallow learners have a poor ability to learn the complex and nonlinear structure of the data.

3.2. Malware Detection Based on DL. In recent days, deep learning (DL) techniques have been conducted in several application domains, such as image processing and speech recognition [62]. The high potential of DL in several applications returns to its capability to extract the representation features from the input data automatically without any human intervention. It has the good capability to work in data, which has a high degree of nonlinearity, in contrast to shallow learners, almost experienced with high false alarms as they require hand-crafted features as input. The high performance of DL in intensive domain areas encourages many organizations and enterprises such as Facebook, Google, and Microsoft to deploy DL in various applications [63]. Researchers are also starting to leverage DL for cybersecurity tasks and malware detection as well.

Huang and Kao [64] proposed a new model, namely, R2-D2 for detecting the Android malware. The authors first obtained fixed size colored images from the classes.dex bytecode of the Android archive file. Then, different network models based on the CNN algorithm were used for the

training and classification processes. A realistic dataset with more than 2 million of benign and malicious malware samples was collected in the period from January 2017 to August 2017 for the research work.

Hardy et al. [65] used the stacked autoencoders- (SAEs-) based DL model for malware detection. The model composed of two phases. In the first phase, unsupervised learning was used to extract the discriminatory features of the input data. The API calls extracted from the Portable Executable (PE) files were used as input features for the DL model. In the second stage, fine-tuning based on supervised learning was used to adapt the weights and offset vectors. A dataset collected from the Comodo Cloud Security Center is used for a comprehensive experimental study to compare various malware detection models.

Kim et al. [66] proposed an Android malware detection model using a multimodal deep learning method. Seven diverse features have been extracted from the unzipped APK, i.e., shared library, dex, and manifest files. The collected features are merged together to create a fixed feature vector. In the classification process, the DNN was used on malware samples produced from three different sources, i.e., VirusShare, MalGenome project, and Google Play app store.

Another detection approach, i.e., MalDozer was proposed by Karbab [67] for Android malware classification. The CNN with one convolutional layer followed by another softmax layer and one fully connected layer was used to detect samples of unknown malware families. The classifier approach has been applied on three different datasets, including MalGenome, Drebin, and merged datasets. While, API calls that appear in the DEX file were extracted for input attributes.

Nix and Zhang [68] investigated the CNN for Android application/malware classification. The code included within the classes.dex was examined to obtain API calls as an input attributes. The CNN approach was compared with LSTM and other n-gram-based methods on a dataset collected from the Contagio Mobile repository. The results showed that the CNN outperformed the other classifier techniques.

Suarez-Tangil et al. [3] proposed DroidSieve to classify the malware samples using a static solution. Several features have been used to identify the normal samples from the malicious malware, such as API calls, native code, invoked components, code structure, and permissions. The authors used Drebin and MalGenome datasets for their experimental evaluation. However, the reported results relied on the Drebin dataset during its large size and covered all MalGenome samples. The obtained results approved that the DroidSieve approach successfully achieved high accuracy when using resource-centered features and reducing code analysis.

Along with dynamic features derived from an application's behavioral profile, such as method calls and inter-component communication (ICC) intents, DroidCat [69] identified and classified Android malware. Using apps that have evolved over the last nine years, it classified them with 97% accuracy. It was able to defeat attacks that targeted system calls or sensitive APIs, as well as malware samples that used obfuscation schemes. It outperformed two state-of-the-art techniques in terms of detection accuracy, using MalGenome, Drebin, AndroZoo, and VirusShare datasets [70]. It relied on a variety of machine learning algorithms, including SVM, Naive Bayes, and RF. The RF with 128 trees outperformed all other methods. Because the dynamic malware analysis technique proposed by Ficco [71] is composed of a combination of generic and specialized detectors that are used throughout the analysis process, it is resistant to specific evasion techniques. To address malware evolution, the proposed technique utilized an alpha-count mechanism to investigate the effect of varying the length of the observation time window during run-time on the accuracy and speed of detectors. He demonstrated the technique's efficacy using data from 27 DREBIN families. Additionally, a second validation dataset, spanning the period June 2013 to March 2014, was used as a validation dataset taken from the VirusShare dataset.

In this research work, we propose an accurate and automated vision-based AMD model to deal with the critical cybersecurity challenges, which are difficult to ignore. A fine-tuned DL-based CNN algorithm is developed to efficiently detect malware attacks on Android OS.

On the Drebin dataset, we test several combinations and compare our approach to some state-of-the-art works, such as LeNet, Inception V3, ResNet50, Vgg16, EfficientNetB0, and SARVOTAM. All these methods are considered 2D convolution filter-based models. Also, we conduct extensive experiments and include the basic-1D-CNN with single and multistreams in our computations. The results indicate that the proposed ResNeXt achieved significant results in terms of accuracy and F1-score. In addition, the extensive computations of the ResNext model are significantly lower since it requires a smaller number of features to be analyzed compared to other methods. As a result, the modified ResNext demonstrates its effectiveness by quickly distinguishing Android malware from benign apps with the fewest recorded errors.

4. Materials and Methods

This section describes the methodology and the used dataset for our proposed model. The architecture and detailed explanation of the learning model are discussed in detail, considering the file size and other used parameters for the model tuning.

4.1. Dataset. Most of the studies that have been published between the periods of 2014 and 2020 use the Drebin dataset for training and evaluating their developed models. It is considered one of the most widely used datasets for malware family classification purposes [21]. So, for that reason, it is used in this experiment setup. The Drebin dataset has the most popular Android malware families, which were collected in the period between August 2010 and October 2012. It contains more than 5,560 files belonging to 179 special malware families like Fake Installer, GoldDream [72], GingerMaster [73], DroidKungFu [74], and many others. Table 4 provides the outlines of different malware datasets that have been used by the research community.

Table 4: Various malware datasets' publication counts.

Dataset	Number of publications
Drebin	20
Repository	8
Collection	6
MalGenome	17

4.2. Transforming Malware APK into Images. The fundamental files considered for visualization in APK are classes.dex, resources, manifest, and certificates. In this work, these four types from the malware APK files are employed to extract the malware images, which are used for our model training.

First, the binaries are transformed into 8-bit vectors, and in the next stage, these vectors are converted to grayscale images. The detailed procedure is discussed. Initially, a malware substring consists of a sequence of numerous substrings where each substring is 8-bit long and is called a pixel. The 8-bit substring is converted to a decimal number in the next step, ranging from 0 to 255. Furthermore, all the malware substrings were transformed into a one-dimensional vector and converted to a two-dimensional matrix of a specific width. We called it a "malicious code matrix." This matrix is considered the two-dimensional grayscale image. The conversion process of APKs to grayscale images is shown in Figure 3. The width of the images was fixed based on the size of the APK files given in Table 5. Hence, the height also depends on the file size. CNN-based models require inputs to be of the same shape. Therefore, instead of trying varying sizes of APK files, we use the dimensions proposed by [26]. The main reason behind the chosen sizes is to retain as much information as required along with keeping the size compact. However, it is empirically decided by [17, 26]. Therefore, to avoid the trial-and-error method for finding the proper sizes, this work follows their proposed procedure. A complete APK can be represented by grayscale images with an underlying structure that follows certain divisions. Fifteen different file structure combinations were used to generate the Drebin Android malware images, each containing at least one image of a distinct malware family. Some images constructed from the files are shown in Figure 4. Classes.dex (CL), AndroidManifest (AM), certificate (CR), and resources (RS) are among the files included.

4.3. Proposed Model. In this work, we use the same ResNeXt classification model, which was proposed by Xie at el. [27] in order to categorize malware families. The basic idea behind ResNeXt is to use an aggregated residual block instead of the basic residual block. This strategy is called "split-transform-merge," and it was implemented in the inception architecture [27].

The inner product of a synthetic neural network is the weighted sum of the primary neurons in each layer, which is calculated for each layer separately. When viewed through

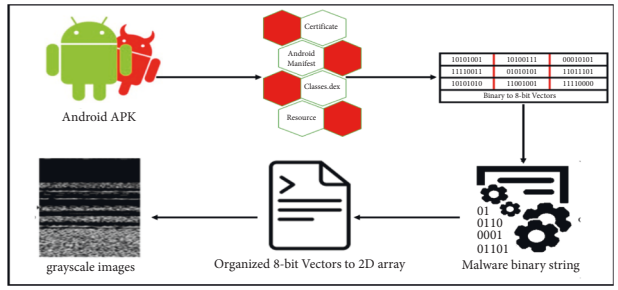


Figure 3: Conversion process of APK into grayscale image.

Table 5: Fixed image width according to the file size.

File size	Width
<50 kB	64
50 kB–100 kB	128
100 kB–200 kB	256
200 kB–500 kB	512
500 kB–1000 kB	1024

the lens of equation (7), the inner product can be thought of as a type of aggregate transformation.

$$\sum_{i=1}^{D} w_i x_i, \tag{7}$$

where w_i represents the filter weight for the ith channel of the neuron, and x_i describes the D-channel input vector of the neuron.

A more inclusive function, which can perform as a network itself, has been developed by Xie at el. instead of a simple aggregating transformation [27]. They demonstrated aggregated transformations as

$$F(x) = \sum_{i=1}^{C} t_i(x), \tag{8}$$

where $t_i(x)$ is an arbitrary function. Analogous to a simple neuron, here t_i projects x into an (optionally, low-dimensional) embedding and then transforms it. C represents the size of the set of transformations to be aggregated, while Xie at el.' study used C to represent the cardinality. Their study claimed that the dimension of cardinality can control great numbers of complex transformations. Figure 5 [75] shows 32 cardinality blocks of ResNeXt.

$$y = x + \sum_{i=1}^{C} t_i(x). \tag{9}$$

The aggregated transformation in (8) serves as the residual function, as shown in (9).

ResNeXt is designed using ResNet's skip concept and cardinality with better accuracy than a wide and deep network. In experiments using the ImageNet dataset, ResNeXt outperformed existing models in terms of accuracy. Due to these advantages, we utilize the ResNeXt model

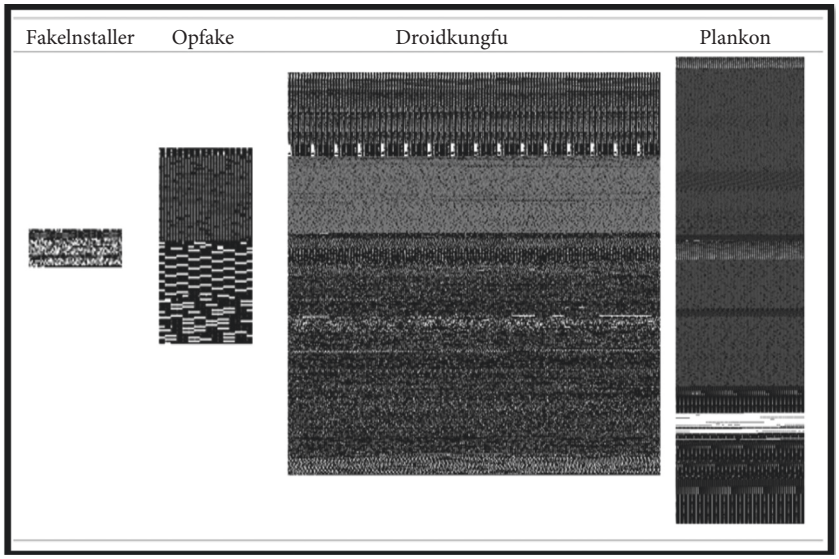


Figure 4: The fingerprint images of different malware families using file sections of the Android certificate (CR), AndroidManifest (M), classes.dex (CL), and resource (RS) of an APK.

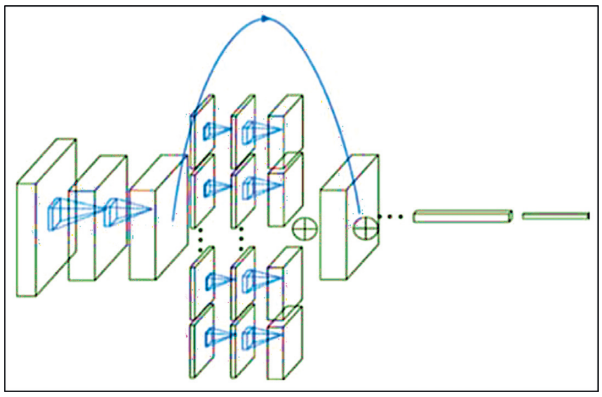


Figure 5: Cardinality of ResNeXt block.

for malware image classification. The architecture of ResNeXt-50 is shown in Figure 6.

A new regularization technique was incorporated into our model in order to control individual weight values and the relationship between weight matrix entries. This is done by taking the weight matrix's standard deviation and multiplying it by λ to create an adaptive weight decay form. Thus, the regularizer prevents the learning model from taking values from the weight space that are too widely distributed. In fact, the new regularizer has been extensively tested on various tasks with different datasets and proved to be more effective than other regularization methods [57, 76–80].

Our model uses the new regularizer technique to control the relationship between entries using weight values from the weight matrix. To create a weight decay adaptive form, we multiplied the standard deviation of the weight matrix by λ. This reduces model complexity by removing unnecessary data and keeping only data that are useful for classification. The explained regularizer was tested extensively in different domains, including computer security, and provided better performance than the other regularization techniques.

5. Results

In this section, we briefly discuss the experimental setup and explain the classification results of the proposed approach.

5.1. Classification Module Training and Validation. In this research, we used the Drebin dataset with a focus on the top 20 malware families as follows: BaseBridge, Plankton, DroidDream, SMSreg, FakeInstaller, OpFake, SendPay, FakeRun, Imlog, FakeDoc, ExploitLinuxLotoor, Iconosys, DroidKungFu, Adrd, Glodream, Gappusin, Kmin, MobileTx, GinMaster, and Geinimi. Table 6 provides Drebin malware class combinations and associated instances. The modified ResNeXt has 50 layers except for the input layer and the fully connected layer. 48 of the 50 layers are divided into 16 blocks. Each block contains three layers, with a total of 32 cardinalities.

5.2. Experimental Setup. In this work, all experiments have been executed using Python programming language. Several

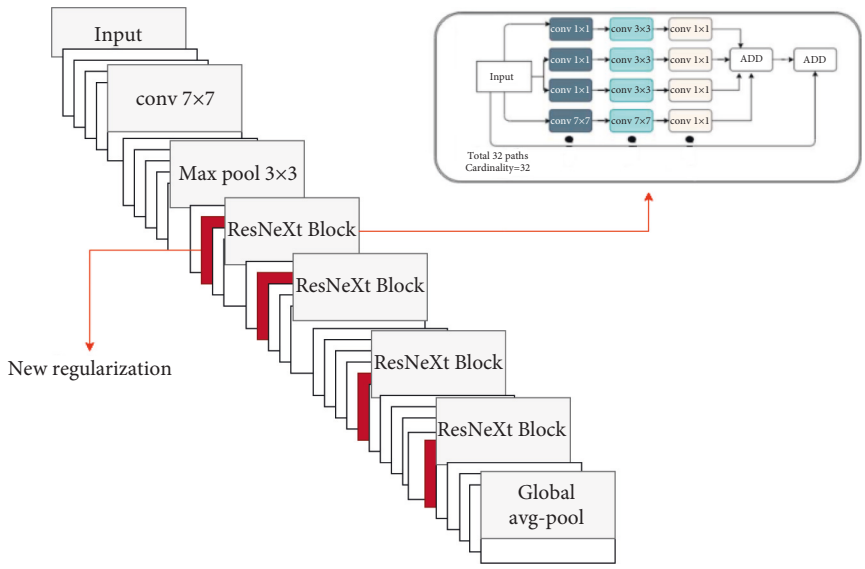


Figure 6: The architecture of the proposed modified ResNext.

machine learning and deep learning libraries such as Keras and TensorFlow are used to build the ResNeXt model.

The simulations were run on a system with 20 GB of RAM and an Intel® CoreTM i3 processor, as well as an NVIDIA GeForce GTX 1080ti graphics processing unit with a frame buffer of 11 GB.

5.3. Experimental Results and Analysis. In our work, the modified ResNext model with an embedded new regularization performed well for approximately 100 epochs. The simulation results were recorded for the Drebin dataset after converting the attributes of raw malware binary executable files to grayscale images. We compared our observed performance measures with various state-of-the-art models, which indicates that the performance of our model is higher (Figure 7). The evaluation is compared for various combinations of the image types as given in Table 7.

The highest accuracy is achieved for the CR combination, as given in Table 7. As a result of the observations and simulation results, it is clear that the maximum amount of relevant information about malicious types is contained in the CR file, resulting in satisfactory classification performance. Apart from classification measures, Table 8 provides a time-based comparison for each combination used in the study and the number of images processed per second that belong to the corresponding class. The relative execution time is within a satisfactory range, which makes it possible to use it in real-time applications.

Thus, after training the model with high-quality classification measures, it can be used for testing in various applications. As given in Table 8, the average processing time for a single image is comparable to previous work [26]. Consequently, once integrated and deployed in software systems, the model's execution performance will be the same as the state-of-the-art method.

The new regularized technique was implemented in order to control individual weight values as well as the relationship between weight matrix entries in order to eliminate unnecessary data while selectively using only data useful for classification. This demonstrates the effectiveness of our model in extracting better features while also maintaining a reasonable overall running time. The detailed confusion matrix for the top 20 malware families is given in Table 9. Table 10 provides the results of family-specific classifications for the original ResNeXt and our modified ResNeXt. As observed, the new regularization method was adaptive in order to avoid overfitting and to improve the CNN's ability to predict whether a new observation of the data was not trained on the model. As a result, it enables a more adaptable method of weight loss. As a result, the regularizer prevents the learning model from using global values from the weight space as input. This reduces the complexity of the model and removes unnecessary data, while keeping only the data that are useful for classification. As shown in Figure 8, the proposed model outperformed all the other methods of family-specific classification in terms of F1-score.

5.4. Malware Family Classification Performance Evaluation. It has recently become a problem for machine learning-based malware classifiers to deal with the evolution of malware, which changes its malicious behavior over time, resulting in the deterioration of the classifiers. It has been suggested that deterioration [20, 22] and model aging [17, 21] are better terms to describe this issue of long-term sustainability. Sustainability is defined as the ability of the classifier to sustain its capabilities over time without frequent retraining. Recently, the sustainability challenge associated with machine learning-based malware detection has been discussed, but with limited investigation depth and solutions.

In the same context, the authors in [20, 22] proposed and compared sustainability metrics with the five most recent Android malware detectors. Another study [81]

Table 6: Various combinations and its associated instances used in the study.

Combination	CR	AM	RS	CL	CR + AM	CR + RS	CR + CL	AM + RS	AM + CL	RS + CL	CR + AM + RS	CR + AM + CL	CR + RS + CL	AM + RS + CL	CR + AM + RS + CL
No. of instances	1826	4659	4659	4660	4659	4659	4660	4659	4660	4660	4659	4660	4660	4660	4660

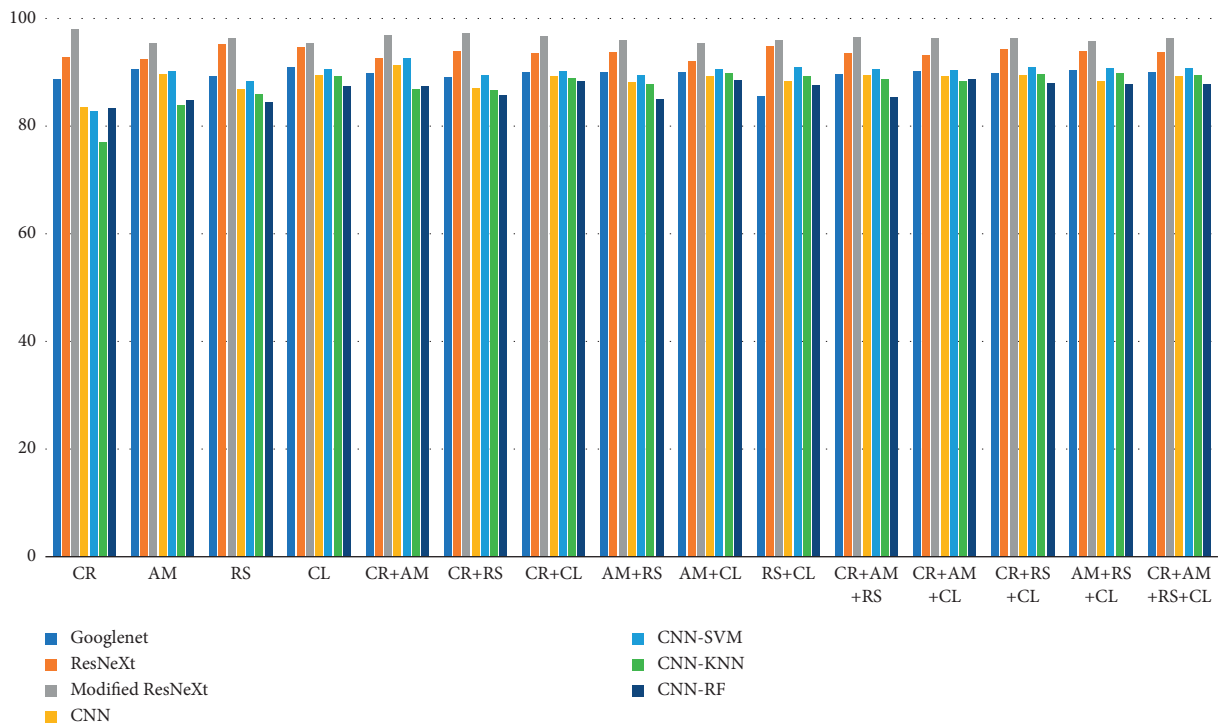


Figure 7: Comparison of combination classification accuracy for each model.

Table 7: Generic and augmented CNN accuracies on 15 different grayscale malware image combinations.

	Image combination	CNN (%)	CNN-SVM (%)	CNN-KNN (%)	CNN-RF (%)	VGG16 (%)	GoogLeNet (%)	ResNeXt (%)	Modified ResNeXt (%)
1	CR	83.58	82.92	77.11	83.42	78.27	88.86	92.96	98.25
2	AM	89.79	90.18	83.94	84.85	85.76	90.76	92.51	95.50
3	RS	86.86	88.56	86.02	84.53	82.12	89.37	95.21	96.50
4	CL	89.46	90.57	89.40	87.58	87.23	91.16	94.74	95.63
5	CR + AM	91.48	92.59	86.93	87.52	90.57	89.81	92.74	96.88
6	CR + RS	87.12	89.47	86.80	85.89	88.91	89.16	94.08	97.38
7	CR + CL	89.33	90.25	89.01	88.43	89.34	90.01	93.85	96.94
8	AM + RS	88.29	89.47	87.78	84.98	86.78	90.07	93.86	96
9	AM + CL	89.33	90.83	89.79	88.69	84.43	90.07	92.06	95.57
10	RS + CL	88.49	90.96	89.34	87.58	84.37	85.77	94.98	96.07
11	CR + AM + RS	89.46	90.77	88.75	85.50	87.67	89.66	93.56	96.75
12	CR + AM + CL	89.33	90.51	88.49	88.82	86.81	90.26	93.40	96.46
13	CR + RS + CL	89.53	90.90	89.66	88.17	84.56	89.80	94.30	96.49
14	AM + RS + CL	88.55	90.70	89.86	87.97	89.29	90.43	94.15	95.88
15	CR + AM + RS + CL	89.33	90.70	89.60	87.84	84.32	90.04	93.86	96.47

Table 8: A comparison of execution time and images processed per second by the proposed model.

S/no.	Combination	Execution time (s)	Images processed/second
1	CR	231.2	6.57
2	AM	663.8	5.1
3	RS	787.4	4.25
4	CL	1102.1	4.21
5	CR + AM	790.2	4.23
6	CR + RS	1004.4	4.64
7	CR + CL	1109.7	4.2
8	AM + RS	850.5	4.34
9	AM + CL	1120.4	4.12
10	RS + CL	1093.3	4.26
11	CR + AM + RS	624.7	5.04
12	CR + AM + CL	1139.4	4.04
13	CR + RS + CL	1235.5	3.78
14	AM + RS + CL	1203.9	3.83
15	CR + AM + RS + CL	1513.7	3.08

Table 9: Confusion matrix for the top 20 malware families in the proposed model.

	FakeInstaller	DroidKungFu	Plankton	OpFake	GinMaster	BaseBridge	Iconosys	Kmin	FakeDoc	Geinimi	Adrd	DroidDream	ExploitLinuxLotoor	Mobile Tx	Glodream	FakeRun	SendPay	Gappusin	Imlog	SMSreg
FakeInstaller	904	0	0	17	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	2
DroidkungFu	0	579	3	2	13	10	0	0	0	0	34	0	6	0	3	0	0	17	0	0
Plankton	1	3	573	0	4	20	0	0	0	11	11	1	0	0	1	0	0	0	0	0
OpFake	0	0	0	560	51	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0
GinMaster	0	1	2	3	315	0	0	0	0	1	12	0	0	0	4	0	0	1	0	0
BaseBridge	0	5	0	1	1	316	0	0	0	0	3	0	0	0	4	0	0	0	0	0
Iconosys	0	0	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0
Kmin	0	0	0	0	0	0	0	135	0	0	12	0	0	0	0	0	0	0	0	0
FakeDoc	3	0	0	0	2	0	0	0	120	0	2	0	0	0	4	0	0	1	0	0
Geinimi	0	0	0	0	3	2	0	0	0	86	1	0	0	0	0	0	0	0	0	0
Adrd	0	1	0	0	1	1	0	0	0	0	88	0	0	0	0	0	0	0	0	0
DroidDream	0	1	0	0	0	0	0	0	0	0	1	78	0	0	0	0	0	1	0	0
ExploitLinuxLotoor	0	1	0	0	3	4	0	0	0	0	1	0	60	0	0	0	1	0	0	0
Mobile Tx	0	0	0	0	0	0	0	0	0	0	0	0	0	69	0	0	0	0	0	0
Glodream	0	2	0	1	1	1	1	0	0	0	2	0	0	0	60	0	0	0	0	0
FakeRun	0	0	21	1	0	0	0	0	1	0	3	0	1	0	4	27	0	3	0	0
SendPay	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0
Gappusin	0	3	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	51	0	0
Imlog	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	41	0
SMSreg	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	39

Table 10: F1-score comparisons between the modified ResNeXt and the original ResNeXt in the Drebin dataset.

Family	ResNeXt	Modified ResNeXt
Adrd	0.67433	0.908108
BaseBridge	0.921283	0.961832
DroidDream	0.962963	0.981132
DroidKungFu	0.916865	0.935737
ExploitLinuxLotoor	0.875912	0.827068
FakeDoc	0.948617	0.988593
FakeInstaller	0.613636	0.99675
FakeRun	0.761194	0.97561
Gappusin	0.858311	0.866667
Geinimi	0.794702	0.891192
GinMaster	0.993464	0.939691
Glodream	0.97619	0.78481
Iconosys	0.957447	0.996721
Imlog	1	0.988506
Kmin	0.934891	1
Mobile Tx	0.93551	1
OpFake	0.991597	0.995938
Plankton	0.95122	0.994378
SMSreg	0.986361	0.886076
SendPay	0.895833	0.944

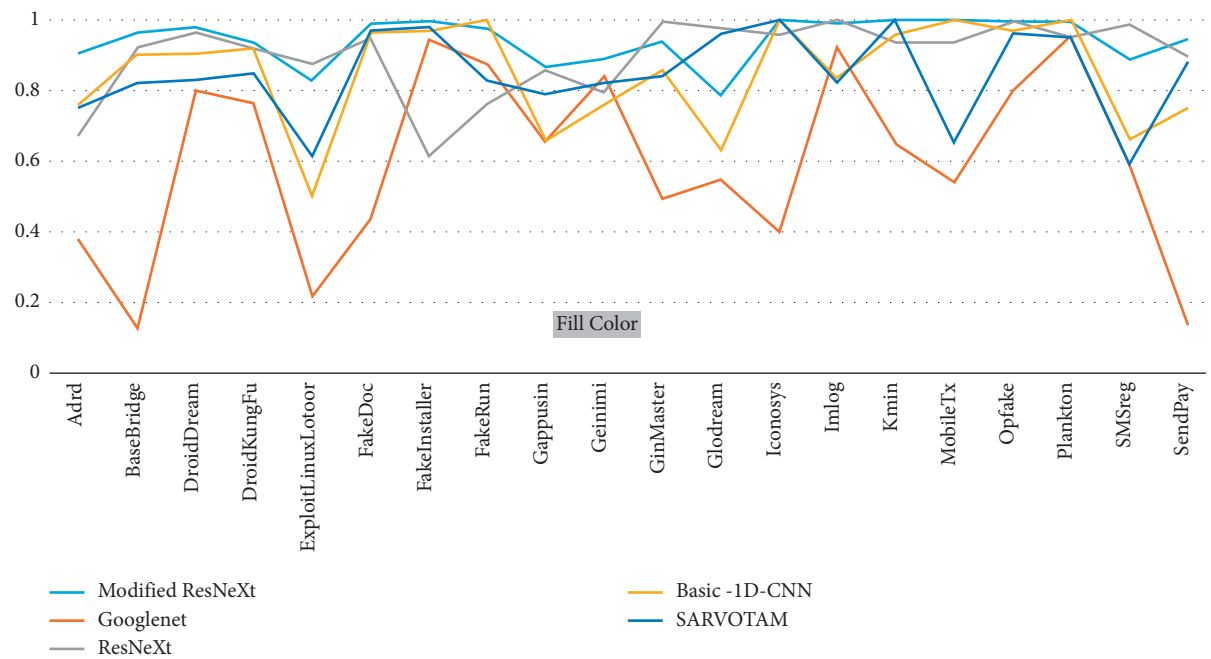


Figure 8: Comparison of family classification F1-score for each model in the Drebin dataset.

outperformed five detectors in sustainability by employing a new behavioral profile for apps. In particular, the authors proposed DroidSpan, which surpassed the five detectors in terms of sustainability. However, their study was limited to malware detection and did not include any discussion of malware family classification. DroidEvolver [82] used a model pool with five linear online learning algorithms and delayed classifiers to perform the necessary updates. API-Graph [21] used API semantic similarity from an Android API relation graph to improve the latest malware classifiers. Therefore, we propose a modified ResNext-based classification network with new regularization for Android malware family classification. Each Android application is distributed via an Android application package (APK). An APK contains multiple folders and files, each of which contains multiple sections; an APK contains multiple folders and files; and an APK contains multiple sections. Among other files and sections, we pay close attention to the AndroidManifest.xml file (AM), classes.dex file (CL), and the certificate files included with each malware sample (CR).

As a result of these sample characteristics, we chose the ResNeXt block because of its simplicity and performance. Following that, we modify ResNeXt to include a distinct block for each component (section or file) of a malware sample in order to account for the differences in characteristics between the components. A new regularization technique is utilized to improve the efficiency of malware family classification by extracting discriminatory features from the malware sample. This enables us to ultimately classify malware samples according to their correct families.

The modified ResNeXt model is divided into two distinct phases: training and testing. During the training phase, it builds a prediction model using a set of labeled samples from the Drebin dataset. Then, the trained model is used to classify samples from the Drebin and AMD datasets during the testing phase. Specifically, the primary motivation and goal for the modified ResNeXt is to demonstrate that this model can improve classification performance once trained on an older dataset and predict new patterns of malware samples from a new dataset without having to retrain on new samples.

We addressed the issue of sustainability by assessing the model's performance when it is trained on the Drebin dataset (collected between 2010 and 2012) and predicting labels for other datasets such as the AMD dataset [10] (collected between 2010 and 2016). We selected the AMD dataset because it was amassed over a longer time period than the Drebin dataset. Our experiment revealed that the feature extraction model trained on the AMD dataset outperformed the model trained on the Drebin dataset in terms of overall performance. This is because the AMD dataset contains more variation information about malware samples than the Drebin dataset, which makes it more suitable to study the evolutionary patterns of malware.

In particular, we divided the samples into three different groups, each with their own set of samples. The first group is the Drebin dataset, where we separated the malware samples from the same year into training, validation, and test sets. The goal is to assess the performance of our model with training and testing malware samples collected in the same period of time. The second group is the AMD dataset. We selected 3,460 malware samples randomly (S-AMD). By doing so, we intend to test the capability of our model in classifying apps that have never been used in training. For the third group (T-AMD), we used the entire AMD dataset, in which we utilized malware samples from different time periods than the training set. The goal is to focus on evaluating the stability of our model performance when it is trained on older datasets and predicting the labels of newer ones, spanning one to four years. Figure 9 shows the experimental results.

In the first experiment, we compared the feature extraction model learned from the Drebin and AMD datasets (same period) to observe the performance of our model. It is clear from Table 11 that the model's generalization is better for Drebin than (S-AMD) and (T-AMD), which showed the suitability of the (S-AMD) and (T-AMD) sets in terms of containing more discriminatory features of malware samples. Next, we compared the accuracy of training and testing on the Drebin and the AMD datasets (same period and over-time). We observed that the accuracy of the Drebin dataset is better than that of the AMD dataset. This is because the Drebin dataset contains fewer malware families than the AMD dataset. In other words, the AMD dataset has sufficient variation

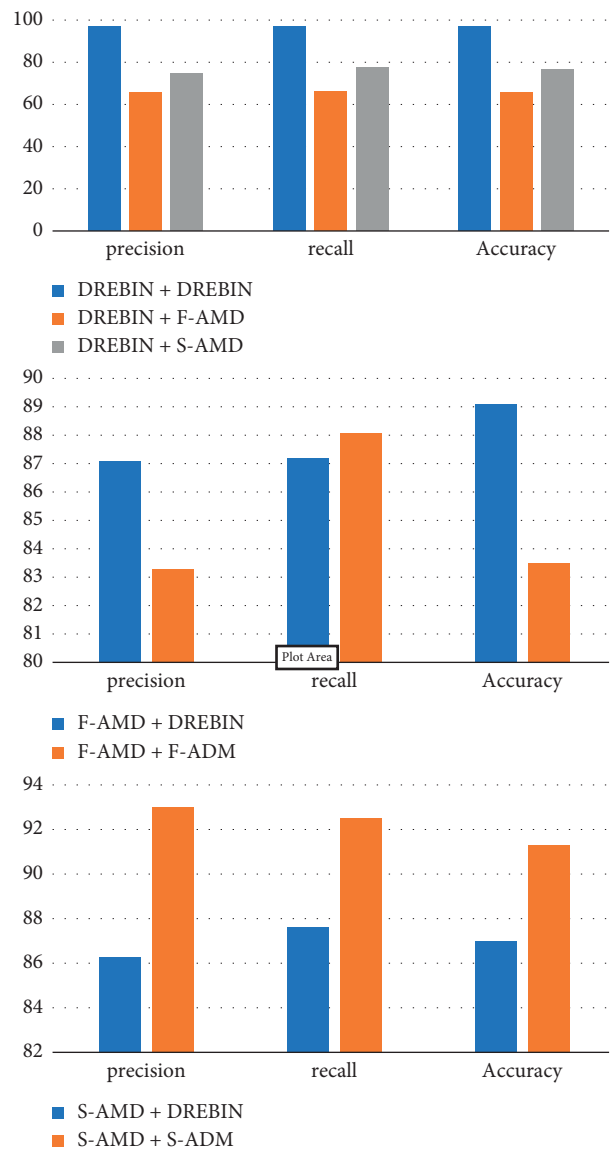


Figure 9: Stability comparison of classification performance on three sets.

Table 11: Stability classification performance of the proposed model for the Drebin and AMD datasets.

Dataset	Precision	Recall	Accuracy
Drebin + Drebin	97.1	97.3	98.2
Drebin + S-AMD	66.2	67.3	66.2
Drebin + T-AMD	75.2	77.8	77.4
S-AMD + Drebin	87.1	87.2	89.1
S-AMD + S-ADM	83.3	88.1	83.5
T-AMD + Drebin	86.3	87.6	87
T-AMD + T-ADM	93	92.5	91.3

information about malware samples, which improves the generalization performance. Thus, the generalization performance improved when the variation information of malware samples in the AMD dataset was learned.

Based on the results from Table 11, it was revealed that, even with a span of four years (the difference between Drebin and AMD datasets collection span), our model detection accuracy dropped noticeably over time from about 98% to below 80% in terms of accuracy for testing samples from year one. Over time, our model tended to be much more stable (with minimal fluctuation) in terms of detecting malware samples. It achieved an average accuracy of 87%, despite the evolution of Android malware. Despite the fact that our results are promising, we cannot claim that our model will continue to perform as well as it has in the case of future malware. The unpredictable evolution of our app in the future would also serve as a trigger for retraining our model in the future.

However, previous studies used a variety of different benchmarks casts doubt on the measurement degree of our findings' validity. As a result, despite the fact that we collected samples from a variety of sources, our datasets may not be representative of the app population during the pertinent years. Our findings and conclusions are best understood in light of the benchmarks we analyzed.

6. Discussion and Limitation

There are numerous benefits to utilizing DL networks for malware family classification. The DL techniques have the capability to classify data automatically without the essential requirements of some expensive processes, such as decryption or reverse engineering. However, to successfully build a lightweight classification model and to avoid the high computational cost, the size of training features must be reduced to speed up the training and detection process.

The distribution of the discriminant information is controlled via the new regularizer by constraining the weight values' dispersion. In other words, the standard deviation of the weight matrix was used to obtain the regularization term and then multiplied by λ. The motivation is to develop a weight-decay adaptive form that helps the regularizer prevent the learning model from extracting values from the weight space that are widely distributed. Thus, it helps the model extract features that are effective for malware family classification. Thus, the modified ResNeXt has the best classification performance among all the methods because we used a new regularization discriminant information distribution to eliminate unnecessary data and selectively use only data useful for classification.

The solutions that were representing the malicious app behaviors using dynamic features (e.g., DroidSieve) suffered from the cost of tracing runtime and scalability, while our approach incurred runtime costs for testing per second.

Additionally, the overall accomplished results for the visual grayscale images may differ from those for the visual color images. This is because color images contain more texture details and visualization features than those included in grayscale images. Also, the tested model used imbalanced Android malware samples for training and validating, so the models need to be tested on balanced Android malware samples. Thus, our proposed model avoided the computational needs of data augmentation and feature-engineering techniques. As a result, we successfully achieved better results and satisfied significant classification performance compared to other existing methods.

7. Conclusions

In this work, we proposed a modified ResNeXt model for the classification of android malware. The ResNeXt is utilized due to its flexibility and requirement for low flops, coupled with a new regularization technique to improve the capability of the model in the classification of android malware. In the first step, various binary malware files from the Drebin dataset were transformed into 8-bit vectors based on the substrings. In the next step, these vectors are converted to grayscale images. For classification, we adopt a modified ResNeXt with skip concept and cardinality to enhance the detection performance. Furthermore, we embedded a new regularization technique to improve the classification detection rate. Different combinations of the images were used to fine-tune the model to look for those files having the most effect on the model. From simulation results, it is concluded that the certificate (CR) is the most suited feature, containing enough information to be used for the identification and classification of malware. We reported various highest measures, including accuracy, recall, precision, and F1 measures, obtained from using the CR images. 97.07% accuracy is observed, which is the highest accuracy so far achieved by using the Drebin dataset.

For intrinsic evaluation of our DL approach, the proposed model is compared with other state-of-the-art techniques, which widely use DL for classification purposes within the Android malware family. In the future, various models can be trained and studied to adopt a less complex model, while enhancing the performance further for the malware classification problem. In addition, when malware authors distribute each malware instance in both its simple and obfuscated forms concurrently, the images between the simple and obfuscated versions are likely to differ. So, utilizing only malware images may be ineffective in this case. A possible solution to this issue is to include another feature that is resistant to obfuscation, packing, and encryption attacks and then combine it with the malware image approach, for example, the extraction of code features from native app binaries and security-sensitive APIs, including reflection-based features.

Data Availability

The datasets used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] M. Ashawa and S. Morris, "Analysis of android malware detection techniques: a systematic review," *International Journal of Cyber-Security and Digital Forensics*, vol. 8, no. 3, pp. 177–187, 2019.

[2] R. Komatwar and M. Kokare, "A survey on malware detection and classification," *Journal of Applied Security Research*, vol. 16, no. 3, pp. 390–420, 2020.

[3] G. Suarez-Tangil, S. K. Dash, M. Ahmadi, J. Kinder, G. Giacinto, and L. Cavallaro, "CODASPY:Data and application security and privacy," in *Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy*, Scottsdale, Arizona, USA, March 2017.

[4] J. Garcia, M. Hammad, and S. Malek, "Lightweight, obfuscation-resilient detection and family identification of android malware," *ACM Transactions on Software Engineering and Methodology*, vol. 26, no. 3, pp. 1–29, 2018.

[5] V. Rastogi, Y. Chen, and X. Jiang, "Catch me if you can: evaluating android anti-malware against transformation attacks," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 1, pp. 99–108, 2014.

[6] O. Mirzaei, J. M. de Fuentes, J. Tapiador, and L. Gonzalez-Manzano, "AndrODet: an adaptive Android obfuscation detector," *Future Generation Computer Systems*, vol. 90, pp. 240–261, 2019.

[7] V. Balachandran, D. J. J. Tan, and V. L. L. Thing, "Control flow obfuscation for Android applications," *Computers & Security*, vol. 61, pp. 72–93, 2016.

[8] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, "Robust intelligent malware detection using deep learning," *IEEE Access*, vol. 7, Article ID 46717, 2019.

[9] J. Fu, J. Xue, Y. Wang, Z. Liu, and C. Shan, "Malware visualization for fine-grained classification," *IEEE Access*, vol. 6, Article ID 14510, 2018.

[10] F. Wei, Y. Li, S. Roy, X. Ou, and W. Zhou, "Deep ground truth analysis of current android malware," in *Proceedings of the Detection of Intrusions and Malware, and Vulnerability Assessment*, pp. 252–276, Bonn, Germany, July 2017.

[11] S. Dong, M. Li, W. Diao et al., "Understanding android obfuscation techniques: a large-scale investigation in the wild," in *Proceedings of the Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 172–192, Singapore, August 2018.

[12] D. Maiorca, D. Ariu, I. Corona, M. Aresu, and G. Giacinto, "Stealth attacks: an extended insight into the obfuscation effects on Android malware," *Computers & Security*, vol. 51, pp. 16–31, 2015.

[13] K. Bakour, H. M. Ünver, and R. Ghanem, "A deep camouflage: evaluating android's anti-malware systems robustness against hybridization of obfuscation techniques with injection attacks," *Arabian Journal for Science and Engineering*, vol. 44, no. 11, pp. 9333–9347, 2019.

[14] N. Xie, X. Wang, W. Wang, and J. Liu, "Fingerprinting Android malware families," *Frontiers of Computer Science*, vol. 13, no. 3, pp. 637–646, 2018.

[15] S. Ni, Q. Qian, and R. Zhang, "Malware identification using visualization images and deep learning," *Computers & Security*, vol. 77, pp. 871–885, 2018.

[16] S. Turker and A. B. Can, "Andmfc: android malware family classification framework," in *Proceedings of the IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*, Istanbul, Turkey, September 2019.

[17] D. Vasan, M. Alazab, S. Wassan, H. Naeem, B. Safaei, and Q. Zheng, "Imcfn: image-based malware classification using fine-tuned convolutional neural network architecture," *Computer Networks*, vol. 171, 2020.

[18] Mcafee, "Mcafee mobile threat report q1," 2020, https://www.mcafee.com/content/dam/consumer/en-us/docs/2020-Mobile-Threat-Report.pdf.

[19] F. Alswaina and K. Elleithy, "Android malware family classification and analysis: current status and future directions," *Electronics*, vol. 9, 2020.

[20] L. Massarelli, L. Aniello, C. Ciccotelli, L. Querzoni, D. Ucci, and R. Baldoni, "Android malware family classification based on resource consumption over time," in *Proceedings of the 12th International Conference on Malicious and Unwanted Software (MALWARE)*, Fajardo, PR, USA, October 2017.

[21] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, and K. Rieck, "Drebin: effective and explainable detection of android malware in your pocket," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2014.

[22] Y. Sun, Y. Chen, Y. Pan, and L. Wu, *Android Malware Family Classification Based on Deep Learning of Code Images*, 2019.

[23] K. Kancherla and S. Mukkamala, "Image visualization based malware detection," in *Proceedings of the IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*, pp. 40–44, Singapore, April 2013.

[24] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: visualization and automatic classification," in *Proceedings of the 8th International Symposium on Visualization for Cyber Security - VizSec '11*, Pennsylvania, PA, USA, July 2011.

[25] Y. Fang, Y. Gao, F. Jing, and L. Zhang, "Android malware familial classification based on dex file section features," *IEEE Access*, vol. 8, 2020.

[26] J. Singh, D. Thakur, F. Ali, T. Gera, and K. S. Kwak, "Deep feature extraction and classification of android malware images," *Sensors*, vol. 20, 2020.

[27] J. H. Go, T. Jan, M. Mohanty, O. P. Patel, D. Puthal, and M. Prasad, "Visualization approach for malware classification with resnext," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC)*, Glasgow, UK, July 2020.

[28] T. Vidas and N. Christin, "Evading android runtime analysis via sandbox detection," in *Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security*, Kyoto, Japan, June 2014.

[29] H. Gascon, F. Yamaguchi, D. Arp, and K. Rieck, "Structural detection of android malware using embedded call graphs," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, Berlin, Germany, November 2013.

[30] C. Yang, Z. Xu, G. Gu, V. Yegneswaran, and P. Porras, "Droidminer: automated mining and characterization of fine-grained malicious behaviors in android applications," in *Proceedings of the 19th European Symposium on Research in Computer Security*, Wroclaw, Poland, September 2014.

[31] P. Faruki, A. Bharmal, V. Laxmi et al., "Android security: a survey of issues, malware penetration, and defenses," *IEEE Communications Surveys & Tutorials*, vol. 17, 2015.

[32] K. Bakour, H. M. Ünver, and R. Ghanem, "The android malware detection systems between hope and reality," *SN Applied Sciences*, vol. 1, 2019.

[33] M. Farrokhmanesh and A. Hamzeh, "A novel method for malware detection using audio signal processing techniques,"

in *Proceedings of the Artificial Intelligence and Robotics (IRANOPEN)*, Qazvin, Iran, April 2016.

[34] G. Suarez-Tangil, J. E. Tapiador, P. Peris-Lopez, and J. Blasco, "Dendroid: a text mining approach to analyzing and classifying code structures in android malware families," *Expert Systems with Applications*, vol. 41, 2014.

[35] S. K. Dash, G. Suarez-Tangil, S. Khan et al., "Droidscribe: classifying android malware based on runtime behavior," in *Proceedings of the IEEE Security and Privacy Workshops (SPW)*, San Jose, CA, USA, May 2016.

[36] M. K. Alzaylaee, S. Y. Yerima, and S. Sezer, *Emulator vs Real Phone*, 2017.

[37] D. Su, J. Liu, X. Wang, and W. Wang, "Detecting android locker-ransomware on Chinese social networks," *IEEE Access*, vol. 7, 2019.

[38] F. Idrees, M. Rajarajan, M. Conti, T. M. Chen, and Y. Rahulamathavan, "Pindroid: a novel android malware detection system using ensemble learning methods," *Computers & Security*, vol. 68, 2017.

[39] B. Jung, T. Kim, and E. G. Im, "Malware classification using byte sequence information," in *Proceedings of the RACS '18: International Conference on Research in Adaptive and Convergent Systems*, Honolulu Hawaii, October 2018.

[40] S. Wu, P. Wang, X. Li, and Y. Zhang, "Effective detection of android malware based on the usage of data flow apis and machine learning," *Information and Software Technology*, vol. 75, 2016.

[41] S. Talukder and Z. Talukder, "A survey on malware detection and analysis tools," *International journal of Network Security & Its Applications*, vol. 12, 2020.

[42] M. A. Omer, S. R. M. Zeebaree, M. A. M. Sadeeq et al., "Efficiency of malware detection in android system: a survey," *Asian Journal of Computer Science & Information Technology*, vol. 7, 2021.

[43] R. S. Arslan, "Androanalyzer: android malicious software detection based on deep learning," *Peerj Computer Science*, vol. 7, 2021.

[44] D. Vasan, M. Alazab, S. Wassan, B. Safaei, and Q. Zheng, "Image-based malware classification using ensemble of cnn architectures (imcec)," *Compter and Security*, vol. 92, 2020.

[45] G. Suarez-Tangil and G. Stringhini, "Eight years of rider measurement in the android malware ecosystem: evolution and lessons learned," 2018, http://arxiv.org/abs/1801.08115.

[46] X. Zhang, Y. Zhang, M. Zhong et al., *Enhancing State-Of-The-Art Classifiers with Api Semantics to Detect Evolved Android Malware*, in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, Virtual Event, USA, November 2020.

[47] H. Cai, "Embracing mobile app evolution via continuous ecosystem mining and characterization," in *Proceedings of the EEE/ACM 7th International Conference on Mobile Software Engineering and Systems*, Seoul Republic of Korea, July 2020.

[48] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[49] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proceedings of the European conference on computer vision*, pp. 818–833, Springer, Zurich, Switzerland, September 2014.

[50] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, July 2017.

[51] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, Boston, MA, USA, June 2015.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.

[53] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size," 2016, https://arxiv.org/abs/1602.07360.

[54] Y. LeCun, "Lenet-5, convolutional neural networks," vol. 20, no. 5, 2015, http://yann.lecun.com/exdb/lenet.

[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[56] A. Y. Ng, "Feature selection, l1 vs. l2 regularization, and rotational invariance," in *Proceedings of the 21st international conference on Machine learning - ICML '04*, July 2004.

[57] M. S. ElSayed, N.-A. Le-Khac, M. A. Albahar, and A. Jurcut, "A novel hybrid model for intrusion detection systems in sdns based on cnn and a new regularization technique," *Journal of Network and Computer Applications*, vol. 191, Article ID 103160, 2021.

[58] F. M. Darus, N. A. A. Salleh, and A. F. M. Ariffin, "Android malware detection using machine learning on image patterns," in *Proceedings of the Cyber Resilience Conference (CRC)*, pp. 1-2, IEEE, Putrajaya, Malaysia, November 2018.

[59] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-An, and H. Ye, "Significant permission identification for machine-learning-based android malware detection," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3216–3225, 2018.

[60] X. Wang and C. Li, "Android malware detection through machine learning on kernel task structures," *Neurocomputing*, vol. 435, pp. 126–150, 2021.

[61] T. Chen, Q. Mao, Y. Yang, M. Lv, and J. Zhu, "Tinydroid: a lightweight and efficient model for android malware detection and classification," *Mobile Information Systems*, vol. 2018, Article ID 4157156, 9 pages, 2018.

[62] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, Article ID 21954, 2017.

[63] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5g network slicing using sdn and nfv: a survey of taxonomy, architectures and future challenges," *Computer Networks*, vol. 167, Article ID 106984, 2020.

[64] T. Huang and H.-Y. Kao, "R2-d2: color-inspired convolutional neural network (cnn)-based android malware detections," in *Proceedings of the IEEE International Conference on Big Data (Big Data)*, pp. 2633–2642, IEEE, 2018.

[65] W. Hardy, L. Chen, S. Hou, Y. Ye, and X. Li, "Dl4md: a deep learning framework for intelligent malware detection," in *Proceedings of the International Conference on Data Science (ICDATA), The Steering Committee of The World Congress in Computer Science*, 2016.

[66] T. Kim, B. Kang, M. Rho, S. Sezer, and E. G. Im, "A multimodal deep learning method for android malware detection using various features," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 3, pp. 773–788, 2019.

[67] E. B. Karbab, M. Debbabi, A. Derhab, and D. Mouheb, "MalDozer: automatic framework for android malware detection using deep learning," *Digital Investigation*, vol. 24, pp. S48–S59, 2018.

[68] R. Nix and J. Zhang, "Classification of android apps and malware using deep neural networks," in *Proceedings of the International joint conference on neural networks (IJCNN)*, pp. 1871–1878, IEEE, Anchorage, AK, USA, May 2017.

[69] H. Cai, N. Meng, B. Ryder, and D. Yao, "DroidCat: effective android malware detection and categorization via app-level profiling," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1455–1470, 2019.

[70] Y. Zhou and X. Jiang, "Dissecting android malware: characterization and evolution," in *Proceedings of the IEEE Symposium on Security and Privacy*, May 2012.

[71] M. Ficco, "Malware analysis by combining multiple detectors and observation windows," *IEEE Transactions on Computers*, vol. 71, p. 1, 2021.

[72] A. Naway and Y. Li, "A review on the use of deep learning in android malware detection," 2018, https://arxiv.org/abs/1812.10360.

[73] W. Wang, M. Zhao, Z. Gao et al., "Constructing features for detecting android malicious applications: issues, taxonomy and directions," *IEEE Access*, vol. 7, 2019.

[74] O. Aslan and R. Samet, "A comprehensive review on malware detection approaches," *IEEE Access*, vol. 8, 2020.

[75] S. Hitawala, "Evaluating resnext model architecture for image classification," 2018, http://arxiv.org/abs/1805.08700.

[76] M. A. Albahar, "Recurrent neural network model based on a new regularization technique for real-time intrusion detection in SDN environments," *Security and Communication Networks*, vol. 2019, Article ID 8939041, 9 pages, 2019.

[77] M. A. Albahar, "Skin lesion classification using convolutional neural network with novel regularizer," *IEEE Access*, vol. 7, Article ID 38306, 2019.

[78] M. A. Albahar, M. Binsawad, and L. Maglaras, "Deep autoencoders and feedforward networks based on a new regularization for anomaly detection," *Security and Communication Networks*, vol. 2020, Article ID 7086367, 9 pages, 2020.

[79] M. A. Albahar, A. A. Albahr, and M. H. Binsawad, "An efficient person re-identification model based on new regularization technique," *IEEE Access*, vol. 8, Article ID 171049, 2020.

[80] A. Albahr, M. Albahar, M. Thanoon, M. Binsawad, and M. Versaci, "Computational learning model for prediction of heart disease using machine learning based on a new regularizer," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 8628335, 10 pages, 2021.

[81] H. Cai, "Assessing and improving malware detection sustainability through app evolution studies," *ACM Transactions on Software Engineering and Methodology*, vol. 29, 2020.

[82] K. Xu, Y. Li, R. Deng, K. Chen, and J. Xu, "Droidevolver: self-evolving android malware detection system," in *Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P)*, Stockholm, Sweden, Jun 2019.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 5314671, 9 pages
https://doi.org/10.1155/2022/5314671

Research Article

Leakage Prediction in Machine Learning Models When Using Data from Sports Wearable Sensors

Qizheng Dong

Zhengzhou University of Science and Technology, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Qizheng Dong; dongqizheng1982@126.com

Received 1 April 2022; Revised 19 April 2022; Accepted 25 April 2022; Published 17 May 2022

Academic Editor: Konstantinos Demertzis

One of the major problems in machine learning is data leakage, which can be directly related to adversarial type attacks, raising serious concerns about the validity and reliability of artificial intelligence. Data leakage occurs when the independent variables used to teach the machine learning algorithm include either the dependent variable itself or a variable that contains clear information that the model is trying to predict. This data leakage results in unreliable and poor predictive results after the development and use of the model. It prevents the model from generalizing, which is required in a machine learning problem and thus causes false assumptions about its performance. To have a solid and generalized forecasting model, which will be able to produce remarkable forecasting results, we must pay great attention to detecting and preventing data leakage. This study presents an innovative system of leakage prediction in machine learning models, which is based on Bayesian inference to produce a thorough approach to calculating the reverse probability of unseen variables in order to make statistical conclusions about the relevant correlated variables and to calculate accordingly a lower limit on the marginal likelihood of the observed variables being derived from some coupling method. The main notion is that a higher marginal probability for a set of variables suggests a better fit of the data and thus a greater likelihood of a data leak in the model. The methodology is evaluated in a specialized dataset derived from sports wearable sensors.

1. Introduction

Machine learning models typically receive input data and solve problems such as pattern recognition by applying a sequence of particular transformations. The majority of these transformations turn out to be extremely sensitive to modest changes in input. Under specific scenarios, using this sensitivity can result in a difference in the behavior of the learning algorithm [1, 2]. Adversarial attack is the design of an adequate input in a specific way that leads the learning algorithm to erroneous outputs while not easily noticed by human observers. It is a severe concern in the reliability and security of artificial intelligence technologies. The issue arises because learning techniques are intended for use in stable situations where training and test data are generated from the same, possibly unknown distribution [3]. A trained neural network, for example, represents a significant decision limit corresponding to a standard class. Of course, the restriction is not without flaws. A correctly designed and implemented attack, which corresponds to a modified input form a slightly differentiated dataset, can cause the algorithm to make an incorrect judgment (wrong class) [4–6].

Developing and selecting machine learning methodologies to solve complex, usually nonlinear, problems is inextricably linked to the area of application and the target problem it seeks to solve. This is one of the essential processes of preprocessing the area of interest and the dataset, as the choice of appropriate algorithms depends on not only the nature and dynamics of the problem but also the characteristics of the available data, such as volume, number, and type of variables in question. The preprocessing of the data concerns the tests and the preparation work that should be carried out in the examined dataset before the use and application of machine learning algorithms. This method is critical because if the quality of usage or training data is not

ensured, the algorithms' performance will be subpar or the algorithms may produce false results [6, 7].

In general, data preparation/preprocessing entails dealing with scenarios when the original data have issues such as contradicting information, coding discrepancies, field terminology, and units of measurement. However, more critical issues such as the presence of lost values, noise, and extreme values and dealing with special requirements that necessitate data transformation, such as discretization, normalization, dimension reduction, or the selection of the most appropriate features, must be addressed [9–11]. It should be noted that several techniques can be used in preprocessing processes, with the choice of the best strategy arising from the nature of the field of knowledge, the problem to be addressed, the available data, and the machine learning algorithm used.

One of the most critical errors that occur during the preprocessing of data for use by machine learning algorithms is data leakage. The leak in question refers to cases where, inadvertently or even intentionally, the value that the model wishes to predict (dependent variable) is contained indirectly or directly in the features that are called to train the algorithm (independent variables). Any variable that provides transparent information about the value that the model is trying to predict is considered a data leak and leads to fictitious results. An obvious solution to this problem is to apply preprocessing only to the training set. Using preprocessing techniques to the whole dataset will make the model learn the training and the test sets, resulting in a data leak, and thus the model fails to generalize [2, 12, 13].

The major problem of data leakage occurs when there is a severe indirect interaction of features which is not easy to detect. It is, for example, a widespread phenomenon in machine learning experiments; the relationship between the dependent and the independent variable is complex (e.g., polynomial, trigonometric, and so on), so new features may be created that seem to help capture this relationship. Still, in practice, they create serious data leaks [14, 15].

Similarly, combinations may exist between independent and dependent variables through, for example, an arithmetic operation, a modification, or a conversion to make them more important in explaining the discrepancies in the data than if they remained separate. Creating a new opportunity through the interaction of existing features creates data leaks and significant bias in the final machine learning model [4, 7, 11].

For example, Lu et al. [15] developed a weighted context graph model (WCGM) for information leakage, with the critical goals of first increasing the contextual relevance of information, second classifying the tested data based on the commonality characteristics of its context graphs, and third preserving data proprietors' privacy. The weighted context network reduces complexity by using key sensitive phrases as nodes and contextual linkages as edges. The proposed maximum subgraph matching approach and deep learning algorithms are used to evaluate the similarity of the tested information and the pattern, as well as the responsiveness of the tested data to match the converted data better. The proposed model surpassed the competition regarding accuracy, recall, and run time, indicating its ability to detect real-time data leaks.

Using a variety of datasets, Salem et al. [14] provided research on the new and developing danger of membership inference attacks, demonstrating the efficacy of the suggested assaults across sectors. They offer two defensive strategies to alleviate the problem. The first, known as dropout, involves randomly deleting specific nodes in each fully linked neural system training step. In contrast, the second, known as model stacking, involves organizing numerous ML models in a ranked order [16]. Extensive testing has shown that our defensive strategies may significantly lower the performance of a membership inference attempt while retaining a high degree of usefulness, i.e., good target model prediction accuracy. They also suggest a defensive mechanism against a larger class of inclusion inference assaults while maintaining the ML model's high usefulness.

In this work, we proposed an innovative system of leakage prediction in machine learning models, which calculates a lower limit for the marginal probability of the observed variables coming from a coupling method, which shows that in an examined machine learning model, there is data leakage. The methodology is implemented based on the Bayesian inference methodology [17–19]. The model's goal is to generate an analytical approach to the reverse probability of unobserved variables [20, 21], to draw statistical inferences about the important correlated variables, and to compute a lower limit for the marginal likelihood of observable variables generated from a coupling method. The highest probability indicates that there is a data leak [22]. This is done to have a solid and generalized forecasting model, which will produce remarkable forecasting results without data leakages.

2. Proposed Approach

The proposed implementation is based on Bayesian inference [23–25], which is a method of approaching intractable problems that arise in highly fuzzy environments. More specifically, the methodology offers a secure solution for the observed variables and unknown parameters and latent states of variables, characterized by different types of relationships (interconnected, transformed, hidden, random, and so on). A prior distribution, a posterior distribution, and a likelihood function are used to illustrate Bayesian inference [26] in Figure 1.

The prediction error is defined as the difference between the previous expectation and the likelihood function's peak (i.e., reality). The variance of the prior is the source of uncertainty. The variance of the likelihood function is referred to as noise [27].

Parameters and latent variables are grouped as "unobserved variables." So, with the proposed method, the purpose is as follows [28–31]:

(1) In order to generate an analytical approach to the reverse probability of unobserved variables, develop statistical findings for the important correlated variables.

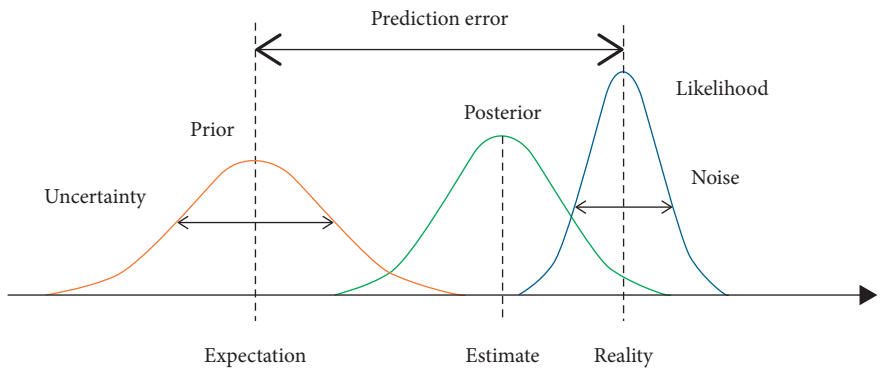


Figure 1: Bayesian inference.

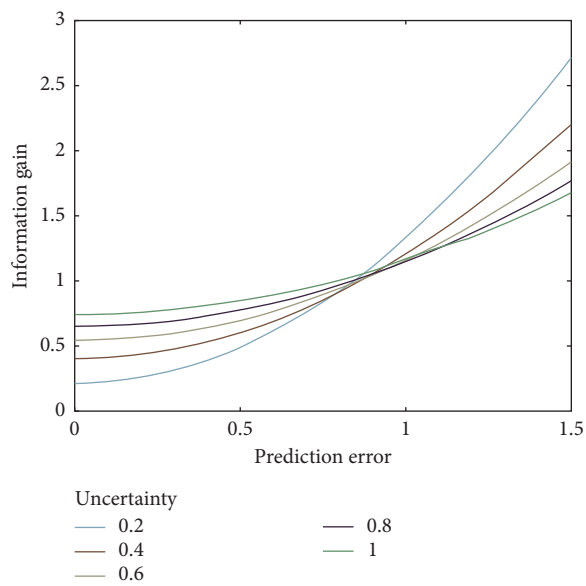


Figure 2: Information gain vs prediction error.

(2) The marginal likelihood of the data presented in the model can be used to derive a lower limit for the marginal probability of the observed data, with the marginalization conducted on unobserved variables. The main notion is that a higher marginal probability for a set of variables suggests a better fit of the data and thus a greater likelihood of a data leak in the model.

An example of information gain vs prediction error is presented in Figure 2.

Information gain is calculated mathematically as a function of prediction errors for uncertainty levels ranging from 0.2 to 1.0. The external noise level is set to 0.1 [23, 27].

The method generally approaches a conditional latent variable density given the observed variables where we assume that a mixture is present. Mixing behavior occurs because the source of each observation is unknown, that is, the classification into a specific, exact domain of a variable [32]. Thus, each observation xi is predetermined to each of $f_i(\cdot \mid \theta_i)$ with probability pi. Depending on the case, the purpose of the inference is to reconstruct the classification of observations into definition fields, construct estimators for the components' parameters, or even estimate the number of components themselves [15]. It is always feasible to map a mixture of k form distributions to a random variable Xi via a delimitation method [25, 33]:

$$\sum_{I=1}^{K} p_i f_i(x|\theta_i). \tag{1}$$

The random variable Zi with $\{1, 2, \ldots, k\}$, is as follows [34]:

$$X_i|Z_i = z \sim f(x|\theta_z)\mu\varepsilon Z_i \sim M_k(1; p_1, \ldots, p_k). \tag{2}$$

Next, we assume that we have observed the extended data, which consist of independent pairs with distribution [35]:

$$P(Z_i = j|X_i = x) = \frac{p_j f_j(x)}{\sum_{I=1}^{K} p_i f_i(x)} \propto p_j f_j(x). \tag{3}$$

In the particular case of the model:

$$pN(\mu_1, 1) + (1-p)N(\mu_2, 1), \tag{4}$$

where we consider the same normal a priori distribution in the media, $\mu_1, \mu_2 \sim N(0, 10)$, we will calculate the ex post weight $\omega(z)$ for a classification z, where in the first component are l observations [24, 36]:

$$\sum_{I=1}^{N} I_{\{z_i=1\}} = l \text{ for } (n_1, n_2) = (l, n-l). \tag{5}$$

So, we have [37]

$$\pi(\underline{z}, \mu_1, \mu_2 | \underline{x}, n_1, n_2) \propto \exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\left[I_{\{z_i=1\}}(x_i-\mu_1)^2 + \left(1-I_{\{z_i=1\}}\right)(x_i-\mu_2)^2\right] - \frac{\mu_1^2}{20} - \frac{\mu_2^2}{20}\right\} \times \frac{1}{(2\pi)^{n/2}} p^{\sum_{i=1}^{n} I_{\{z_i-1\}}} (1-p)^{n-\sum_{i=1}^{n} I_{\{z_i-1\}}}. \tag{6}$$

The ex-weight $\omega(z)$ is obtained by completing the above function in RxR for $\mu 1$ and $\mu 2$, which is a double integral which is easily calculated. For the completion in terms of $\mu 1$, excluding the parts that do not contain it, it is enough to calculate [24, 33, 36, 38]

$$I_1 = \int_{-\infty}^{+\infty} \exp\left\{-\frac{1}{2}\sum_{i=1}^{n} I_{z_i=1}(x_i-\mu_1)^2 - \frac{\mu_1^2}{20}\right\} d\mu_1. \tag{7}$$

But

$$\begin{aligned}
&\exp\left\{-\frac{1}{2}\sum_{i=1}^{n} I_{\{z_i=1\}}(x_i-\mu_1)^2 - \frac{\mu_1^2}{20}\right\} \\
&= \exp\left\{-\frac{1}{2}\mu_1^2\left(\sum_{i=1}^{n} I_{\{z_i=1\}} + \frac{1}{10}\right) - \frac{1}{2}\sum_{i=1}^{n} x_i^2 I_{\{z_i=1\}} + \mu_1 \sum_{i=1}^{n} x_i I_{\{z_i=1\}}\right\} \\
&= \exp\left\{-\frac{1}{2}\left(\sum_{i=1}^{n} I_{\{z_i=1\}} + \frac{1}{10}\right)\left(\mu_1^2 + \frac{\sum_{i=1}^{n} x_i^2 I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10} - 2\mu_1 \frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10}\right)\right\} \\
&= \exp\left\{\frac{1}{2}\left(\sum_{i=1}^{n} I_{\{z_i=1\}} + \frac{1}{10}\right)\left(\frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10}\right)^2\right\} \\
&\times \exp\left\{-\frac{1}{2}\left(\sum_{i=1}^{n} I_{\{z_i=1\}} + \frac{1}{10}\right)\left(\mu_1^2 + \frac{\sum_{i=1}^{n} x_i^2 I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10} - 2\mu_1 \frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10} + \left(\frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10}\right)^2\right)\right\} \\
&= c_1 \exp\left\{-\frac{1}{2}\left(\sum_{i=1}^{n} I_{izz}, = 1 + \frac{1}{10}\right)\left(\mu_1^2 - 2\mu_1 \frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10} + \left(\frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10}\right)^2\right)\right\} \\
&= c_1 \exp\left\{-\frac{1}{2}\left(\sum_{i=1}^{n} I_{\{z_i=1\}} + \frac{1}{10}\right)\left(\mu_1 - \frac{\sum_{i=1}^{n} x_i I_{\{z_i=1\}}}{\sum_{i=1}^{n} I_{\{z_i=1\}} + 1/10}\right)^2\right\}, \\
&\text{where } c_1 = \exp\left\{-\frac{1}{2}\sum_{i=1}^{n} x_i^2 I_{\{z_i=1\}} + \frac{\left(\sum_{i=1}^{n} x_i I_{\{z_i=1\}}\right)^2}{2(l+1/10)}\right\},
\end{aligned} \tag{8}$$

So, to calculate the integral, we have

$$I_1 = c_1 \int_{-\infty}^{+\infty} \exp\left\{ -\frac{1}{2}\left(\sum_{i=1}^{n} I_{\{z_t=1\}} + \frac{1}{10} \right)\left(\mu_1 - \frac{\sum_{i=1}^{n} x_i I_{\{z_t=1\}}}{\sum_{i=1}^{n} I_{\{z_t=\}1} + 1/10} \right)^2 \right\} d\mu_1 \Rightarrow$$
$$I_1 = c_1 \frac{\sqrt{2\pi}}{\sqrt{\sum_{i=1}^{n} I_{\{z_t=1\}} + 1/10}} = c_1 \frac{\sqrt{2\pi}}{\sqrt{l + 1/10}}. \tag{9}$$

because the last integral is crucial in the full support of the exponential distribution [39]:

$$N\left(\frac{\sum_{i=1}^{n} x_i I_{\{z_t=1\}}}{\sum_{i=1}^{n} I_{\{z_t=1\}} + 1/10}, \frac{1}{\sum_{i=1}^{n} I_{\{z_t=1\}} + 1/10} \right). \tag{10}$$

For the completion in terms of $\mu 2$, excluding the parts that do not contain it, it is enough to calculate [23, 36, 38, 40]

$$I_2 = \int_{-\infty}^{+\infty} \exp\left\{ -\frac{1}{2} \sum_{i=1}^{n} \left(1 - I_{\{z_t=1\}} \right) (x_i - \mu_2)^2 - \frac{\mu_2^2}{20} \right\} \mathrm{d}\mu_2. \tag{11}$$

Following the same methodology as before, we conclude that [41]

$$I_1 = c_2 \frac{\sqrt{2\pi}}{\sqrt{n - l + 1/10}}$$
$$\text{where } c_2 = \exp\left\{ -\frac{1}{2} \sum_{i=1}^{n} x_i^2 \left(1 - I_{\{z_t=1\}} \right) + \frac{\left(\sum_{i=1}^{n} x_i \left(1 - I_{\{z_i=1\}} \right) \right)^2}{2(n - l + 1/10)} \right\}. \tag{12}$$

So, the ex post probability $\omega(z)$ is calculated as follows [21, 23, 42, 43]:

$$\omega(\underline{z}) = c_1 \frac{\sqrt{2\pi}}{\sqrt{l + 1/10} c_2 \sqrt{2\pi} / \sqrt{n - l + 1/10}} p^{\sum_{i=1}^{n} I_{[z-1]} - 1} (1 - p)^{n - \sum_{i=1}^{n} I_{zz} - 1\}}$$
$$= c_1 c_2 \frac{2\pi}{\sqrt{(l + 1/10)(n - l + 1/10)}} p^l (1 - p)^{n-l}. \tag{13}$$

If we replace $c1$, $c2$, we take the relation:

$$\omega(\underline{z}) = \frac{\sqrt{2\pi}}{\sqrt{(l + 1/10)}\sqrt{(n - l + 1/10)}} \times \exp\left\{ -\frac{1}{2}\left(\sum_{i=1}^{n} x_i^2 - \frac{\left(\sum_{i=1}^{n} x_i I_{\{z_t=1\}} \right)^2}{l + 1/10} - \frac{\left(\sum_{i=1}^{n} x_i \left(1 - I_{\{z_t=1\}} \right) \right)^2}{n - l + 1/10)} \right) \right\} p^l (1 - p)^{n-l}. \tag{14}$$

Thus, from the above analysis, it appears that it is practically possible to arrive at detailed expressions of the maximum probability and Bayes estimators [44] for the ex ante distributions of the variables of interest and thus marginalize the set of variables for models where there is a data leak [28, 33].

3. Experiments and Results

A specialized scenario was implemented to model the proposed system that uses sports wearables data to record the movements of athletes playing beach volleyball. The dataset comprises three-dimensional acceleration data from joint actions of beach volleyball athletes, each of whom was fitted with an accelerometer worn on the wrist and sampled at 39 Hz. The signal was recorded at 14 bits per axis and then compressed to 16 g. The x, y, and z axes relate to the athletes' spatial arrangement, which is recorded in an independent coordinate system based on the sensor configuration, as there was no transfer to real-world coordinates [45, 46]. The 30 athletes recorded ranged in expertise from novice to professional

volleyball players. The set's goal is to create an identification and classification system that extracts relevant portions from continuous input and classifies them [47]. The categorization includes ten various volleyball activities, such as homemade service, block, nail, and so on. For the evaluation of the system, 10 characteristics were selected, which were randomly combined into pairs to identify the observed variables, whether they come from a coupling method and whether there is a data leak.

We first describe some key features. Let $g(\cdot,\cdot|\theta)$ be the joint density function of (X, Z) given by the parametric vector θ, $f(\cdot|\theta)$ be the density function of X given θ, and $k(\cdot|x,\theta)$ be the function density of the bounded distribution of Z given by observations x and θ. The algorithm is based on the use of incomplete data, i.e., we can write the distribution of sample x as follows [1, 2, 40]:

$$\begin{aligned} f(x|\theta) &= \int g(\underline{x},\underline{z}|\theta)\mathrm{d}\underline{z} \\ &= \int f(x|\theta)k(\underline{z}\,|\,\underline{x},\theta)\mathrm{d}\underline{z}. \end{aligned} \tag{15}$$

So, logarithm it:

$$g(x,z|\theta) = f(x|\theta)k(\underline{z}\,|\,\underline{x},\theta). \tag{16}$$

We arrive at a complete (unobserved) logarithm of probability:

$$L^c(\theta|\,\underline{x},\underline{z}) = L(\theta|\,\underline{x}) + \log k(\underline{z}\,|\,\underline{x},\theta), \tag{17}$$

where L is the observed logarithm of the probability. The algorithm fills in the missing variables z based on $k(z|x, \theta)$ and then maximizes with θ the expected full logarithm probability [21, 25, 48].

So, the algorithm is configured as follows:

(1)Give some initial values to $\theta(0)$.

(2)For each t, $t = 1, 2, \ldots, n$, calculate $Q(\theta|\theta^{(t-1)}, \underline{x}) = E_{\theta^{(t-1)}}(L^c(\theta|\,\underline{x},\underline{Z}))$ where $\underline{Z} \sim k(z|x,\theta)$.

(3)Maximize concerning θ the $Q(\theta|\theta^{(t-1)}, \underline{x})$ and set $\theta^{(t)} = \arg\max_{\theta} Q(\theta|\theta^{(t-1)}, \underline{x})$.

When performing the above algorithm, the result is that in each iteration, the (observed) $L(\theta|x)$ increases.

As an application of the above, we consider the particular case of the model of mixing two regular variables, where all parameters are known except $\theta = (\mu1, \mu2)$. For a simulated sample of 500 observations and actual values $p = 0.7$ and $(\mu1, \mu2) = (0, 2.5)$, the logarithm of probability has two peaks. Applying the algorithm to this model, we have that the total probability is [20, 49, 50]

$$\begin{aligned} &p^{\sum_{i=1}^{n} I_{izi-1}}(1-p)^{n-\sum_{i=1}^{n} Iz_i-1\}}(2\pi)^{-n/2} \\ &\cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\left[I_{\{z_i=1\}}(x_i-\mu_1)^2+\left(1-I_{\{z_i=1\}}\right)(x_i-\mu_2)^2\right]\right\}, \end{aligned} \tag{18}$$

where its logarithm is

$$\begin{aligned} L^c(\theta\underline{x},\underline{z}) &= \sum_{i=1}^{n} I_{\{z_i=1\}}\log p+\left(n-\sum_{i=1}^{n} I_{\{z_i=1\}}\right)\log(1-p) \\ &\quad -\frac{n}{2}\log(2\pi) \\ &\quad -\frac{1}{2}\sum_{i=1}^{n}\left[I_{\{z_i=1\}}(x_i-\mu_1)^2+\left(1-I_{\{z_i=1\}}\right)(x_i-\mu_2)^2\right]. \end{aligned} \tag{19}$$

For the first step, we need to calculate

$$Q\left(\theta|\theta^{(t-1)},\underline{x}\right) = E_{\theta^{(t-1)}}(\log L^c(\theta|\,\underline{x},\underline{Z})), \tag{20}$$

where the mean value is taken for $\underline{Z} \sim k(z|x,\theta)$, and we have that Zi are independent of [51–54]

$$\begin{aligned} &P(Z_i = 1|\,\underline{\theta},\underline{x}) \\ &= \frac{p\exp\left\{-(x_i-\mu_1)^2/2\right\}}{p\exp\left\{-(x_i-\mu_1)^2/2\right\}+(1-p)\exp\left\{-(x_i-\mu_2)^2/2\right\}} \\ &= 1 - P(Z_i = 2\mid \underline{\theta},\underline{x}). \end{aligned} \tag{21}$$

In step t, the expected rankings are equal to

$$\begin{aligned} \hat{z}_i^{(t-1)} &= E\left(\sum_{i=1}^{n} I_{\{z_i=1\}}|\overset{(t-1)}{\underline{\theta}},\underline{x}\right) \\ &= P\left(Z_i = 1|\overset{(t-1)}{\underline{\theta}},\underline{x}\right). \end{aligned} \tag{22}$$

Therefore:

$$\begin{aligned} Q\left(\theta|\theta^{(t-1)},\underline{x}\right) &= \sum_{i=1}^{n}\hat{z}_i^{(t-1)}\log p+\left(n-\sum_{i=1}^{n}\hat{z}_i^{(t-1)}\right)\log(1-p) \\ &\quad -\frac{n}{2}\log(2\pi) \\ &\quad -\frac{1}{2}\sum_{i=1}^{n}\left[\hat{z}_i^{(t-1)}(x_i-\mu_1)^2+\left(1-\hat{z}_i^{(t-1)}\right)(x_i-\mu_2)^2\right]. \end{aligned} \tag{23}$$

which we maximize in the second step in terms of $(\mu1, \mu2)$ and get

$$\begin{aligned} \mu_1^{(t)} &= \frac{\sum_{i=1}^{n}\hat{z}_i^{(t-1)}x_i}{\sum_{i=1}^{n}\hat{z}_i^{(t-1)}}, \\ \mu_2^{(t)} &= \frac{\sum_{i=1}^{n}1-\hat{z}_i^{(t-1)}x_i}{\sum_{i=1}^{n}1-\hat{z}_i^{(t-1)}}. \end{aligned} \tag{24}$$

This example involved running the algorithm 20 times (each time with 100 repeats) while picking random numbers from a range of possibilities for the initial conditions. However, the proposed approach was only drawn to the highest and principal vertex of the logarithm

probability eight times out of every 20 times in the experiments. It was drawn to the pseudo-vertex of the logarithm probability distribution for the remaining 12 times (although the likelihood is much lower). The original values were closer to the lower peak than the final values, indicating that the early values were more accurate. The algorithm converges to the pseudo-peak of likelihood, at which point we may make 84 percent correct predictions about the coupling between the variables in the dataset. Accordingly, we will have 93 percent of the variables accurately predicted to couple their coefficients if the algorithm converges to the dominant peak in probability.

4. Discussion and Conclusions

In this work, we proposed an innovative system of leakage prediction in machine learning models, which is based on Bayesian inference, to calculate a lower limit for the marginal probability of the observed variables coming from a coupling method, which shows that in an examined machine learning model, there is data leakage. The methodology is evaluated in a specialized dataset from sports wearable sensors, where the ability of the method to detect variable coupling is demonstrated, even when it is done randomly.

The proposed methodology is a Bayesian approach to statistical discoveries in complicated distributions that are difficult to evaluate directly or by sampling, and this is the methodology that has been offered. It is a method of selection that is different from Monte Carlo sampling methods. While Monte Carlo techniques use a sequence of samples to approximate a rear distribution numerically, the proposed algorithm provides a locally optimal, correct analytical solution, allowing even hidden variable coupling to be found. From the maximum ex post estimate of each variable's unique most probable value to the fully Bayesian estimation that calculates (approximately) the entire rear distribution of parameters and latent variables, the algorithm finds a set of optimal parameters of the interrelated variables, which can then be solved in detail using the information obtained from the data. Indeed, this is true even for conceptually comparable variables, such as a basic nonhierarchical model with only two parameters and no latent variables.

The extension of the methodology can focus on integrating countervailing machine learning techniques to be a complete defense system in case of attacks that attempt to deceive the models by providing misleading information. Determine strategies and procedures for running the model on specified sets of issues with training and test data generated from the same statistical distribution. Moreover, a future expansion of the proposed system will review the taxonomies of the characteristics of transfer learning, particularly whether and how this system can mitigate them. Finally, learning transfer approaches are investigated from known distribution attack methods seeking to exploit the dynamics of categorization decision-making limits.

Data Availability

The data used in this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] K. M. R. Alam, N. Siddique, and H. Adeli, "A dynamic ensemble learning algorithm for neural networks," *Neural Computing & Applications*, vol. 32, no. 12, pp. 8675–8690, 2020.

[2] J. Gawlikowski, "A Survey of Uncertainty in Deep Neural Networks," 2021, http://arxiv.org/abs/2107.03342.

[3] K. Demertzis, L. Iliadis, and P. Kikiras, "A Lipschitz - Shapley Explainable Defense Methodology against Adversarial Attacks," in *Proceedings of the Artificial Intelligence Applications and Innovations. AIAI 2021 IFIP WG 12.5*, pp. 211–227, Crete, Greece, June, 2021.

[4] R. Chauhan and S. Shah Heydari, "Polymorphic Adversarial DDoS attack on IDS using GAN," in *Proceedings of the 2020 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6, Shenzhen, China, July. 2020.

[5] Q. Liu, J. Guo, C.-K. Wen, and S. Jin, "Adversarial attack on DL-based massive MIMO CSI feedback," *Journal of Communications and Networks*, vol. 22, no. 3, pp. 230–235, 2020.

[6] P. Yu, K. Song, and J. Lu, "Generating adversarial examples with conditional generative adversarial net," in *Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 676–681, Beijing, China, August. 2018.

[7] Z.-A. Zhu, Y.-Z. Lu, and C.-K. Chiang, "Generating adversarial examples by makeup attacks on face recognition," in *Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP)*, pp. 2516–2520, Taipei, Taiwan, September. 2019.

[8] J. Yu, Y. Lee, K. C. Yow, M. Jeon, and W. Pedrycz, "Abnormal event detection and localization via adversarial event prediction," *IEEE Transactions on Neural Networks and Learning Systems*, no. –15, pp. 1–15, 2021.

[9] Z. Shi, Y. Ma, and X. Yu, "An effective and efficient method for word-level textual adversarial attack," in *Proceedings of the 2021 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1–6, Athens, Greece, September. 2021.

[10] P. Tang, W. Wang, J. Lou, and L. Xiong, "Generating adversarial examples with distance constrained adversarial imitation networks," *IEEE Transactions on Dependable and Secure Computing*, p. 1, 2021.

[11] B. Tarchoun, I. Alouani, A. Ben Khalifa, and M. A. Mahjoub, "Adversarial attacks in a multi-view setting: an empirical study of the adversarial patches inter-view transferability," in *Proceedings of the 2021 International Conference on Cyberworlds (CW)*, pp. 299–302, Caen, France, September. 2021.

[12] P. Gattineni and G. S. Dharan, "Intrusion Detection Mechanisms: SVM, random forest, and extreme learning machine (ELM)," in *Proceedings of the 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 273–276, Coimbatore, India, September. 2021.

[13] P. Rathore, A. Basak, S. H. Nistala, and V. Runkana, "Untargeted, targeted and universal adversarial attacks and defenses on time series," in *Proceedings of the 2020*

International Joint Conference on Neural Networks (IJCNN), pp. 1–8, Glasgow, UK, July. 2020.

[14] A. Salem, Y. Zhang, M. Humbert et al., "Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models," 2018, http://arxiv.org/abs/1806.01246.

[15] Y. Lu, X. Huang, Y. Ma, and M. Ma, "A weighted context graph model for fast data leak detection," in *Proceedings of the 2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, Kansas City, MO, USA, May, 2018.

[16] M. Miyatake, H. Sawai, Y. Minami, and K. Shikano, "Integrated training for spotting Japanese phonemes using large phonemic time-delay neural networks," *International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 449–452, 1990.

[17] J. O. Berger, "Bayesian analysis, Springer Series in Statistics," in *Statistical Decision Theory and Bayesian Analysis*, pp. 118–307, Springer, New York, NY, USA, 1985.

[18] J. O. Berger, "Basic concepts," in *Statistical Decision Theory and Bayesian Analysis*, pp. 1–45, Springer, New York, NY, USA, 1985.

[19] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2014, http://arxiv.org/abs/1312.6114.

[20] A. J. M. Garrett, "Review: probability theory: the logic of science," *Probability and Risk*, vol. 3, no. 3-4, pp. 243–246, 2004.

[21] L. E. B. Salasar, J. G. Leite, and F. Louzada, "Likelihood-based inference for population size in a capture-recapture experiment with varying probabilities from occasion to occasion," *Brazilian Journal of Probability and Statistics*, vol. 30, no. 1, pp. 47–69, 2016.

[22] J. Lü and P. Wang, "Modeling and analysis of large-scale networks," in *Modeling and Analysis of Bio-Molecular Networks*, pp. 249–292, Springer, Singapore, 2020.

[23] Y. Emma Wang, Y. Zhu, G. G. Ko, B. Reagen, G.-Y. Wei, and D. Brooks, "Demystifying bayesian inference workloads," in *Proceedings of the 2019 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, pp. 177–189, Madison, WI, USA, March, 2019.

[24] S. Jun, "Bayesian Inference and Learning for Neural Networks and Deep Learning," in *Proceedings of the 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 569–571, Seoul, Korea, Octobar. 2020.

[25] Z. Rudong, S. Xianming, W. Qian, S. Xiaobo, and S. Xing, "Bayesian inference for ammunition demand based on Gompertz distribution," *Journal of Systems Engineering and Electronics*, vol. 31, no. 3, pp. 567–577, 2020.

[26] Handbook of Statistics, "Bayesian Thinking, Modeling and Computation - PDF Free Download," 2022, https://epdf.tips/handbook-of-statistics-volume-25-bayesian-thinking-modeling-and-computation.html.

[27] H. Yanagisawa, O. Kawamata, and K. Ueda, "Modeling emotions associated with novelty at variable uncertainty levels: a bayesian approach," vol. 13, 2019 https://www.frontiersin.org/article/10.3389/fncom.2019.00002.

[28] X.-d. Zhang, "An improved bayesian network inference algorithm," in *Proceedings of the 2010 Third International Conference on Intelligent Networks and Intelligent Systems*, pp. 389–392, Shenyang, China, August. 2010.

[29] J. Yun-Jie, C. Wen-Qi, and H. Ling, "Risk identification and simulation based on the bayesian inference," in *Proceedings of the 2018 4th Annual International Conference on Network and Information Systems for Computers (ICNISC)*, pp. 407–411, Wuhan, China, April. 2018.

[30] D. Hou, T. Driessen, and H. Sun, "The Shapley value and the nucleolus of service cost savings games as an application of 1-convexity," *IMA Journal of Applied Mathematics*, vol. 80, no. 6, pp. 1799–1807, 2015.

[31] G. Alessandrini, M. V. D. Hoop, R. Gaburro, and E. Sincich, "Lipschitz stability for a piecewise linear Schrödinger potential from local Cauchy data," *Asymptot. Anal*, vol. 108, no. 3, pp. 115–149, 2018.

[32] "Permutation principles for the change analysis of stochastic processes under strong invariance," 2022, https://dl.acm.org/doi/abs/10.5555/1124448.1716910.

[33] J. Barbier, "Overlap matrix concentration in optimal Bayesian inference," *Information and Inference: A Journal of the IMA*, vol. 10, no. 2, pp. 597–623, 2020.

[34] O. Lee, "Probabilistic properties of a nonlinear ARMA process with markov switching," *Communications in Statistics - Theory and Methods*, vol. 34, no. 1, pp. 193–204, 2005.

[35] Y. Lu, X. Huang, D. Li, and Y. Zhang, "Collaborative graph-based mechanism for distributed big data leakage prevention," in *Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, Abu Dhabi, UAE, September. 2018.

[36] M. T. Koudahl and B. de Vries, "Batman: bayesian target modelling for active inference," in *Proceedings of the 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3852–3856, Barcelona, Spain, Febrauary. 2020.

[37] D. K. Dussmann, "Computational Systems Biology," 2022, https://www.kulturkaufhaus.de/en/detail/ISBN-2244012260139/Lecca-Paola/Computational-Systems-Biology.

[38] E.-H. Choi, T. Fujiwara, and O. Mizuno, "Weighting for combinatorial testing by bayesian inference," in *Proceedings of the 2017 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pp. 389–391, Tokyo, Japan, March, 2017.

[39] S. Fan, Y. Wang, and L. Xiao, "Multidimensional BSDEs with uniformly continuous generators and general time intervals," *Bulletin of the Korean Mathematical Society*, vol. 52, no. 2, pp. 483–504, 2015.

[40] Z. Fei, K. Liu, B. Huang, Y. Zheng, and X. Xiang, "Dirichlet process mixture model based nonparametric bayesian modeling and variational inference," in *Proceedings of the 2019 Chinese Automation Congress (CAC)*, pp. 3048–3051, Hangzhou, China, August. 2019.

[41] X. Hong, "Study of intergenerational mobility and urbanization based on OLS method and ordered probit mode," in *Proceedings of the 2020 Management Science Informatization and Economic Innovation Development Conference (MSIEID)*, pp. 435–447, Guangzhou, China, September. 2020.

[42] H. Chen and J. Ren, "Structure-variable hybrid dynamic bayesian networks and its inference algorithm," in *Proceedings of the 2012 24th Chinese Control and Decision Conference (CCDC)*, pp. 2815–2820, Taiyuan, China, Febrauary. 2012.

[43] H. Guan, J.-C. Ni, Q. Zhang, L. Sun, and K. Wang, "Saliency detection for $\mathbf{L}_{1/2}$ regularization-based SAR image feature enhancement via bayesian inference," in *Proceedings of the IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 4483–4486, Valencia, Spain, July. 2018.

[44] Z. Lijun, H. Guiqiu, L. Qingsheng, and D. Guanhua, "An intuitionistic calculus to complex abnormal event recognition

on data streams," *Security and Communication Networks*, vol. 2021, pp. 1–14, 2021.

[45] T. Kautz, B. H. Groh, J. Hannink, U. Jensen, H. Strubberg, and B. M. Eskofier, "Activity recognition in beach volleyball using a deep convolutional neural network," *Data Mining and Knowledge Discovery*, vol. 31, no. 6, pp. 1678–1705, 2017.

[46] J. Link, T. Perst, M. Stoeve, and B. M. Eskofier, "Wearable sensors for activity recognition in ultimate frisbee using convolutional neural networks and transfer learning," *Sensors*, vol. 22, no. 7, p. 2560, 2022.

[47] T. Aira, K. Salin, T. Vasankari et al., "Training volume and intensity of physical activity among young athletes: the health promoting sports club (HPSC) study," *Advances in Physical Education*, vol. 09, no. 04, pp. 270–287, 2019.

[48] H. Worthington, R. S. McCrea, R. King, and R. A. Griffiths, "Estimation of population size when capture probability depends on individual states," *Journal of Agricultural, Biological, and Environmental Statistics*, vol. 24, no. 1, pp. 154–172, 2019.

[49] M. Burgin and P. Rocchi, "Ample probability in cognition," in *Proceedings of the 2019 IEEE 18th International Conference on Cognitive Informatics & Cognitive Computing (ICCI∗CC)*, pp. 62–65, Milan, Italy, July. 2019.

[50] S. Guopan, "The effect of probability on risk perception and risk preference in decision making," in *Proceedings of the 2010 International Conference on Education and Management Technology*, pp. 690–693, Wasinghton, USA, November. 2010.

[51] T. M. F. Alves, R. O. J. Soeiro, and A. V. T. Cartaxo, "Probability distribution of intercore crosstalk in weakly coupled MCFs with multiple interferers," in *Proceedings of the 2019 IEEE Photonics Conference (IPC)*, pp. 1–4, San Antonio, TX, USA, September. 2019.

[52] B. H. H. Gade, C. N. Vooren, and M. Kloster, "Probability distribution for association of maneuvering vehicles," in *Proceedings of the 2019 22th International Conference on Information Fusion (FUSION)*, pp. 1–7, Ottawa, Canada, July. 2019.

[53] H. Igarashi and K. Watanabe, "Complex adjoint variable method for finite-element analysis of eddy current problems," *IEEE Transactions on Magnetics*, vol. 46, no. 8, pp. 2739–2742, 2010.

[54] J. Qian, J. P. Lu, S. L. Hui, Y. J. Ma, and D. Y. Li, "Dynamic analysis and CFD numerical simulation on backpressure filling system," *Mathematical Problems in Engineering*, vol. 2015, Article ID 160641, 8 pages, 2015.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9828602, 1 page
https://doi.org/10.1155/2023/9828602

Retraction

Retracted: Tech Optimization in Cybersecurity Defenses by Advanced ML Methods: The Use Case of Volleyball Industry

Computational Intelligence and Neuroscience

Received 15 August 2023; Accepted 15 August 2023; Published 16 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Xiao, Z. Bi, and Z. Chen, "Tech Optimization in Cybersecurity Defenses by Advanced ML Methods: The Use Case of Volleyball Industry," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 9907427, 7 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 9907427, 7 pages
https://doi.org/10.1155/2022/9907427

Research Article

Tech Optimization in Cybersecurity Defenses by Advanced ML Methods: The Use Case of Volleyball Industry

Yuchun Xiao,[1] Zhuo Bi,[1] and Zhibin Chen[2]

[1]*Physical Education Teaching and Research Department, Hunan Institute of Technology, Hengyang 421002, China*
[2]*Admissions and Career Service Office, Hunan Institute of Engineering, Xiangtan 411104, China*

Correspondence should be addressed to Zhibin Chen; czb@hnie.edu.cn

Received 31 March 2022; Revised 8 April 2022; Accepted 16 April 2022; Published 17 May 2022

Academic Editor: Konstantinos Demertzis

Individual and team performance can be improved by utilizing "smart" devices and applications that are connected through networks. In sports, the Internet of Things (IoT) refers to all of the "smart" devices and applications linked through networks to reduce injuries to the bare minimum, develop advanced training techniques, and apply analytical advanced sports improvement methodologies to improve sports performance in general. The Internet of Things (IoT) in sports is closely related to the objective of both security and privacy in sports, which has become a topic of crucial concern for the sports business in recent years, as evidenced by the adoption of IoT in sports years. For this reason, security flaws can have catastrophic consequences, including the disclosure of personal data, the manipulation of statistical findings, the harming of organizations' reputations, and enormous financial losses for the sporting organization. One or more of the consequences, as previously mentioned, is related to sports organizations and the athletes who are members of those organizations, and they have a direct impact on the corresponding set of sports-related, medical-related, and paramedical enterprises, specifically those that provide specialized sports equipment and associated services. A critical need to detect and quantify threats has long been recognized to better support decision-making when adopting or constructing a safe and reliable sports Internet-of-Things infrastructure, which is becoming increasingly common. Using advanced machine learning algorithms, this research provides a methodology for technology optimization in cybersecurity defenses that is then used in a unique case study utilizing volleyball players to demonstrate its effectiveness. In conjunction with a Monte Carlo optimization technique, an upgraded variant of fuzzy cognitive maps (FCM) is presented in greater detail. This model is utilized for a specific scenario of risk identification of volleyball industry, assessment, and optimization for IoT sports networks.

1. Introduction

The rapid development of modern sports technology contributes to increased performance and the impetus for exceeding the sport's limits. The big business giants of the sports industry invest in large-scale research for the development and production of state-of-the-art equipment products for athletes in collaboration with scientists, doctors, occupational physiologists, ergometers, and coaches [1]. The body of the athlete of each sport separately is simulated in special computer programs, where all the parameters that could potentially help produce a better athletic result are scientifically analyzed [2]. For these reasons, a set of wearable technologies sensors has been developed that are applied to countless links and fields of sports activity to assist in expanding human boundaries. These sensors are arranged in a sports IoT ecosystem, in which the bodies related to the sports industry participate [3].

Specifically, wearable athletic devices are small devices attached to the body in the form of a waistband or a skin patch. The gadgets then connect through Bluetooth and GPS, transmitting data in real time to IT equipment for analysis, recording, and feedback. Coaches and players can use data to improve performance, prevent injury, and reduce effort. For example, the impact monitor stickers alert coaches and trainers to possible concussions, brain trauma, overexertion, or injured muscles, tendons, and ligaments. Also, soft-tissue injuries can be identified early, allowing

coaches to withdraw athletes before serious issues arise. GPS trackers sewed into the players' clothing track their balance, speed, acceleration, and mobility. In addition, IoT gadgets assess heart rate, metabolism, stress load, core temperature, and the physical repercussions of trauma.

However, the wide variety of "smart devices" in the sports industry introduces new security risks that make the industrial environment particularly dangerous in terms of cybersecurity. In a thorough effort to investigate the problem, the following causes are identified [4]:

(1) Interconnected IoT devices mainly exchange "sensitive" data of athletes, which can be a pole of attraction for malicious activity and mainly black-market products.

(2) Problems of complexity and incompatibility arise from the interaction of many devices and the heterogeneous networks that connect them.

(3) As the IoT is a new and emerging sector, sports industry manufacturers are rushing to adopt smart system solutions without paying attention to security issues related to the confidentiality, integrity, and availability of the data and information they handle.

(4) Most IoT sensors transmit and receive data wirelessly, and they carry the usual risks of wireless security breaches into the IoT sports ecosystem.

(5) In addition, almost all IoT solutions include applications for their operation, monitoring, and control, the corresponding risks associated with software development and especially with authentication, authorization breaches, and the overall security and availability of these applications and connected databases.

(6) The absence of solid computing resources in sports sensors is equivalent to the lack of strong encryption through "smart" sports devices, a fact that opens fields for the discovery and exploitation of the IoT network by malicious attackers.

Therefore, the pursuit of safety and privacy in sports IoT is a significant issue directly related to the evolution of the modern sports industry [5]. Based on the criticality of the environment in question in this work, a technique for optimizing cyber defense technologies using advanced machine learning methods is proposed. Specifically, a risk assessment model based on an advanced form of fuzzy cognitive maps (FCM) is presented [6], combined with a Monte Carlo optimization technique [7], which is applied to a specific scenario of risk identification, assessment, and optimization for the development of IoT sports networks.

2. Related Literature

The literature on cybersecurity and machine learning is rich, and the newer research focuses on dealing with the vast increase in cyber threats in modern information systems [8, 9].

Zhao et al. [10] looked at computational information approaches in IoT information security, such as computational intelligence-enabled cyberattacks and privacy services, cyber defense techniques, intrusion techniques, and data security. They also used computational intelligence capabilities to try to identify new study paths and trends for the growing IoT security challenges. They looked at the status of algorithmic intelligence-enabled cybersecurity concerns and IoT research trends. They outlined the primary obstacles that CI-enabled protection solutions face and new research topics that may be pursued. CI and cyber security-based strategies should be incorporated into the design of IoT to create robustness and make it more reliable.

Li [9] looked into two elements of the confluence of AI and digital protection. Deep learning may be used in cyber security to build intelligent models for malware categorization and intrusion detection. On the other side, AI models will be exposed to various cyberattacks, disrupting their sample, learning, and choices. To prevent adversarial machine learning, maintain privacy in machine learning, and safe federated training, AI models need cybersecurity and mitigation solutions. They then dissected the counterattacks that AI might face, categorized the appropriate defensive techniques, and examined the counterattacks that AI could face. Finally, they highlighted current research on developing a safe AI system from the standpoints of designing encrypted neural networks and implementing secure collaborative deep learning.

As the frequency of cyberattacks grows, Bresniker et al. [11] highlighted cybersecurity as a critical risk for every firm. Computational AI and machine training can assist cyber analysts in detecting threats and making suggestions and expanding the usage of AI/ML in cybersecurity needs worldwide collaboration between businesses, universities, and states. Companies are increasingly concerned about cyberattacks. Adequate cybersecurity necessitates automation, which requires recording cybersecurity analysts' actions. They believe that this will begin to happen soon, and because of the more significant usage of AI and ML in cybersecurity, assaults will become less successful and impactful.

Dasgupta et al. [12] conducted a review of recent work on machine learning in information security, describing the fundamentals of cyberattacks and their defenses, the fundamentals of the most frequently used methodologies, and suggested data mining strategies for information security in terms of capability, dimension reduction, and categorization techniques. This study also covers hostile machine learning and the security features of deep learning approaches. Finally, open topics, difficulties, and future research areas have been offered for aspiring researchers and engineers.

From the literature above, we conclude that researchers have identified that the sheer increase in cyberattacks can only be effectively dealt with the help of machine learning methodologies [12, 13].

3. FCM Methodology

The proposed implementation of technology optimization in cybersecurity defenses is based on the use of FCMs, which are a method of modeling complex systems capable of describing the causal relationships between critical factors' concepts that determine the behavior, symbolic description, and representation of system dynamics of cybersecurity used by the case study's sports organization [14].

FCMs are an excellent concept for analyzing the static and dynamic features of the IoT ecosystem and its evolving dynamic structure. Furthermore, it is an application motivated by various theoretical advancements recently revealed in IoT research. Moreover, the capacity of FCMs to forecast and classify time series is an intriguing element that aligns with the IoT specification.

In application view, FCMs provide concepts that describe various elements of system behavior and how these concepts are reacted to, either by their interaction or by the general dynamics of the system. Fuzzy rules are used to replicate the human experience and expertise of specialists who understand the function of system security and its behavior in various conditions. Each rule reflects an ideal situation or a specific system characteristic.

In our case, the designed FCM consists of nodes-concepts, $C_i, i = 1, 2, 3, \ldots, N$, where N is the total number of ideas to be modeled systemically, which are its qualities, major factors, or properties. Concepts are linked together via connections with different weights, reflecting how concepts interact with one another [6]. Figure 1 depicts a basic concept of FCM.

In our example, there are three forms of causal links between two notions C_i and C_j [15]:

(1) Positive: a positive weight indicates that an increase or reduction in the value of a causal concept leads this concept to move in the same direction W_{ij}.

(2) Negative: changes in the notions of cause and effect occur in opposite directions, as shown by the weight W_{ij} having a negative sign.

(3) Nonexistent: It indicates an interconnection with zero weight. The value of weight, e.g., W_{ij}, describes the concept C_i and affects the concept C_j and the interval [−1, 1].

At each time point, the value of each idea A_i is determined using a block function f from the sum of all the other concepts' influences and the total effect's limitation using the following rule [16]:

$$A_i^{t+1} = f\left(A_i^t + \sum_{i=1, i \neq j} W_{ji} A_j^t\right), \tag{1}$$

where A_i^{t+1} and A_i^t are the concept's values C_i at time $t+1$ and t, respectively, A_j^t is the significance of the notion C_j at time t, W_{ji} is the connection's weight in the direction C_j to the meaning C_i, and f is a block function that is used to limit the concept's value to a certain range, commonly in the interval [0, 1] [17, 18].

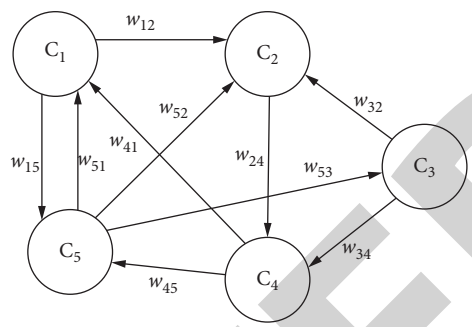


Figure 1: Simple FCM.

A new state of concepts emerges at each phase. After a specific number of repeats, the FCM can arrive at a point of equilibrium, a confined circle, or chaotic behavior. When the FCM reaches a given equilibrium point, it is concluded that the map has converged, and the end state corresponds to the real state of the system to which the values' transition when the map is applied.

The simulation activation function used calculates the value A_j of a C_j concept at the end of an iteration as the sum of its causal concepts' contributions at the start of the iteration [6, 14, 18]:

$$A_j^{(t)} = f\left(\sum_{\substack{i=1 \\ i \neq j}}^{n} A_i^{(t-1)} w_{ij} + A_j^{(t-1)}\right), \tag{2}$$

where $A_j^{(t)}$ is the value of the concept C_j at the end of the iteration, $A_i^{(t-1)}$ is the value of the notation C_j at the beginning of the iteration, w_{ij} is the weight of the relation between C_i and C_j, and f is a threshold function, which is used to normalize the values in each step. The process assumes that the weight table includes an autocorrelation by placing a unit value in the main diagonal of the table (the new value of the concept necessarily equals the previous value plus (or minus) the contribution of the other concepts associated with it). However, there is optional self-correlation because it is determined only by the values of the weight table's principal diagonal; as a result, self-correlation is implied and included in the first term of the equation, while the second term is ignored:

$$A_j^{(t)} = f\left(\sum_{i=1}^{n} A_i^{(t-1)} w_{ij}\right). \tag{3}$$

In cases where there is no information about certain concepts/situations or experts/stakeholders cannot adequately describe the initial state of a variable, the equation takes the form:

$$A_j^{(t)} = f\left(\sum_{i=1}^{n} \left(2A_i^{(t-1)} - 1\right) w_{ij} + 2A_j^{(t-1)} - 1\right). \tag{4}$$

In case the problem incorporates the concept of time delay, the weight of a relationship with the value of the idea I at time t between recital node i and impact node j the lag_{ij} lag of the corresponding effect is

$$A_j^{(t)} = f\left(\sum_{\substack{i=1 \\ i \neq j}}^{n} A_i^{\left(t-\text{lag}_{ij}\right)} w_{ij} + A_j^{(t-1)}\right). \tag{5}$$

The value of node j at time t is calculated as follows:

$$A_j\left(t_{n+1}\right) = f\left(\sum_{i=1}^{N} \mu_{ij}\left(t_n\right) \cdot A_i\left(t_n\right)\right), \tag{6}$$

where $\mu_{ij}(t_n)$ is the value of the result of node i at node j at time t_n. The value of the time function (t) depends on the type of function.

Finally, when the weights have not been defined before the start of the simulation but are dynamically adjusted during the simulation; the activation function for calculating the value of a concept has the following form:

$$A_j^{(t+1)} = A_j^{(t)} + \sum_{\substack{i=1 \\ i \neq j}}^{n} A_{i,\text{scaled}}^{(t)} w_{ij}. \tag{7}$$

The above function is used to scale a value over an interval, as with the term $A_{i,\text{scaled}}^{(t)}$; the method does not use a threshold function [15, 18].

4. The Use Case of the Volleyball Industry

For the modeling of the proposed system, a specialized application scenario was implemented in the sports environment and specifically in a volleyball team. The usage scenario is based on optimizing the technological deployment cycle of IoT applications in cyber security. This particular sports volleyball team has several IoT technologies that do not make the most of their capabilities in the context of cyber security. The scenario aims to eliminate technology waste, make the most of already installed products, and maintain the responsibility of business application and information technology partners [13, 19].

The risk assessment process follows the steps below [20]:

(1) Asset identification and prioritization: servers, customer contact information, critical partner documents, trade secrets, and other assets are examples of assets. Information on software, devices, features, data, interfaces, users, support, mission, purpose, operational needs, IT security policies, IT security architecture, network topology, data storage protection, information flow, and physical security environment is gathered for each component.

(2) Threat identification: a threat could exploit a weakness to breach security and harm the team, including natural disasters, logical threats, and system failures' networks, inadvertent human interference, malicious acts, and vandalism.

(3) Identifying vulnerability analysis: audit reports, vulnerability databases, incident response team data, and system software security analysis can all be used to identify vulnerabilities. Security and evaluation tests, penetration testing techniques, and automated vulnerability scanning are possible.

(4) Controls' analysis: analyzing the controls in place or being designed reduces or eliminates the likelihood that a threat will exploit the system's vulnerability. Technical tools, such as hardware or software, encryption, intrusion detection measures, and authentication and authentication subsystems can be used to execute checks. Security policies, administrative activities, and physical and logical instruments are examples of nontechnical controls.

(5) Calculating the likelihood of an event high, medium, and low categorization verbs examines the possibility of an assault or other adverse effects rather than numerical rating. The possibility of exploiting a vulnerability is calculated by considering the type of vulnerability, the ability and motive of the threat source, and the existence and efficacy of current measures.

(6) Impact evaluation: a threat impact study considers the mission of the system, the methods it employs, its criticality, the value of the data handled, and the system's sensitivity.

(7) Priority is given to information security risks: the level of risk to the system is determined for each threat/vulnerability pair based on the likelihood that the threat will exploit the vulnerability, the impact of successful exploitation of the exposure, and the adequacy of existing or planned security controls for the system to eliminate or reduce the risk.

(8) Controls that are proposed: determine the steps to be made to mitigate the risk for each risk level using the risk level as a guideline: high, medium, and low.

The FCM's architecture is primarily reliant on the experience and expertise of a few experts, who, as experts, have enough knowledge to model a system and offer the initial values of the weights for the concepts' interconnections. In our situation, these weights are determined through a learning process [14]. The algorithm is iterated until a termination requirement, such as the maximum number of iterations or convergence to the target error based on a fitness metric is fulfilled. The FCM training algorithm is based on the Hebb rule and takes into account the fact that each node is activated asynchronously. This means that the balance of the map is accomplished by activating different nodes at different periods. As a result of this method, the FCM nodes are separated into activated nodes and nodes that will be activated. The node price renewal rule is adjusted in this scenario based on the following function [21, 22]:

$$A_i^{t+1} = f\left(A_i^t + \sum_{i=1, i\neq j}^{N} W_{ji} A_j^{\mathrm{act}(t)}\right), \tag{8}$$

where the act pointer indicates the activated node. The weight refresh rule based on this algorithm takes the form

$$w_{ij}^{(t+1)} = \left(1-\gamma^{(t)}\right) w_{ij}^{(t)} + \eta^{(t)} A_i^{(t)} \left(A_j^{(t)} - w_{ij}^{(t)} A_i^{\mathrm{act}(t)}\right), \tag{9}$$

where the learning rate n and the weight reduction factor in repetition t are calculated from the following equation:

$$\eta^{(t)} = b_1 e^{\left(-\lambda_1 t\right)}, \gamma^{(t)} = b_2 e^{\left(-\lambda_2 t\right)}, \tag{10}$$

where $0.01 < b_1 < 0.09$ and $0.1 < \lambda_1 < 1$, while b_2 and λ_2 are positive fixed numbers selected by test and observation.

The optimization problem based on the above modeling can be applied as a problem of finding the mean value and variance of the sum of random variables. Let $1 \leq i \leq n$ be random variables and $Z = \sum_{i=1}^{n} X_i \ i$. If $\mu x_i = (X_i)$ is the average value of $1 \leq i \leq n$, then the mean value of Z is valid [23, 24]:

$$\mu_Z = E(Z) = E\left(\sum_{i=1}^{n} X_i\right) = \sum_{i=1}^{n} E(X_i). \tag{11}$$

Let $Z = X + \gamma$, where X and Y are random variables. Then,

$$\begin{aligned}
&\sigma_Z^2 = E\left[(Z-\mu_Z)^2\right] \Rightarrow \sigma_Z^2 = E\left[(X+Y-\mu_X-\mu_Y)^2\right] \Rightarrow \\
&\sigma_Z^2 = E\left[(X-\mu_X)^2 + 2(X-\mu_X)(Y-\mu_Y) + (Y-\mu_Y)^2\right] \Rightarrow \\
&\sigma_Z^2 = E\left[(X-\mu_X)^2\right] + E\left[(Y-\mu_Y)^2\right] + 2E\left[(X-\mu_X)(Y-\mu_Y)\right].
\end{aligned} \tag{12}$$

So, in the end, it turns out that the following relation holds for the variance:

$$\sigma_Z^2 = \sigma_X^2 + \sigma_Y^2 + 2\rho_{XY}\sigma_X\sigma_Y. \tag{13}$$

The above relation is also generalized for the sum of n random variables:

$$\sigma_z^2 = \sum_{i=1}^{n} \sigma_i^2 + \sum_{i=1}^{n} \sum_{j=1\neq i}^{n} \sigma_{ij}, \tag{14}$$

which is calculated as

$$\sigma_z^2 = \left[\sigma_1 \sigma_2 \sigma_3 \cdots \sigma_n\right] \cdot \begin{bmatrix} 1 & \rho_{12} & \rho_{13} & \cdots & \rho_{1n} \\ \rho_{21} & 1 & \rho_{23} & \cdots & \rho_{2n} \\ \rho_{31} & \rho_{32} & 1 & \cdots & \rho_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \rho_{n2} & \rho_{n3} & \cdots & 1 \end{bmatrix} \cdot \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_n \end{bmatrix} = \sigma^T \cdot C \cdot \sigma. \tag{15}$$

So, we use the collective risk model by looking at the IoT risks of the volleyball team with X_h losses, the number of which is the random variable, as opposed to the individual risk model where we have a fixed number of losses. The random variables X_h, $\hbar = 1, 2, \ldots$, are losses and the random variable N is the number of losses that have occurred up to time t. Then, according to the collective risk model, the random variable of total losses is

$$S = \begin{cases} X_1 + X_2 + \cdots + X_{N_t}, & N_t \geq 1, \\ 0, & N_t = 0. \end{cases} \tag{16}$$

To study the collective risk model, we assume that the random variables are independent. So, the moments of the random variable S for the collective risk model are given by the formulas:

$$\begin{aligned}
E(S) &= E(N_t)\, E(X), \\
\mathrm{Var}(S) &= E(N_t)\mathrm{Var}(X) + \mathrm{Var}(N_t)\, E^2(X).
\end{aligned} \tag{17}$$

From the double mean theorem, we have

$$E(S) = E\left[E(S|N_t)\right] = E\left[E(X)N_t\right] = E(N_t)E(X), \tag{18}$$

respectively, from the double mean theorem; for the variance, we have

$$\begin{aligned}
\mathrm{Var}(S) &= E\left[\mathrm{Var}(X)N_t\right] + \mathrm{Var}\left[E(X)N_t\right], \\
&= \mathrm{Var}(X)E(N_t) + E^2(X)\mathrm{Var}(N_t).
\end{aligned} \tag{19}$$

To find the distribution of the random variable and calculate the probability or probability density function, we use the convolution methodology. For the distribution function [25],

$$F_S(x) = \sum_{n=0}^{\infty} p_n F_X^*(x). \tag{20}$$

And the proper tail function is

$$\overline{F}_S(x) = \sum_{n=1}^{\infty} p_n \overline{F}_X^* n(x). \tag{21}$$

As we can conclude from the above example of risk modeling, with the distribution of individual and team risk assumed, the optimization process can prove to be highly beneficial for the optimal use of the sports IoT network studied. About the classical risk calculation and the independent application methodologies, the above methodology allows the application in a relatively easy, simple, and mainly automated way to change some of the application parameters of the system to create different equilibrium conditions and to re-evaluate the set of situations that affect the formation of risk about available existing-data, as well as to check the two-way interfaces under extreme cases.

5. Conclusions

The urgent need for intelligent detection and dynamic risk assessment in cases of instability, especially regarding cybersecurity, is an ongoing issue of concern to the research community. It is a severe and updated issue, especially when these risks are related to the support and decision-making processes when designing a safe and reliable sports IoT system. This study presents a methodology for tech optimization in cybersecurity defenses by advanced ML methods applied in a particular case study related to the volleyball industry. Specifically, a risk assessment model based on an advanced form of FCM is presented, combined with a unique form of Monte Carlo optimization, applied to a specific individual and collective risk scenario.

As proved, the proposed FCM is an excellent concept for assessing the IoT ecosystem's static and dynamic properties and its challenging structure as it evolves. Additionally, it is an application prompted by several recent theoretical breakthroughs in IoT research. FCMs give concepts that characterize various aspects of system behavior and how these concepts are reacted to, either through their interaction or through the system's overall dynamics. Fuzzy rules are used to emulate the human experience and skill of system security specialists who understand the function of the system and its behavior under varying conditions. Each rule is based on an ideal circumstance or a characteristic of a particular system.

A sophisticated application scenario is used to demonstrate the proposed multiscale simulation idea. This application aims to show how to apply the simulation concept, the core model, the building of detailed models, and the interpretation of simulation results. Because the simulation is relevant to all stages of production and diverse aims, the application displayed for IoT in sports is closely related to the purpose of security and privacy in sports, which has become a critical problem for the sports industry in recent years.

As a future research area, we suggest investigating more advanced optimization approaches, such as bio-inspired heuristic methods, that better reflect an IoT network's self-organization and development potential. Additionally, we believe that a hybrid methodology is necessary, in which the FCM may produce neural network topologies that can be deployed to any scenario-based solely on time limitations. Finally, some improvements must enhance the risk management process of cybersecurity defenses' methodologies, assessing and controlling potential threats.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This study was supported by the Foundation of Hunan Educational Committee, Projects of Hunan Educational Reform (Research and Practice on Implementing Comprehensive Physical Education Teaching in Adult Higher Education, no. HNJG-2020-1085).

References

[1] D. Patel, D. Shah, and M. Shah, "The intertwine of brain and body: a quantitative analysis on how big data influences the system of sports," *Annals of Data Science*, vol. 7, no. 1, pp. 1–16, 2020.

[2] U. Granacher and R. Borde, "Effects of sport-specific training during the early stages of long-term athlete development on physical fitness, body composition, cognitive, and academic performances," *Frontiers in Physiology*, vol. 8, 2017.

[3] S. Banerjee, T. Hemphill, and P. Longstreet, "Wearable devices and healthcare: data sharing and privacy," *The Information Society*, vol. 34, no. 1, pp. 49–57, 2018.

[4] B. Ma, S. Nie, M. Ji, J. Song, and W. Wang, "Research and analysis of sports training real-time monitoring system based on mobile artificial intelligence terminal," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8879616, 10 pages, 2020.

[5] T. Aira, K. Salin, T. Vasankari et al., "Training volume and intensity of physical activity among young athletes: the health promoting sports club (HPSC) study," *Advances in Physical Education*, vol. 9, no. 4, pp. 270–287, 2019.

[6] B. Kosko, "Fuzzy cognitive maps," *International Journal of Man-Machine Studies*, vol. 24, no. 1, pp. 65–75, 1986.

[7] B. H. Dickman and M. J. Gilman, "Monte Carlo optimization," *Journal of Optimization Theory and Applications*, vol. 60, no. 1, pp. 149–157, 1989.

[8] M. E. Webb, A. Fluck, J. Magenheim et al., "Machine learning for human learners: opportunities, issues, tensions and threats," *Educational Technology Research & Development*, vol. 69, no. 4, pp. 2109–2130, 2021.

[9] J. h. Li, "Cyber security meets artificial intelligence: a survey," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 12, pp. 1462–1474, 2018.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4777010, 9 pages
https://doi.org/10.1155/2022/4777010

Research Article

Mitigating Bias and Error in Machine Learning to Protect Sports Data

Jie Zhang and Jia Li

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Jie Zhang; zhangjie19910811@163.com

Received 28 March 2022; Accepted 9 April 2022; Published 11 May 2022

Academic Editor: Konstantinos Demertzis

One of the essential processes in modern sports is doping control. In recent years, specialized methods of artificial intelligence and large-scale data analysis have been used to make faster and simpler detection of violations of international regulations on the use of banned substances. The smart systems in question depend directly on the quality of the data used, as high-quality data will produce algorithmic approaches of correspondingly high quality and accuracy. It is evident that there are many sources of errors in data collections and intentional algorithmic interventions that may result from cyber-attacks, so end-users of artificial intelligence technologies should be able to know the exact origins of data and analytical methods of these data at an algorithmic level. Given that artificial intelligence systems based on incomplete or discriminatory data can lead to inaccurate results that violate the fundamental rights of athletes, this paper presents an advanced model for mitigating bias and error in machine learning to protect sports data, using convolutional neural network (ConvNet) with high-precise class activation maps (HiPrCAM). It is an innovative neural network interpretability technique, wherewith the addition of Bellman reinforcement learning (BRL) and Broyden–Fletcher–Goldfarb–Shanno (BFGS) optimization; it can produce high-precision maps that deliver high definition, clarity, and the input and output capture when the algorithm makes a prediction. The evaluation of the proposed system uses the Shapley value solution from the cooperative game theory to provide algorithmic performance propositions for each of the produced results, assigning partial responsibility to parts of the architecture based on the impact that the efforts have on the relative success measurement, which it has been preset.

1. Introduction

With the commercialization of sport, the lure of a brilliant career with plenty of money and fame is great. Champion-protagonists, whether they are popular team sports or individuals, are idols. The use of substances to increase performance is a well-known practice that concerns the authorities worldwide and those involved in the championship. Doping [1] is related to substances such as anabolic steroids, stimulants, drugs, diuretics, creatine, and many other substances and methods that are very harmful to health and receiving them in large doses for a long time can cause severe problems or even death [2].

An athlete can be tested for doping according to a specific procedure both after a sporting event and without warning during training [3]. Efforts are being made at the national and international level to prevent and reduce the use of doping, which includes, among other things, controls of competitors during nonwarning races [4]. In recent years, specialized methods of artificial intelligence and large-scale data analysis have made it faster and simpler to detect violations of international regulations on banned substances and drugs [5, 6]. The intelligent systems in question depend directly on the quality of the data used, as high-quality data will produce algorithmic approaches of correspondingly high quality and accuracy [7].

There are many errors in data collections and intentional algorithmic interventions that may result from cyber-attacks. Malware can infiltrate a system and change the results of some samples, a process that can easily be proven by repeating the test. However, there are cases where the penetration into the system may involve data alteration or, even worse, the configuration of the artificial intelligence system used to evaluate the samples. Machine learning holds

enormous promise for enhancing products, processes, and research. However, computers typically do not explain their predictions, a hurdle to machine learning adoption. Finding patterns and structures in massive amounts of data in an automated manner is a critical component of data science. It is now driving applications in fields as disparate as cybersecurity. However, such a huge positive influence is accompanied by a significant challenge: how can we grasp the decisions proposed by these algorithms to trust them.

The reason is that machine learning techniques were initially designed for stable environments where training and test data come from the same statistical distribution. However, when these models are applied in the real world, the presence of intelligent and adaptive opponents may, depending on the opponent, to some extent violate this statistical hypothesis. By this logic, a malicious opponent can secretly falsify the input data or parameters of the model to exploit specific vulnerabilities of the learning algorithms and endanger the system's security. So, the end-users of artificial intelligence technologies and especially of high importance systems such as antidoping control [8] should be able to know the exact sources of the data and the analytical ways of using and analyzing these data at an algorithmic level [5].

The need for interpretable and explainable machine learning techniques stems from the need to design intelligible machine learning systems, that is, ones that can be comprehended by a human mind, as well as to understand and explain predictions made by opaque models, such as deep neural networks or gradient boosting machines. The interpretability and explainability [9–11] of neural networks are broad. Usually, they have to do with the ability of the algorithm to explain its decisions and whether humans understand the network behavior. If we know the network's input, we can predict and interpret its output. This process is inherent in simple models but practically impossible to achieve in deep neural networks [9, 12]. In these networks, the basic interpretability technique is CAM. The main problem is that the maps are produced from the last convergent level on CNN, which is much less coherent, so the interpretations are provided without sufficient and precise details [13]. This is problematic for many applications, which require a more specific and detailed justification.

With the rising frequency and complexity of methodologies, stakeholders are increasingly concerned about model disadvantages, data-specific biases, and so on. This study aims to design an architecture that will address the problems mentioned above. Based on CAM, we will try to extend them in such a way as to increase their resolution. This is done by adding BRL- and BFGS-type optimization so that the network can produce high-precision maps that render with outstanding clarity and interpretability, the input and output mapping when the algorithm makes a prediction [14]. After motivating the subject generically, we examine the important developments, including the principles that allow us to study transparent vs. opaque models, as well as model-specific or model-agnostic post hoc explainability approaches, from an organizational standpoint. We also give a quick overview of deep learning models before concluding with a discussion of future research areas.

2. Related Literature

The literature utilizes the terms interpretability, explainability, and class activation mapping to mitigate the issue of doping that is becoming more sophisticated [15].

Finding appropriate mathematical tools to model deep neural networks' expression ability and training ability and gradually transforming parameter-based deep learning based on empiricism into deep learning based on quantitative guidance of some evaluation indicators is a new topic in artificial intelligence research. The authors of the [16] they study how the neural network search technology in autonomous machine learning can be used as a tool to assist people in furthering their understanding of the "black box" problem of artificial intelligence.

Angelov et al. [9] pinpointed explainability and proposed a solution that addresses the bottlenecks of the traditional deep learning approaches. A deep learning architecture linked reasoning and learning together, which they delivered. It is noniterative, nonparametric, and human-friendly from the user's point of view. Their method outperformed the other techniques in tough classification cases, including deep learning, accuracy, time to train, and an explainable classifier. They aim to continue their research in developing a tree-based architecture, synthetic data generation, and local optimization to improve the proposed deep answerable approach.

Mehrotra et al. [17] stated that when the protected attributes were noisy or missing some or all of the entries, it was also attempted to counteract bias in a selection. Algorithms need to account for real-world noise to avoid bias. There was some thought put into a model of noise in which the protected properties were given a probability. They created a framework for mitigating bias that could satisfy a wide range of fairness requirements with a minimal multiplicative error and a high degree of probability. Their empirical analysis found that their methodology could achieve a high level of fairness on standard measures, even when the probabilistic information regarding protected qualities was skewed, and had a better tradeoff between utility and fairness than several previous methods.

In addition, in this study [18], the authors focus on a popular and commonly used XAI method, layer-wise relevance propagation (LRP). LRP has evolved as a method since its first assertion, and a best practice for using the technique has arisen tacitly, based solely on humanly witnessed data. They also study—and for the first time quantify—the effect of existing best practices on feedforward neural networks in a visual object identification context. The results show that the layer-dependent approach to LRP used in recent literature better depicts the model's reasoning while improving object localization and class discriminability.

Leon [15] concentrated on the Shapley value and created a technique for refining the architecture of algorithms based on it. This game-theoretic solution idea measures the importance of each network piece to accomplishment. The final setting was still a classic layered collection of nodes in their scenario. They demonstrated that the quantity of nodes could be massively reduced while keeping a good, user-

defined efficiency by using the Shapley value and a hill-climbing process to finish the fine-tuning. They noted in their findings that more network pieces might be reduced simultaneously, resulting in faster execution times and better outcomes. Furthermore, calculation time was not a problem when employing an estimate of the Shapley value since the user could choose between better precision and longer execution time. Finally, many synapses might be destroyed simultaneously, reducing the number of steps required to complete the operation.

Lundberg et al. [11] did an intriguing study on the developing conflict among model accuracy and interpretability. They proposed Shapley Additive exPlanations, a cohesive approach for analyzing predictions. For each estimate, this system gave a significant value to each feature. It featured the discovery of a new class of additive feature significance measures and empirical models, demonstrating that this class has a single answer with a set of desired qualities. The proposed new strategies critical insights gained through the convergence that outperformed earlier methodologies of computing performance and compatibility with guesswork. The development of speedier model-type-specific estimate techniques with limited information, the integration of work on estimating interaction effects from game theory, and the definition of the additional explanatory classifier are all potential future stages.

Finally, in 2016, Zhou et al. [19] introduced class activation mapping (CAM) for CNNs with globally averaged mixing. They could categorize trained CNNs without utilizing any bounding box annotations because of their method. They were able to show the predicted class scores on every given picture using category activation maps, which highlighted the discriminative object sections discovered by CNN. They tested their strategy on semi-supervised object localization and found that their global average pooling CNNs could execute accurate object localization. They also showed that the CAM localization approach applied to additional vision tasks.

3. Methodology

A CAM is an input area that activates a CNN for a particular class [19]. With the map of a class, we can interpret that features of the data set make CNN choose the class to which it belongs. This becomes especially interesting when we produce the CAM of the network that predicts the network, where we see where the network focused when it made its prediction. For a network to create CAM, it must combine a global average pooling (GAP) level at the end of its architecture and a unique fully connected (FC) level [20].

For a given convergent network, let $f_k(x, y)$ be the activation of neuron k of the last convergent level, at the location (x, y). The next level is a GAP that performs the following operation [21]:

$$F^k = \sum_{x,y} f_k(x, y). \tag{1}$$

Next, the weighted average of all the neurons is passed to the softmax activation function:

$$z_c = \sum_k w_k^c F^k, \tag{2}$$

where w_k^c is the weight of the neuron k for class c and z_c is the value given by the neuron for this class (that is, the input of softmax). Combining the above relationships, the CAM for class c can be produced as [22]

$$S_c(x, y) = \sum_k w_k^c f_k(x, y). \tag{3}$$

A more intuitive explanation is that from the last level weight table, which correlates the GAP output with each output class, we isolate the desired class c. The weight table column we isolated shows us how each of the GAP outputs affects this class. Each GAP output, however, is nothing more than the average value of the previous level activation map (i.e., the last convergent). In this sense, by summarizing the map at a value, we can see that map affects the input and to what extent. Due to the cohesive network structure, the local input characteristics are retained in the activation maps [23, 24]. Finally, we create the CAM by combining these two pieces of information, namely the activation maps and their relation to class c. We do this by taking the sum of all the maps, weighted by the weight of each one.

To view the maps on the original image, it must be converted to have the same consistency. During the last step of the process, the produced map is of very low coherence. It is an ideal solution for the evaluation and, above all, the interpretability of the categorization process. This is due to the inherent feature of CNN that their last level is much lower than the input. We propose a secondary architecture to solve this problem, which aims to create HiPrCAM.

This technique uses BRL and Quasi-Newton-type optimization [15, 25] to produce high-precision maps that deliver input and output when the algorithm predicts outstanding clarity and interpretability. Specifically, in reinforcement learning, the agent receives a representation of the state of the environment and acts, influencing the next state of the environment and receiving a reward. The reward signal is a sequence of real numbers the agent uses to make decisions. In general, the agent's goal is to maximize the sum of the total rewards he receives from the environment in perpetuity and not maximize the immediate reward. This idea is expressed by the reward hypothesis, according to which any goal can be modeled as maximizing the expected value of the sum of a graded reward signal. Since an agent's goal is to select actions to maximize future returns, the value $\gamma = 1$ in an ongoing job would make it impossible to compare different values of the random variable. In each case, the discount factor γ determines the value of the future rewards. A reward at time $t + k$ contributes to the sum of the returns. Therefore, the discount factor regulates how vital the long-term rewards are to the agent. For $\gamma = 0$, the process of maximizing the expected return is reduced to selecting the action with the highest immediate reward. For $\gamma \longrightarrow 1$, the agent gives more value to the long-term rewards [26]. The way the agent makes decisions is determined by the policy he follows. The policy is defined as a function π: $S \longrightarrow p(A)$, which corresponds to states in probability distributions in the action area, and we consider that it is stationary [27]:

$$\pi(a|s) = \Pr[A_t = a|S_t = s]. \tag{4}$$

The status value function is defined as the function v_π: $S \longrightarrow R$ that gives the expected return from a state s, assuming that the agent selects actions based on a policy π:

$$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s], \quad s \in \mathcal{S}. \tag{5}$$

Respectively we can define the state-action value function q_π: $S \times A \longrightarrow R$, which gives the expected return from a state s, assuming that the agent selects action a and then behaves according to the policy π:

$$q_\pi(s,a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a], \quad s \in \mathcal{S},\ a \in \mathcal{A}. \tag{6}$$

A fundamental property of value functions is that they can be expressed retrospectively using the observation that [28]:

$$G_t = R_{t+1} + \gamma G_{t+1}. \tag{7}$$

And the law of total expectation E [X] = E [E [X |Y]] we get

$$\begin{aligned} v_\pi(s) &= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s] \\ &= \mathbb{E}_\pi[R_{t+1} + \gamma v_\pi(s')|S_t = s]. \end{aligned} \tag{8}$$

And, respectively, for the status-action value function:

$$q_\pi(s,a) = \mathbb{E}_\pi[R_{t+1} + \gamma q_\pi(s',a')|S_t = s,\ A_t = a]. \tag{9}$$

Developing the above function for the possible actions from the state's according to the policy π and for its dynamics we have

$$v_\pi(s) = \sum_{a\in\mathcal{A}} \pi(a|s) \sum_r \sum_{s'\in\mathcal{S}} p(r,s'|s,a)[r + \gamma v_\pi(s')], \tag{10}$$

which is the Bellman equation for the condition value function [12].

The proposed methodology uses the Bellman equation to implement a learning system that seeks to learn through direct interaction with the environment. When applied to the value function, the Bellman equation separates it into two parts: the current reward and the discounted future values. Specifically, the Bellman equation with the help of $R_s^a, P_{s,s'}^a$ is converted to

$$v_\pi(s) = \sum_{a\in\mathcal{A}} \pi(a|s)\left[R_s^a + \gamma \sum_{s'\in\mathcal{S}} P_{s,s'}^a v_\pi(s')\right]. \tag{11}$$

This equation simplifies the computation of the value function, allowing us to find the best solution of a complex problem by breaking it down into simpler, recursive subproblems and finding their optimal solutions rather than summing over numerous time steps. Assuming that the decision for action an in state's has been made, the equation for possible actions a′ from state s′ according to policy π and its dynamics becomes

$$q_\pi(s,a) = \sum_{a'\in\mathcal{A}} \sum_r \sum_{s'\in\mathcal{S}} \pi(a|s) p(r,s'|s,a)[r + \gamma q_\pi(s',a')]. \tag{12}$$

Respectively:

$$q_\pi(s,a) = R_s^a + \gamma \sum_{s'\in\mathcal{S}} P_{s,s'}^a \sum_{a'\in\mathcal{A}} \pi(a'|s') q_\pi(s',a'). \tag{13}$$

The following two diagrams depicted in Figure 1 explain a standard for identifying the variables and their relationships to facilitate comprehension of the formulation in the suggested approach:

So based on the Bellman equation, we can calculate the value of a state's as the weighted average value according to policy π for each pair (s, a):

$$v_\pi(s) = \sum_{a\in\mathcal{A}} \pi(a|s) q_\pi(s,a). \tag{14}$$

Respectively, the value of a state-action pair is equal to the sum of the immediate reward given by the environment and the discounted, weighted according to the dynamics of the environment, average value of each possible next state's [20, 29]:

$$q_\pi(s,a) = R_s^a + \gamma \sum_{s'\in\mathcal{S}} P_{s,s'}^a v_\pi(s'). \tag{15}$$

The above shows that the specific methodology requires optimization to better deal with non-linear and bad states. The state of a function describes the rate at which the function changes when minor disturbances occur in its input data. Operations that change rapidly with minor changes in data can cause many problems in iterative processes where minor input rounding errors cause significant changes in output [30].

In the proposed smart algorithmic framework, we use an optimization that deals with such objective functions using quasi-Newton type second-order information of the stochastic method. The quasi-Newton method is a class of optimization methods that attempt to address the computationally expensive it is to calculate the Hessian and invert it, especially when dimensions get large. The quasi-Newton approach is used to include multidimensional objective functions. This method imposes additional limitations instead of approximating the second derivative with a finite difference as in the secant technique. However, the standard-issue persists, as each new Hessian must need to be calculated using historical gradient information at each iteration.

So, the BFGS methodology is used, which significantly improves the convergence rates of the technique. Specifically, the iterative formula BFGS for minimizing a twice-continuously generable function F: $R^d \longrightarrow R$ is

$$w_{k+1} \leftarrow w_k - \alpha_k H_k \nabla F(w_k). \tag{16}$$

H_k is a symmetric and positively defined array that approaches the array $\nabla F(w_{k+1})$. The difference of the above iterative formula that makes it quasi-Newton is that the sequence $\{H_k\}$ is updated dynamically when the algorithm is executed and is not just a second-order derivative calculation in each iteration [31, 32]. The maximum paraboloid is presented in Figure 2.

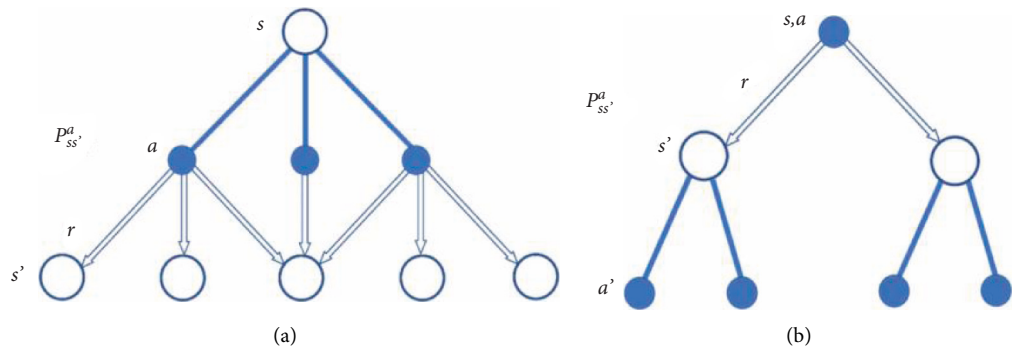


Figure 1: Diagrams for (a) $V\ \pi$ (s) and (b) Q π (s, a) (https://towardsdatascience.com/).

Specifically, the new inverse Essien is given by the difference in the parametric vectors resulting from the iterative process and the difference in the slopes in them [7, 28, 33]:

$$\begin{aligned} s_k &\coloneqq w_{k+1} - w_k \chi \alpha l y_k \\ &\coloneqq \nabla F(w_{k+1}) - \nabla F(w_k). \end{aligned} \tag{17}$$

The reverse update type of the essential table for the BFGS method is

$$H_{k+1} \leftarrow \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right)^T H_k \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k}. \tag{18}$$

The above formula satisfies the quasi-Newton under certain conditions:

$$\begin{aligned} H_{k+1} y_k &= \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right)^T H_k y_k \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k} y_k \\ &= \left(H_k y_k - \frac{y_k^T s_k}{s_k y_k^T} H_k y_k\right)\left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k} y_k \\ &= (H_k y_k - H_k y_k)\left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + s_k = s_k \Rightarrow H_{k+1}^{-1} s_k = y_k. \end{aligned} \tag{19}$$

The above proves that BFGS has a locally super-linear convergence rate, and this speed is achieved only from first-order information, without the need to solve a linear system, significantly reducing the cost per repetition of the method while ensuring linear convergence.

3.1. Method Evaluation. Abnormal Blood Profile Score (ABPS) [28] is used to detect blood doping in sports and was tested using artificial data. As part of the package's ABPS functionality, users must provide the seven hematological marker values for one or more samples. The score or scores will then be calculated and returned. As a single data frame (the basic structure for managing data in R) containing the seven parameters, or by specifying each of the seven variables individually (the standard units are indicated): HCT (hematocrit level, in percent), HGB (the hemoglobin level, in g/dL), MCH (the mean corpuscular hemoglobin, in pg), MCHC (the mean corpuscular hemoglobin concentration, in (g/dL)), MCV (the Mean corpuscular volume, in fL), RBCs 361 of the 607 cases with fabricated data are expected, and 246 are abnormal.

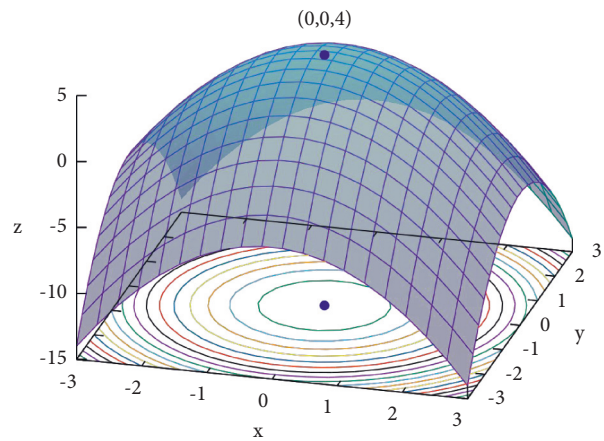


Figure 2: Maximum paraboloid (https://wikipedia.org/).

Table 1: Comparison of performance.

ML method	Recall	Precision	f1-score	Accuracy	auc
Proposed nn	92.85	92.80	92.80	92.86	0.9678
mlp	89.80	89.90	89.90	89.91	0.9317
svm	92.90	93.00	92.95	92.99	0.9898
xgboost	93.10	93.15	93.15	93.23	0.9916

Initially, a test of the proposed neural network and competing methods was performed to evaluate the categorization ability of the system. The results are presented in Table 1.

The evaluation of the proposed system uses the Shapley value solution from the cooperative game theory, to provide algorithmic performance propositions for each of the produced results, assigning partial responsibility to parts of the architecture based on the impact that the efforts have on the relative success measurement in which they have been preset. Specifically, the Shapley value has been proposed as a cooperation game solution, given as $\varphi i(v)$ for the ith player. It proposes a specific payout for each player from the total winnings from all N players in the game. This share is

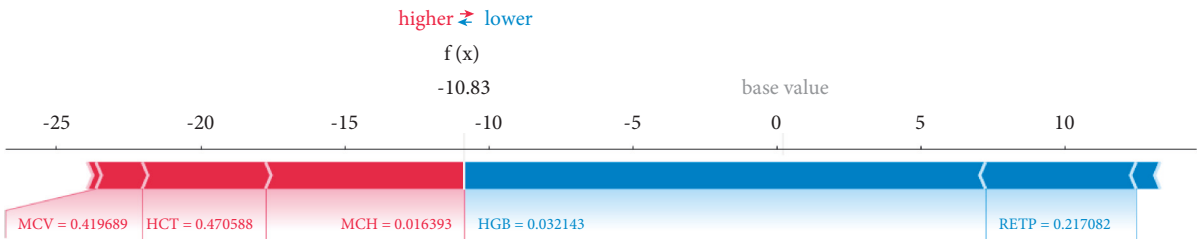


Figure 3: Shapley explanations for the prediction (10 evaluations) of the random sample 156.

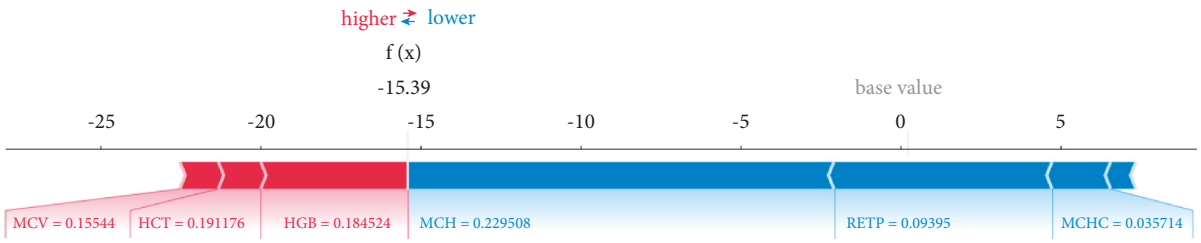


Figure 4: Shapley explanations for the prediction (10 evaluations) of the random sample 309.

proportional to how important each player is in the coalition. The foundation of this value was based on four axioms, which are [15, 34–36]:

(1) Symmetry: if i and j are two players of equal value in a game, i.e., when

$$v(S \cup \{i\}) = v(S \cup \{j\}). \tag{20}$$

For each coalition S of N, then $\varphi_i(v) = \varphi_j(v)$.

(2) Cumulative: if two games are combined that have the characteristic equations v and w, respectively, then the total payout of a player i who participates in both games is equal to the payout that he would have separately in the game with characteristic equation v plus the payout had separately in the game with distinct equation w: $\varphi_i(v+w) = \varphi_i(v) + \varphi_i(w)$.

(3) Efficiency: the sum of the payouts of all players is equal to the total payout of the game. The relation describes this condition:

$$\sum_{i=1}^{n} \phi_i(v) = v(N). \tag{21}$$

(4) Zero player: the value of Shapley $\phi_i(v)$ for each player with zero contribution to the coalition is zero, or otherwise a player's contribution is zero when $v(S \Upsilon \{i\}) = v(S)$ in a coalition S.

The Shapley value satisfies the above four axioms and is given by the relation [37, 38]:

$$\phi_i(v) = \sum_{S, i \notin S} \frac{n_s!(n - n_s - 1)!}{n!} (v(S \cup \{i\}) - v(S)), \tag{22}$$

where n_S is the number of players in the coalition S, n is the number of players in the game, $v(S)$ is the value of the characteristic equation for coalition S, and $v(S \cup \{i\})$ is the value of the characteristic equation for coalition S after player i joins him.

The factor $[v(S \Upsilon \{i\} - v(S)]$ indicates the increase or decrease in the payout of Coalition S due to the participation of Player i in this coalition. It calculates the extra profit or loss that the involvement will cause to player i in an already formed partnership S. The factor:

$$\frac{n_s!(n - n_s - 1)!}{n!}. \tag{23}$$

Indicates the probability that player i is the (S+1) participant in the S coalition that already has n_s players from the n participating in the game.

The image below uses a selection of a random sample from the data set to represent the typical attribute values. Then ten samples are used to estimate the Shapley values for a given prediction. This task requires $10 \times 1 = 10$ evaluations of the model. Figure 3 shows the procedure for sample 156, Figure 4 for sample 309, and Figure 5 for sample 567.

Essentially the Shapley value is the sum of the extra profit (or loss respectively) due to the i-player participation in all possible alliances separately, multiplying the extra profit by the probability that player i is the next participant in each association. Thus, the Shapley price gives a unique solution and is monotonous. The greater the player's influence, the greater the payout that he distributes. Shapley values also have universal explanation capabilities, summing the values of a set of samples [34, 35].

Extensive research was then conducted to evaluate the values of the variables, how they contribute to the prediction, and to explain each decision of the implemented models using the Shapley values. Figure 6 shows the classification of the values of the variables used in the bar plot. In contrast, the exact effect value of each is presented in the adjacent table, which shows the period of influence of each variable in the given problem.

Figure 7 depicts the data set's overall impact concerning each attribute. Each attribute's Shapley values is summed across all samples in the group, and then the details are ranked accordingly. The beeswarm plot provides a concise

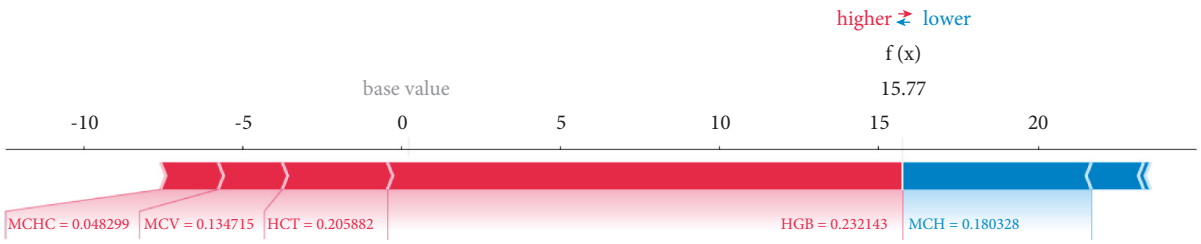
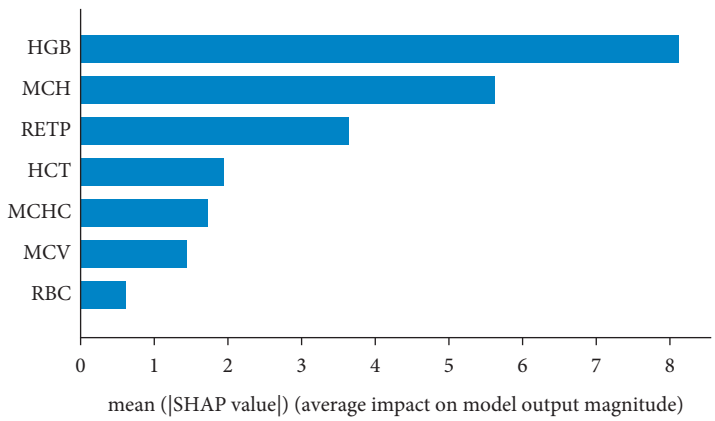


Figure 5: Shapley explanations for the prediction (10 evaluations) of the random sample 567.



ID	variable	effect
0	MCH <= 0.05	0.714445
1	0.03 < RETP <= 0.05	0.178413
2	0.08 < HGB <= 0.11	-0.120140
3	0.15 < HCT <= 0.19	0.035600
4	0.08 < MCV <= 0.11	0.021048
5	0.07 < RBC <= 0.11	-0.016332
6	0.04 < MCHC <= 0.05	0.001669

Figure 6: Average impact on model output and effect of each variable.

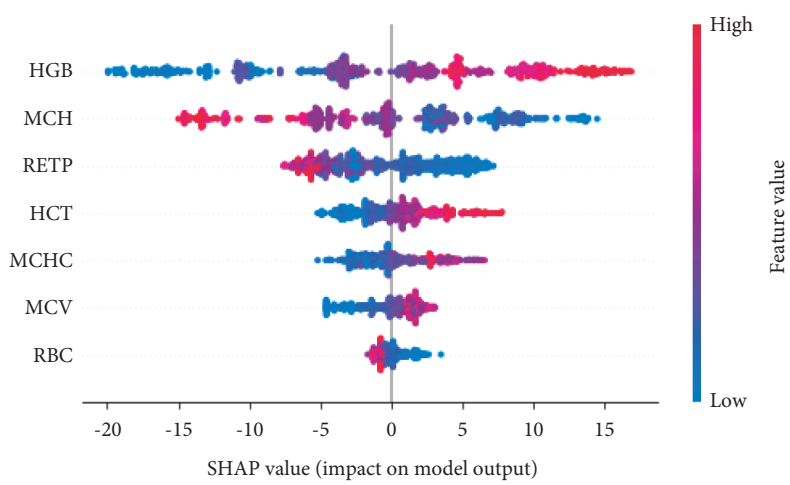


Figure 7: Summary beeswarm plot.

description of how the top attributes in a data set influence the model's output. The supplied explanation is represented by a single dot on each feature flow in each case. The feature's Shapley value defines the dot's x position, and the dots pile-up along with each feature row display density. Color is used to indicate a feature's original value. From top to bottom, the model's most important features are highlighted. Dots represent each feature of the package, and the color of the dot indicates how important it is (blue corresponds to a low value, while red to a high value). The dot's horizontal position on the axis is determined depending on its Shapley value.

We can observe that the HGB feature has the most significant impact on the model predictions. A sample with high Shapley values (red dots) is more likely to be atypical. Because of this, hence the Shapley value has a high positive effect. On the other hand, the Shapley value harms the forecast because it has low values (blue dots). This means that it raises the possibility that the forecast does not come from a standard sample [27, 39].

As it is understood, the proposed model can identify the most critical areas of the entrance and at the same time provide clear explanations for the final decision of the problem. Thus, the information passed to the classifier

during the training becomes less and less until we have reached the slightest possible input that does not affect his predictive ability. At the end of the training, the model has already learned to recognize the essential pieces of information provided by the class identifiers.

4. Conclusions

In this work, an intelligent framework for protecting sensitive data with explainable artificial intelligence methods has been proposed. Specifically, using an innovative ConvNet assisted by a combined system of an innovative BRL system optimized with the BFGS algorithm, it produces HiPrCAMs, which fully explain and render the input and output mapping with great clarity when the algorithm makes a prediction.

The test of the proposed system was performed on a set of data related to detected in the blood of athletes if there are illegal substances. Respectively, the evaluation of the method was done using Shapley values, which are inspired by the cooperative game theory, to provide algorithmic performance proposals for each of the produced results, assigning partial responsibility to parts of the architecture based on the effect they have on the final decision.

The extension of the proposed system with additional possibilities for recording local and universal variables and their dependence on intermediate representations of the neural network is considered very important to achieve even more accurate and complete knowledge of using the input data.

Data Availability

The data used in this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This study was supported by General Project of Humanities and Social Sciences Research in Colleges and Universities of Henan Province in 2023 (No. 2023-ZDJH-565, Practical Research on 'Sports Technology Feedback Method' in Aerobics Teaching Reform in We Media Environment).

References

[1] D. A. Baron, D. M. Martin, and S. Abol Magd, "Doping in sports and its spread to at-risk populations: an international review," *World Psychiatry: Official Journal of the World Psychiatric Association (WPA)*, vol. 6, no. 2, pp. 118–123, 2007.

[2] A. J. Schneider and T. Friedmann, "The problem of doping in sports," *Gene Doping in Sports: The Science and Ethics of Genetically Modified Athletes*, vol. 51, pp. 1–9, 2006.

[3] B. Sharma, "A Critical Analysis of the Impact of Doping in Sports Domain," *International Journal of Law Management & Humanities*, vol. 4, 2022.

[4] M. Negro, N. Marzullo, F. Caso, L. Calanni, and G. D'Antona, "Opinion paper: scientific, philosophical and legal consideration of doping in sports," *European Journal of Applied Physiology*, vol. 118, no. 4, pp. 729–736, 2018.

[5] M. R. Rahman, J. Bejder, T. C. Bonne et al., "AI-based Approach for Improving the Detection of Blood Doping in Sports," 2022, http://arxiv.org/abs/2203.00001.

[6] G. Lima, B. Muniz-pardos, A. Kolliari-turner et al., "Anti-doping and other sport integrity challenges during the COVID-19 pandemic," *The Journal of Sports Medicine and Physical Fitness*, vol. 61, no. 8, 2021.

[7] J. Vamathevan, D. Clark, P. Czodrowski et al., "Applications of machine learning in drug discovery and development," *Nature Reviews Drug Discovery*, vol. 18, no. 6, pp. 463–477, 2019.

[8] T. Kelly, A. Beharry, and M. Fedoruk, "Applying Machine Learning Techniques to Advance Anti-doping," 2019, https://www.scholarsresearchlibrary.com/articles/applying-machine-learning-techniques-to-advance-antidoping-18437.html.

[9] P. Angelov and E. Soares, "Towards explainable deep neural networks (xDNN)," *Neural Networks*, vol. 130, pp. 185–194, 2020.

[10] M. Carletti, C. Masiero, A. Beghi, and G. A. Susto, "Explainable machine learning in industry 4.0: evaluating feature importance in anomaly detection to enable root cause analysis," in *Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 21–26, Bari, Italy, Oct. 2019.

[11] S. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," Nov. 2017, http://arxiv.org/abs/1705.07874.

[12] J. Gawlikowski, C. R. N. Tassi, M. Ali et al., "A Survey of Uncertainty in Deep Neural Networks," Jul. 2021, http://arxiv.org/abs/2107.03342.

[13] S. Cano-Berlanga, J.-M. Giménez-Gómez, and C. Vilella, "Enjoying cooperative games: the R package GameTheory," *Applied Mathematics and Computation*, vol. 305, pp. 381–393, 2017.

[14] A. Dinar and A. Wolf, "International markets for water and the potential for regional cooperation: economic and political perspectives in the western Middle East," *Economic Development and Cultural Change*, vol. 43, no. 1, pp. 43–66, 1994.

[15] F. Leon, "Optimizing neural network topology using Shapley value," in *Proceedings of the 2014 18th International Conference on System Theory, Control and Computing (ICSTCC)*, pp. 862–867, Sinaia, Romania, October 2014.

[16] S. Li and Y. Gao, "Using NAS as a tool to explain neural network," in *Proceedings of the 2020 Chinese Automation Congress (CAC)*, pp. 1865–1868, Shanghai, China, August 2020.

[17] A. Mehrotra and L. E. Celis, "Mitigating bias in set selection with noisy protected attributes," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 237–248, Virtual Event Canada, March 2021.

[18] M. Kohlbrenner, A. Bauer, S. Nakajima, A. Binder, W. Samek, and S. Lapuschkin, "Towards best practice in explaining neural network decisions with LRP," in *Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, Glasgow, UK, July 2020.

[19] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning Deep Features for Discriminative Localization," pp. 2921–2929, 2016, https://openaccess.thecvf.com/content_cvpr_2016/html/Zhou_Learning_Deep_Features_CVPR_2016_paper.html.

[20] A. Dhillon and G. K. Verma, "Convolutional neural network: a review of models, methodologies and applications to object

detection," *Progress in Artificial Intelligence*, vol. 9, no. 2, pp. 85–112, 2020.

[21] K. Md, R. Alam, N. Siddique, and H. Adeli, "A dynamic ensemble learning algorithm for neural networks," *Neural Computing & Applications*, vol. 32, no. 12, pp. 8675–8690, 2020.

[22] Q. Rao, B. Yu, K. He, and B. Feng, "Regularization and iterative initialization of softmax for fast training of convolutional neural networks," in *Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Budapest, Hungary, July 2019.

[23] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-Based Fully Convolutional Networks," Jun. 2016, http://arxiv.org/abs/1605.06409.

[24] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep `s," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.

[25] S. Gupta, S. Al-Obaidi, and L. Ferrara, "Meta-analysis and machine learning models to optimize the efficiency of self-healing capacity of cementitious material," *Materials*, vol. 14, no. 16, p. 4437, 2021.

[26] J. Xie, X. Bai, D. Feng, and D. Gan, "Peaking cost compensation in northwest China power system," *European Transactions on Electrical Power*, vol. 19, no. 7, pp. 1016–1032, 2009.

[27] M. Ahmadlou and H. Adeli, "Enhanced probabilistic neural network with local decision circles: a robust classifier," *Integrated Computer-Aided Engineering*, vol. 17, no. 3, pp. 197–210, 2010.

[28] R. de Adelhart Toorop, F. Bazzocchi, L. Merlo, and A. Paris, "Constraining flavour symmetries at the EW scale I: the A 4 Higgs potential," *Journal of High Energy Physics*, vol. 2011, no. 3, p. 35, 2011.

[29] L. T. Thanh Bui and Q.-H. Nguyen, "Gradient weighted norm inequalities for very weak solutions of linear parabolic equations with BMO coefficients," *Asymptotic Analysis*, vol. 127, no. 4, pp. 339–353, 2022.

[30] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning Deep Features for Discriminative Localization," Dec. 2015, http://arxiv.org/abs/1512.04150.

[31] A. J. M. Garrett, "Review: probability theory: the logic of science, by E. T. Jaynes," *Law, Probability and Risk*, vol. 3, no. 3-4, pp. 243–246, 2004.

[32] J. Qian, J. P. Lu, S. L. Hui, Y. J. Ma, and D. Y. Li, "Dynamic analysis and CFD numerical simulation on backpressure filling system," *Mathematical Problems in Engineering*, vol. 2015, pp. 1–8, 2015.

[33] A. Ferrari, L. Zhao, and W. Alhoshan, "NLP for requirements engineering: tasks, techniques, tools, and technologies," in *Proceedings of the 2021 IEEE/ACM 43rd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, pp. 322-323, Madrid, ES, Feb. 2021.

[34] B. Guo, S. Hao, G. Cao, and H. Gao, "Profit distribution of liner alliance based on Shapley value," *Journal of Intelligent and Fuzzy Systems*, vol. 41, no. 4, pp. 5081–5085, 2021.

[35] F. Meng, X. Chen, and Q. Zhang, "Some uncertain generalized Shapley aggregation operators for multi-attribute group decision making," *Journal of Intelligent and Fuzzy Systems*, vol. 29, no. 4, pp. 1251–1263, 2015.

[36] K. Demertzis, K. Tsiknas, D. Takezis, C. Skianis, and L. Iliadis, "Darknet Traffic Big-Data Analysis and Network Management to Real-Time Automating the Malicious Intent Detection Process by a Weight Agnostic Neural Networks Framework," 2021.

[37] L. Petrosyan, A. Sedakov, H. Sun, and G. Xu, "Time consistency of the interval Shapley-like value in dynamic games," *Journal of Intelligent and Fuzzy Systems*, vol. 30, no. 4, pp. 1965–1972, 2016.

[38] S. Lipovetsky and W. M. Conklin, "Meaningful regression analysis in adjusted coefficients Shapley value model," *Model Assisted Statistics and Applications*, vol. 5, no. 4, pp. 251–264, 2010.

[39] L. E. B. Salasar, J. G. Leite, and F. Louzada, "Likelihood-based inference for population size in a capture–recapture experiment with varying probabilities from occasion to occasion," *Braz. J. Probab. Stat.*, vol. 30, no. 1, pp. 47–69, 2016.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9821676, 1 page
https://doi.org/10.1155/2023/9821676

Retraction

Retracted: Tackling Explicit Material from Online Video Conferencing Software for Education Using Deep Attention Neural Architectures

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Yang and S. Xu, "Tackling Explicit Material from Online Video Conferencing Software for Education Using Deep Attention Neural Architectures," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6334802, 11 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6334802, 11 pages
https://doi.org/10.1155/2022/6334802

Research Article

Tackling Explicit Material from Online Video Conferencing Software for Education Using Deep Attention Neural Architectures

Yongzhao Yang and Shasha Xu

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Yongzhao Yang; yangyongzhao2022@163.com

Received 12 March 2022; Revised 18 March 2022; Accepted 18 April 2022; Published 11 May 2022

Academic Editor: Konstantinos Demertzis

The spread of the COVID-19 pandemic affected all areas of social life, especially education. Globally, many states have closed schools temporarily or imposed local curfews. According to UNESCO estimations, approximately 1.5 billion students have been affected by the closure of schools and the mandatory implementation of distance learning. Although rigorous policies are in place to ban harmful and dangerous content aimed at children, there are many cases where minors, mainly students, have been exposed relatively or unfairly to inappropriate, especially sexual content, during distance learning. Ensuring minors' emotional and mental health is a priority for any education system. This paper presents a severe attention neural architecture to tackle explicit material from online education video conference applications to deal with similar incidents. This is an advanced technique that, for the first time in the literature, proposes an intelligent mechanism that, although it uses attention mechanisms, does not have a square complexity of memory and time in terms of the size of the input. Specifically, we propose the implementation of a Generative Adversarial Network (GAN) with the help of a local, sparse attention mechanism, which can accurately detect obscene and mainly sexual content in streaming online video conferencing software for education.

1. Introduction

Going through the second wave of the digital age, humanity is now called upon to manage the multilevel social effects that arise through the ever-accelerating growth of the Internet. At the international level, efforts are being made to establish an institutional framework for protecting minors using new technologies. But as children's use of the Internet and new technologies are constantly evolving, few countries have implemented a fully operational framework in enacting regulations for illegal behaviors exclusively in the Internet environment. The harmonization of the laws of the nations is an essential precondition for the effective transnational treatment of cybercrime and the protection of minors. The prevention and response initiatives proposed as good practices by experts and stakeholders focus on children, parents, and educators, whose effectiveness is constantly being explored because Internet issues are continually evolving [1–3].

Obscene and mainly sexual content, such as pornography, is not allowed in applications accessible to minors, primarily in educational environments. In general, the modern legal framework imposes strict policies on nudity and sexual content, especially when it relates to children. Implementing these policies from a technological point of view is mainly based on the development and implementation of techniques (filters) that implement these policies. Corresponding techniques are applied internationally in the educational networks of many advanced countries and prevent with significant success rate access to sites belonging to categories such as: "porn" (sites with pornographic content), "gambling" (gambling sites), "drugs" (websites promoting drugs), "aggressive" (websites promoting aggressive behavior and racism), and "violence" (websites promoting violence) [4]. Because websites are categorized in the above categories using an automated process (due to the vast number of websites on the Internet), a website can be ranked incorrectly. For this reason, every

educational organization follows international practice. It enables its users to inform the competent technicians when they find any malfunction of the service, who now manually correct the database that should be excluded.

In addition, social media giants enforce strict policies and established procedures for dealing with content and any harmful behavior, prohibiting content that endangers minors [5]. These include sexual harassment, abuse, and harmful and dangerous acts, uploading, streaming, commenting, engaging in activities that harm children, etc. Also, in recent years, these companies have become significant investors in the design of systems that detect sexually explicit material on the video clearly and effectively to prevent the release of material with unacceptable content [6].

The huge unresolved issue now is in cases of intentional or unintentional exposure to sexual content when using real-time video conferencing software, such as online video conferencing software, used extensively during the pandemic. In these cases, where content and streaming occur in real-time, it is challenging to detect obscene or sexual content, so there is no protection for underage students [7].

Obscene and primarily sexual content can be detected in streaming online video conferencing software for education with great precision. Based on the gap presented in the procedures and minors' risks, mainly students, this paper proposes an innovative deep attention neural architecture system to tackle explicit material from online education video conference applications. It is an advanced machine learning technique, precisely computer vision, which uses an intelligent attention mechanism that does not have a square of memory and time complexity in terms of the size of its input data. Specifically, the implementation uses a GAN, with the help of a local, sparse attention mechanism, of complexity $O(n\sqrt{n})$. We take advantage of the probability distributions generated within this particular attention mechanism while maintaining the 2d geometry of the multimedia content.

2. Related Literature

The literature concerning the field of detection mechanisms concerning specific or explicit content [1, 3] is varying due to the different approaches that the research community has:

Li et al. [8] studied numerous motion classification algorithms, concentrating on video using classifiers, mostly frame-based. They divided the basic processes into three main categories: the first was frame-by-frame recognition, the second was extracting sequences, and the third was temporal-information monitoring, which used the LSTM structure or the optical flow approach to remove training data between sequences. They also divided and characterized the various types of deep learning-based cameras as follows: Convolutional Neural Networks-based methods, Restricted Boltzmann Machine-based methods [9], and Autoencoder-based techniques, all examples of unsupervised ML algorithms that could acquire the representations and produce data frames with similar attributes.

Longlong et al. [10] looked at self-supervised generic image learning techniques based on deep learning from media files. They defined the key terms and examined the most prevalent self-supervised learning deep neural network topologies. They next looked at the architecture and evaluation criteria for self-supervised learning techniques, as well as the most often used samples, primarily for videos, and current self-supervised visual feature learning techniques [11]. They examined the practices of the shapes on image and video feature learning benchmark datasets. They finished their proposal by outlining several potential avenues of development for self-supervised visual feature learning.

Arachchi et al. [12] introduced a state-exchanging long short-term memory (SE-LSTM) two-stream neural network approach, based on the benefits of using spatial and motion information to identify dynamic patterns. This method was used to identify movie reactions using appearance motion characteristics. It could also be used to expand the general purpose of LSTM by sharing data with past cell states in both the look and action streams. The movies could not include any other active items than the target objects to achieve better classification performance, and the contexts had to be static [13]. The trial findings showed that the technique surpassed other collections in precision, particularly when it came to static background dynamic patterns classifications. To decrease discrepancies, they proposed eliminating all mislabeled information in the next round of their study.

Duboyskii et al. [14] used automated emotional state recognition and video conferencing technologies to transmit distant material in travel communication systems, surveys, and other applications. They created a peer-to-peer framework for remote communication sessions, allowing clients to share audio and visual information. At the operator end, convolutional neural networks were used for stream processing and to evaluate the customer's emotional responses. Three mechanisms (video, audio, and text) and multimodal recognition were employed to establish the dynamic conditions. The test was carried out between persons in which one served as an operator and posed closed questions while the other answered them. The proposed technology could be used in various sectors, including service delivery and healthcare, where real-time human emotion identification is essential. The neural network produced the highest accuracy values when multimodal recognition was applied, indicating its effectiveness in video conferencing systems for classifying human emotions. Their system had the disadvantage of only supporting one-to-one user connections, which they plan to address by expanding the number of concurrent user connections.

In 2016, Vondrick et al. [15] introduced a generative adversarial network for films using a Spatio-temporal convolutional architecture that untangled the scene's images by investigating how to learn behaviors from vast volumes of unstructured camera footage. It is expected that the scene dynamics will be critical for the next phase of computer vision systems and learning from unlabeled data would be a promising option. Tests and simulations revealed that the model recognized important aspects for detecting actions with little control on the inside. Despite the fact that fully realizing the potential of unlabeled video is still a work in progress, their findings suggest that having a lot of

unsupervised videos might be beneficial for both training to create films and acquire graphical images.

Tulyakov et al. [16] proposed the Motion and Content deconstructed Generative Adversarial Network (MoCoGAN) framework for motion and content decomposed video production using the Generative Adversarial Network. In an unsupervised fashion, the MoCoGAN was trained to distinguish signal from content, and a movie was created by mapping a set of random vectors to a set of image sequences. They presented a unique adversarial learning method that learned motion and content decomposition unsupervised using both image and video discriminators. A Gaussian distribution was used to describe the content subspace, while a recurrent neural network to model the motion domain. The efficiency of the suggested framework was confirmed by experimental findings on datasets with qualitative and quantitative comparisons to state-of-the-art techniques [17]. They also demonstrated how their scheme could be used to produce videos with the same material but distinct motion, as well as films.

To overcome the short sample issue in hyperspectral image classification, Feng et al. [18] presented a symmetric convolutional GAN based on collaborative learning and attention mechanism (CA-GAN). A combined spatial-spectral intricate attention module was used in the Generator to filter out misleading and confusing aspects of the produced samples and force the distribution of generated models to resemble the pattern of genuine hyperspectral images. To retrieve combined spatial-spectral information of images, a convolutional LSTM layer was fused in the Discriminator. In addition, by using the actual sample information retrieved by the Discriminator, a collaborative learning process was devised to aid sample production in the generator. It allowed the Generator and Discriminator to be refined alternately and collaboratively via competition. Tests on noteworthy sources of data revealed that their method outperformed the other approaches in terms of classification accuracy, particularly when the number of training samples was restricted. The studies indicated that they will look into more efficiently and automatically determining the placements and numbers of different modules, and they will experiment with different sampling methods to eliminate overlap between training and testing sets.

From the literature mentioned, we see that the research community is actively focusing on finding methods and techniques to increase the performance of media classification, according to the specific needs of each individual Case [3, 19].

3. Methodology

The proposed implementation is based on the GANs architecture [18, 20], which uses an optimal local, sparse attention mechanism. Using a previous frame's context, a video prediction algorithm can foretell the next frame in a video. Unlike a static image, a video allows the viewer to see the changes and motion patterns over a more extended period. For this reason, the model must take into account both time and space to accurately predict the future frames in a video. Modeling temporal dynamics is typically done using Recurrent Neural Networks. However, GANs have become the most popular method for predicting future video frames. A vital element of the structure of GANs is the existence and simultaneous training of two networks, the Generator that creates samples as close as possible to those of the training set and the Discriminator that is trained to distinguish which samples come from the training set (i.e., are they real) and which one from the Generator (i.e., are they artificial or fake). Specifically, at each training Step (i.e., inside the training loop), the Discriminator receives samples from the training data set and samples generated by the Generator and is trained to have a probability of close to 1 for the first and close to 0 for the second. In contrast, the Generator is trained so that from input noise to output images to the output realistic enough to "trick" the Discriminator.

Going a little deeper into the analysis of how GANs are trained, we can say that both Generator and Discriminator are represented by (continuously) differentiable functions with trainable parameters, such as neural networks, each with its cost function. The two networks are trained through back-propagation using the Discriminator cost function, but with a different goal. The Discriminator tries to reduce the cost function for both natural and artificial samples, while Generator tries to increase the Discriminator cost function for the synthetic samples it produces. It is noteworthy that the training data set alone determines the type of samples that the Generator learns to create.

The Binary Cross-Entropy cost function is used in the proposed methodology. For each predicted probability, Binary Cross Entropy compares it to the class output of 0 or 1. Once the score has been calculated, probabilities are penalized based on the distance from the expected value. This is a measure of how close or how far the calculated value is from the actual value. Specifically, for a set of m samples per batch is as follows [17]:

$$J_m 23(\vec{\partial}) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(h\left(x^{(i)};\vec{\partial}\right)\right)+\left(1-y^{(i)}\right)\log\left(1-h\left(x^{(i)};\vec{\partial}\right)\right)\right], \quad (1)$$

where the initial sum and division by the number of samples approximates the mean value operator, $x(i)$ is the i-th sample, $y(i)$ is the label of the i-th sample, and $\sim\theta$ is the vector of the trainable model parameters. During the Discriminator training of the proposed GAN, the labels will be 1 for the actual samples and 0 for the artificial ones. In contrast, for the training of the Generator, the reverse is true, i.e., together with the synthetic samples, label 1 will be given to calculate whether it may "trick" the Discriminator.

Focusing on the formation of the cost function and the values it receives for the 0/1 tags given during the training of a GAN, we see that when the tag is 1, only the first term of the sum acts. Considering the negative sign at the beginning of the equation, we see that the above Binary Cross-Entropy approach for a batch takes values from 0 to $+\infty$ when the classification function $h(x)$ with parameters θ takes values from 0 to 1.

Optionally, the Binary Cross Entropy cost function has two parts (one for each class) and takes values close to 0 for correct configuration (diagonal confusion matrix) while approaching the positive infinity for error (diagonal confusion matrix) - behavior graphically illustrated in Figure 1 below [17]:

Thus, for the Discriminator, the Binary Cross-Entropy cost function given that during GAN training, the actual data is contractually assigned the tag 1 and the artificial data to 0, will be [17]:

$$\begin{aligned} J_D\left(\overrightarrow{\partial_D}, \overrightarrow{\partial_G}\right) &= -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(h\left(x^{(i)}; \overrightarrow{\partial}\right)\right) + \left(1 - y^{(i)}\right)\log\left(1 - h\left(x^{(i)}; \overrightarrow{\partial}\right)\right)\right] \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left[\log\left(D\left(x^{(i)}; \overrightarrow{\partial_D}\right)\right) + \log\left(1 - D\left(G\left(\overrightarrow{z}^{(i)}; \overrightarrow{\partial_G}\right); \overrightarrow{\partial_D}\right)\right)\right] \\ &= -\frac{1}{m}\sum_{i=1}^{m}\log\left(D\left(x^{(i)}; \overrightarrow{\partial_D}\right)\right) - \frac{1}{m}\sum_{i=1}^{m}\log\left(1 - D\left(G\left(\overrightarrow{z}^{(i)}; \overrightarrow{\partial_G}\right); \overrightarrow{\partial_D}\right)\right) \\ &\approx -\mathbb{E}_{x\sim p_{\text{data}}}\log[D(x)] - \mathbb{E}_{z\sim p_{\text{prior}}}\log[1 - D(G(z))], \end{aligned} \tag{2}$$

where $D(x)$ is the Discriminator output (i.e., the probability of realism of the input x), $G(z)$ is the output of the Generator network for random vector input z (i.e., an artificial image), pdata is the distribution followed by the data input (in these images it will be a very high dimensional distribution that can only be indirectly and approximately modeled by GAN), and pprior the prior distribution from which we sample to get the random vector at the Generator input. Since Discriminator predicts probability and therefore $D(x) \in [0, 1]$, it follows that to minimize its cost function, Discriminator must learn to assign a high probability to samples labeled 1 (derived from the set of training data) and low on those generated by the Generator [21].

The Generator network, in turn, tries to "trick" the Discriminator so that the chances it assigns to the artificial samples at its output are high. It aims to maximize the second term of the Discriminator cost function - after all, only this term can affect the Discriminator's cost function to increase it. Therefore, the following will apply to the Generator [8]:

$$\begin{aligned} J_G\left(\overrightarrow{\partial_G}, \overrightarrow{\partial_D}\right) &= \frac{1}{m}\sum_{i=1}^{m}\left[\left(1 - y^{(i)}\right)\log\left(1 - h\left(x^{(i)}; \overrightarrow{\partial}\right)\right)\right] \\ &= \frac{1}{m}\sum_{i=1}^{m}\left[\log\left(1 - D\left(G\left(\overrightarrow{z}^{(i)}; \overrightarrow{\partial_G}\right); \overrightarrow{\partial_D}\right)\right)\right] \\ &= \frac{1}{m}\sum_{i=1}^{m}\log\left(1 - D\left(G\left(\overrightarrow{z}^{(i)}; \overrightarrow{\partial_G}\right); \overrightarrow{\partial_D}\right)\right) \\ &\approx \mathbb{E}_{z\sim p_{\text{prior}}}\log[1 - D(G(z))], \end{aligned} \tag{3}$$

where the negative sign at the beginning has now been removed as the Generator tries by minimizing its cost function to increase that of the Discriminator, while all other sizes are as before. Because the first term of the equation depends only on the training data set, the above Generator cost function is declared as negative of the Discriminator cost function [22, 23]:

$$J_G\left(\overrightarrow{\partial_G}, \overrightarrow{\partial_C}\right) = -J_D\left(\overrightarrow{\partial_C}, \overrightarrow{\partial_G}\right). \tag{4}$$

Focusing on the continuous 1-Lipschitz function f, in the proposed GAN is the Discriminator network itself, which, taking an image, x, is called upon to give a real number. Therefore, the function will be

$$c\colon X \longrightarrow \mathbb{R}, \|c\|_L \leq 1. \tag{5}$$

To successfully approach a neural network with trainable parameters $\sim\theta$ a continuous 1-Lipschitz function, the measure of some of the network output derivatives in terms of trainable parameters must be at most 1 at each point in the domain. Thus, the Discriminator neural network must satisfy the following continuity condition to be a 1-Lipschitz continuous function [24, 25]:

$$\nabla_{\widehat{\partial_C}} C\left(x; \overrightarrow{\partial_C}\right)_2 \leq 1 \forall x \in \mathcal{X} \leftrightarrow \|c\|_L \leq 1. \tag{6}$$

This condition enforcement ensures that the cost function is valid when measuring the allocation distance. It is continuous and differentiable and does not increase too fast. The proposed model introduces a normalization term that imposes a penalty when the norm of some of the output derivatives of Discriminator concerning its input is greater than 1 so that [21]

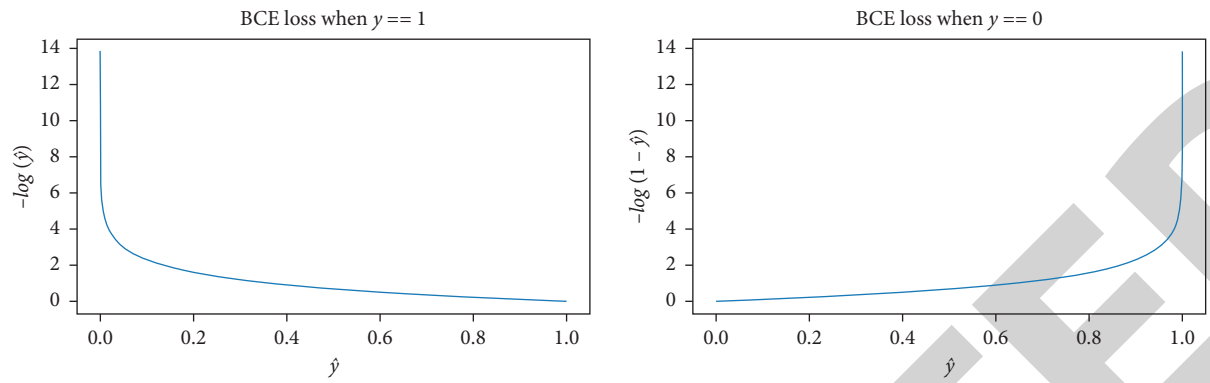


Figure 1: The binary Cross-Entropy loss function for real (left) and fake (right) data.

$$\mathrm{reg}_{GP} = \left(\nabla_{\hat{\partial}_C} C(\tilde{x};\tilde{x})_2 - 1\right)^2, \tag{7}$$

and so, the cost functions that the two neural networks try to minimize will be [8, 21, 22]:

$$\begin{aligned} J_C\left(\overrightarrow{\partial_C}, \overrightarrow{\partial_G}\right) &= -\frac{1}{m}\sum_{i=1}^{m}\left[C\left(x^{(i)};\overrightarrow{\partial_C}\right)\right] + \frac{1}{m}\sum_{i=1}^{m}\left[C\left(G\left(\overrightarrow{z}^{(i)};\overrightarrow{\partial_G}\right);\overrightarrow{\partial_C}\right)\right] + \hat{\lambda}_{GP} * reg_{GP} \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left[C\left(x^{(i)};\overrightarrow{\partial_C}\right)\right] + \frac{1}{m}\sum_{i=1}^{m}\left[C\left(G\left(\overrightarrow{z}^{(i)};\overrightarrow{\partial_G}\right);\overrightarrow{\partial_C}\right)\right] \\ &\quad + \lambda_{GP} * \frac{1}{m}\sum_{i=1}^{m}\left[\left(\nabla_{\overrightarrow{\partial_C}} C\left(\varepsilon * x + (1-\varepsilon) * \hat{x};\overrightarrow{\partial_C}\right)_2 - 1\right)^2\right] \\ &\approx -\mathbb{E}_{x\sim p_{\mathrm{data}}}[C(x)] + \mathbb{E}_{z\sim p_{\mathrm{priar}}}[C(G(z))] + \hat{\lambda}_{GP} * \mathbb{E}_{\tilde{x}\sim p_{\tilde{x}}}\left[\left(\nabla_{\tilde{x}} C(\tilde{x})_2 - 1\right)^2\right]. \end{aligned} \tag{8}$$

To model the sequence of input symbols under a single framework, we propose in this work the use of optimal attention mechanisms both qualitatively and computationally. The proposed sparse attention mechanism requires much less memory, is faster, achieves better performance, and requires fewer training steps than intensive attention due to incorporating appropriate assumptions into its architectural design.

In particular, the quadratic complexity of attention is due to the calculation of the table [17, 23]:

$$M_{Q,K} = Q \cdot K^T, \ \in \mathbb{R}^{N_X \times N_Y}. \tag{9}$$

Instead, we propose multidimensional attention mechanisms in this work. In each Step i, attention is limited to a set of predefined positions given by a mask:

$$A_i \in \{0,1\}^{N_X \times N_Y}. \tag{10}$$

In each Step i, we calculate

$$M^i_{Q,K}[a,b] = \begin{cases} M_{Q,K}[a,b], A^i[a,b] = 1, \\ -\infty, A^i[a,b] = 0. \end{cases} \tag{11}$$

In addition, using information flow charts and the two-dimensional geometry conservation mechanism, we construct a sparse multistep attention layer that can model any dependencies on the input data and respects the native pixel locality in a video. An indicative representation of spherical 2-D points far away from the sphere is very unlikely to fall in the same area at all random rotations, which is reversed for very close points to the sphere, as shown in Figure 2.

This process is directly related to the tendency of the softmax function to yield sparse distributions. So, by this logic, we argue that the dense models produce sparse attention maps:

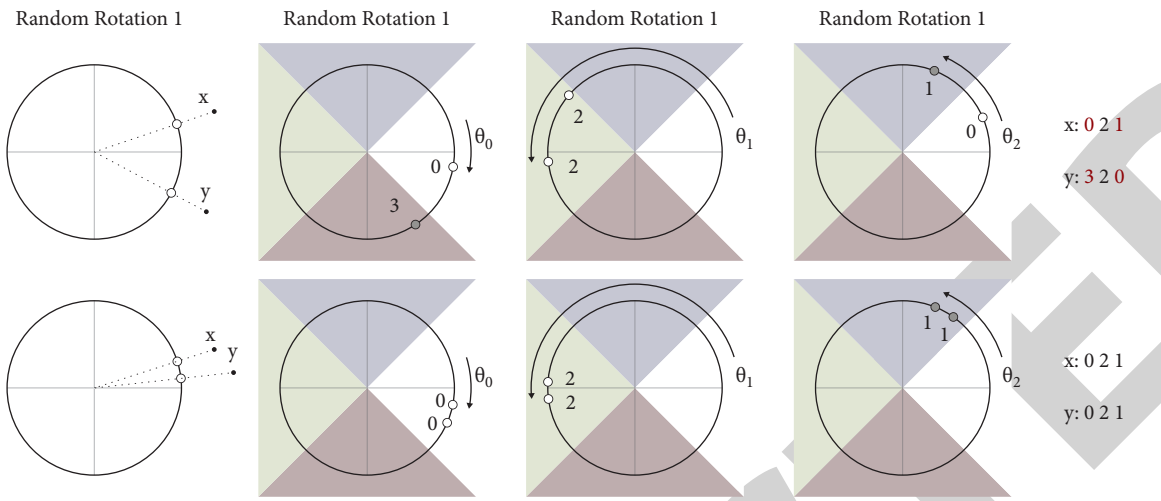


Figure 2: Spherical 2-D case study.

$$E\left(X_{k:n}\right) \le \mu + \sigma\sqrt{\frac{k-1}{n-k+1}}. \tag{12}$$

Based on the above relation, we can prove the rarity of the probabilistic distributions obtained from softmax [21–23]:

$$\frac{e^{E\left(X_{k:n}\right)}}{\sum_{i=1}^{n} e^{E\left(X_i\right)}} \le \frac{e^{\mu+\sigma\sqrt{k-1/n-k+1}}}{\sum_{i=1}^{n} e^{E\left(X_i\right)}} = \frac{e^{\mu+\sigma\sqrt{k-1/n-k+1}}}{ne^{\mu}}. \tag{13}$$

which with limits $\mu = 0$ and $\sigma 2 = 1$ can be calculated as

$$E\left(\frac{e^{X_{k:n}}}{\sum_{i=1}^{n} e^{X_{i:n}}}\right) \le \epsilon,$$

$$\frac{e^{\sqrt{k-1/n-k+1}}}{n} \le \epsilon \overset{n\,\epsilon\,\ge 1}{\rightleftarrows} \sqrt{\frac{k-1}{n-k+1}} \le ln(n\epsilon)k \le \frac{1+(n+1)\ln^2(n\epsilon)}{1+\ln^2(n\epsilon)}. \tag{14}$$

The challenge in multistep attention mechanisms is the design of dual masks for each step. This paper uses an information theory tool to successfully design sparse attention patterns. Specifically, information flow graphs are used, which are guided, acyclic graphs that model the flow of network information into graphs of distributed systems. For our problem, these graphs show the flow of information between the attention steps and the corresponding transformations that follow. The most common of the proposed transformations are [8, 22, 23]:

$$\begin{cases} F\left(q_i\right) = \left[q_i; \frac{1}{2}, \ldots; \frac{1}{2}\right], G\left(k_i\right) = \left[Uk_i; Uk_{i2}^2; \ldots; Uk_{i2}^2\right], \\ F\left(q_i\right) = \left[q_i; 0\right], G\left(k_i\right) = \left[k_i; \sqrt{M_K^2 - k_{i2}^2}\right], \\ F\left(q\right) = \frac{M_K}{\|q\|_2} \cdot \left[q; 0\right], G\left(k\right) = \left[k; \sqrt{M_K^2 - \|k\|_2^2}\right]. \end{cases} \tag{15}$$

To smooth out the deformations resulting from the above transformations, the proposed system allows the focus on the previous and next stage, as shown in Figure 3 below:

For each set of masks $\{A^1, \ldots, A^p\}$ we make a polymer graph $G(V = \{V^0, V^1, \ldots, V^p\}, E)$ where the edges between V^i, V^{i+1} are determined by the mask Mi. Thus, a sparse pattern has complete information if the relevant information graph has a path from each node a ∈ V0 to each node $b \in$ Vp. So, in addition to the computational improvement of the dense attention mechanism, the sparse attention mechanisms also achieve better results due to the integration of prior knowledge of locality into the information flow chart.

Our mechanism has $O(n\sqrt{n})$ memory complexity and speed, significantly reducing the square complexity of intensive attention. The probability distributions created within the attention map make a new method for reversing the proposed attention GAN. Essentially, the proposed technique provides the methodology for evaluating the boundaries of indeterminate forms so that by applying them, an indefinite form can be quickly assessed by substitution [21, 22]:

$$\begin{aligned} \lim_{h\longrightarrow 0^+} \frac{1}{h}\int_{x_j}^{x_j+h} f\left(x|\theta\right)\mathrm{d}x &= \lim_{h\longrightarrow 0^+} \frac{d/\mathrm{d}h \int_{x_j}^{x_j+h} f\left(x|\theta\right)\mathrm{d}x}{\mathrm{d}h/\mathrm{d}h} \\ &= \lim_{h\longrightarrow 0^+} \frac{f\left(x_j+h|\theta\right)}{1} \\ &= f\left(x_j|\theta\right). \end{aligned} \tag{16}$$

Then,

$$\begin{aligned} \mathrm{argmax}_\theta \mathscr{L}\left(\theta|x_j\right) &= \mathrm{argmax}_\theta\left[\lim_{h\longrightarrow 0^+} \mathscr{L}\left(\theta|x \in \left[x_j, x_j+h\right]\right)\right] \\ &= \mathrm{argmax}_\theta\left[\lim_{h\longrightarrow 0^+} \frac{1}{h}\int_{x_j}^{x_j+h} f\left(x|\theta\right)\mathrm{d}x\right] \\ &= \mathrm{argmax}_\theta f\left(x_j|\theta\right). \end{aligned} \tag{17}$$

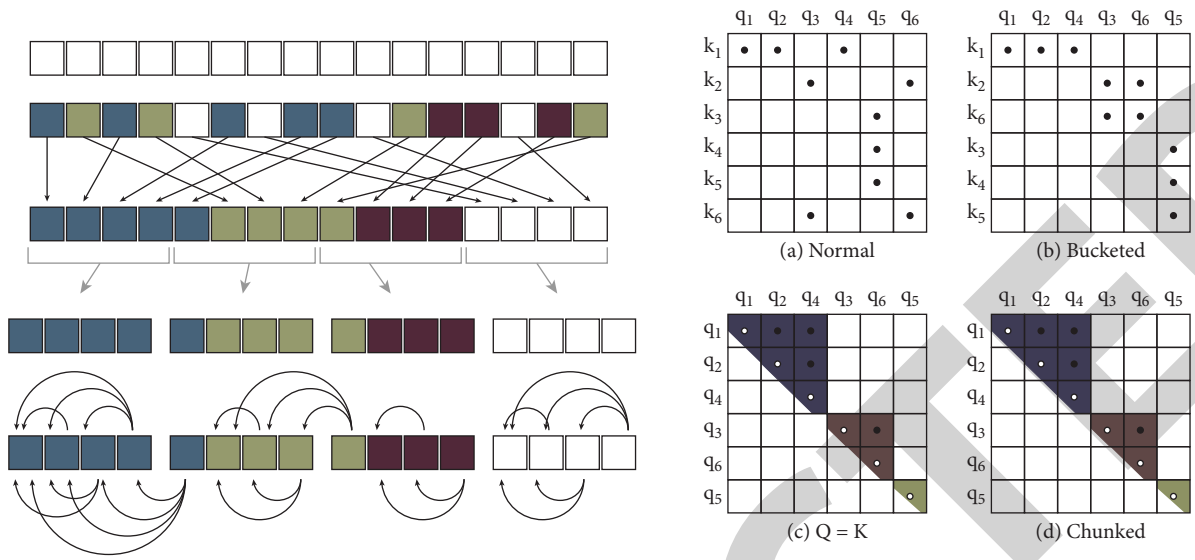


Figure 3: Depiction of transformer attention.

Therefore,

$$\text{argmax}_\theta \mathscr{L}\left(\theta|x_j\right) = \text{argmax}_\theta f\left(x_j|\theta\right), \tag{18}$$

and so, maximizing the probability density in x_j equals maximizing the probability of that observation in x_j, thus creating the method of the proposed attention.

As a novel approach, this technique is an intelligent advanced mechanism that uses attention mechanisms but does not have a square complexity of memory and time in terms of the input size. So, it is possible to accurately detect obscene and primarily sexual content in streaming online video conferencing software.

4. Scenarios and Results

The research was also conducted to assess the likelihood that the user will engage in abnormal behavior related to displaying inappropriate content [7, 26]. A specialized scenario was implemented to model the proposed system to calibrate the user's actions during the live video stream about an activity that might be considered provocative or inappropriate. This process was based on the technique of visual flow, which involves the movement of objects between successive snapshots of a video, which arises due to the action of objects. Sparse optical flow detects characteristic points, such as angles and edges of the image, and their monitoring in successive snapshots, while dense visual flow refers to the estimation of the motion vectors of the whole image, i.e., all pixels.

More specifically, the scenario assumes that the optical flux is a standard estimate where the position of each point is defined using a square polynomial of the form $f_1(x) = x^T A_1 x + b^T x + c_1$, where A is a symmetric array, b vector, and c graded number. An adjustment of least squares determines the coefficients. Respectively for the second scene, it applies that [27–29]:

$$f_2(\mathrm{x}) = f_1(\mathrm{x} - \mathrm{d}). \tag{19}$$

Therefore, we have

$$\begin{aligned} f_2(x) &= f_1(x - d) \\ &= (x - d)^T \mathrm{A}_1 (x - d) + \mathrm{b}_1^T (x - d) + c_1 \\ &= \mathrm{x}^T \mathrm{A}_1 x + (\mathrm{b}_1 - 2\mathrm{A}_1 \mathrm{d})^T x + \mathrm{d}^T \mathrm{A}_1 d - \mathrm{b}_1^T d + c_1 \\ &= \mathrm{x}^T \mathrm{A}_2 x + \mathrm{b}_2^T x + c_2. \end{aligned} \tag{20}$$

If the coefficients of the square polynomials are equated, we have

$$\begin{aligned} &\mathrm{A}_2 \mathrm{A}_1, \\ &\mathrm{b}_2 = \mathrm{b}_1 - 2\mathrm{A}_1 d, \\ &c_2 = \mathrm{d}^T \mathrm{A}_1 d - \mathrm{b}_1^T d + c_1. \end{aligned} \tag{21}$$

And since A is reversible, we have

$$\mathrm{d} = -\frac{1}{2}\mathrm{A}_1^{-1}(\mathrm{b}_2 - \mathrm{b}_1). \tag{22}$$

This condition does not apply to the entire image signal, as there is no universal permutation. Thus, the universal polynomial equation is converted to local with coefficients $A1(x)$, $b1(x)$, and $c1(x)$. Even the condition $A1 = \mathrm{A}2$ is practically not valid, so it is estimated as [27, 28]

$$\mathrm{A}(\mathrm{x}) = \frac{\mathrm{A}_1(\mathrm{x}) + \mathrm{A}_2(\mathrm{x})}{2}. \tag{23}$$

Finally, we define

$$\Delta \mathrm{bb}(\mathrm{x}) = -\frac{1}{2}\left(\mathrm{b}_2(\mathrm{x}) - \mathrm{b}_1(\mathrm{x})\right). \tag{24}$$

We have

$$\mathrm{A}(\mathrm{x})\mathrm{d}(\mathrm{x}) = \Delta \mathrm{b}(\mathrm{x}), \tag{25}$$

where $d(x)$ now has local power and is not universal. Finally, to improve the accuracy, we can apply this condition to the whole neighboring area and not to each pixel separately, minimizing the relationship [13, 23, 26]:

$$\sum_{\Delta x \in I} w(\Delta x)\|\mathrm{A}(\mathrm{x}+\Delta \mathrm{x})\mathrm{d}(\mathrm{x}) - \Delta \mathrm{b}(\mathrm{x}+\Delta \mathrm{x})\|^2, \tag{26}$$

where $\mathrm{w}(\Delta x)$, weight function of adjacent points. So, the field of view is ultimately

$$\mathrm{d}(\mathrm{x}) = \left(\sum w\mathrm{A}^T A\right)^{-1} \sum w\mathrm{A}^T \mathrm{b}. \tag{27}$$

So, the algorithm's operation is based on the minimization of a function that includes an information term using the L1 norm and a normalization term using the optical fluctuation. Brightness constancy assumption is initially considered as

$$\frac{d}{dt} I(x(t), y(t), t) = 0, \tag{28}$$

where $I(x(t), y(t), t)$ the video and $(x(t), y(t))$ the trajectory of a point in the image. Applying the chain rule results in

$$\nabla I \cdot (\dot{x}, \dot{y}) + \frac{\partial}{\partial t} I = 0. \tag{29}$$

It is also defined as the speed of the orbits:

$$\mathrm{u}(x, y) = (u_1(x, y), u_2(x, y)), \tag{30}$$

and the visual flow is committed to locating the reference point, which in the resulting case is the inappropriate material:

$$\nabla I \cdot u + \frac{\partial}{\partial t} I = 0. \tag{31}$$

For each point in the image, this equation has 2 unknown variables, the velocity components u. Therefore, the system does not have a unique solution. To solve this problem, we use a smoothing term to force the normalization of u.

In the proposed model, the solution is performed by minimizing the energy function resulting from the sum of the variability of u and the term L1 when the following function is applied [13, 17, 21]:

$$E(\mathrm{u}) = \int_\Omega |\nabla u_1| + |\nabla u_2| + \lambda|\rho(\mathrm{u})|. \tag{32}$$

The minimization process for finding u is performed for different image scales. The vector u is initially calculated for large scales, initial values for the more minor scales. Thus, the vector u is gradually determined more accurately.

Finally, for the proposed algorithm to better render the classification coded features, the Gaussian Mixture Model (GMM) is first calculated to model the distributions of video descriptions. The vectors then encode the slope of the logarithmic probability of the features according to the GMM parameters. Let $\mathrm{X} = \{x1, x2, xt\}$ the n-dimensional features. The GMM parameters are estimated based on these characteristics: weights, averages, and variability. Accordingly, the logarithmic probability slopes for the GMM parameters are calculated as follows [11, 30, 31]:

$$\begin{aligned} \nabla_{\alpha_k} \log p(X) &= \sum_{i=1}^{t} \nabla_{\alpha_k} \log p(x_t), \\ \nabla_{\mu_k} \log p(X) &= \sum_{i=1}^{t} \nabla_{\mu_k} \log p(x_t), \\ \nabla_{\sigma_k} \log p(X) &= \sum_{i=1}^{t} \nabla_{\sigma_k} \log p(x_t), \end{aligned} \tag{33}$$

where from the sum of the three vectors results [31, 32]:

$$FV = \left[\nabla_{\alpha_k} \log p(X), \nabla_{\mu_k} \log p(X), \nabla_{\sigma_k} \log p(X)\right]. \tag{34}$$

The pornography database [4, 6, 19, 30], which contains nearly 80 hours of 400 pornographic and 400 non-pornographic videos, was used to locate the scenes of inappropriate material. The pornographic material comes from relevant sites that host only such material. At the same time, it should be emphasized that the set consists of various types of pornography and depicts actors of many ethnicities. Respectively, the non-pornographic content came from browsing the web with general-purpose videos.

During pre-processing, all videos were initially segmented into shots. A basic (non-inappropriate) frame was used to summarize the content of the picture into a still image. Some typical static images from photos contained in this dataset are shown in Figure 4 below [1, 5, 30].

All the exterior shots, such as beach shots, were removed, and only indoor pictures were used. In total, 12,182 videos were used, of which 6,743 were inappropriate, and 5,439 were inappropriate.

The video observations based on the density estimation were given in time-series images, where the x-axis symbolizes time. In probability and statistics, density estimation is constructing an estimate of an unobservable underlying probability density function using observed data. The unobservable density function describes the distribution of a vast population; the data are typically viewed as a random sampling from that population. Density estimation techniques such as Parzen windows and various data clustering techniques, including vector quantization, are used. The simplest method for estimating density is to use a rescaled histogram. In this paper, for uniformity and comparison of the results, along with the pictures of the model estimation, a heuristic method was used based on the images of the experts' observations and their votes in terms of content for each scene. Models trained with batch learning in the material in question were used as specialists. This procedure was done for each video, based on the total time in seconds that each category lasted within the video [6, 30].

The 10-fold cross-validation method was used for the experimental evaluation. In contrast, the Mean Average Precision (MAP) and Accuracy Rate (AcR) were used as the scoring measure, where most evaluators take the final class of the examined video. Finally, the ROC Curve and F-measure metrics displayed the results. The results of the procedure are shown in Table 1 below.

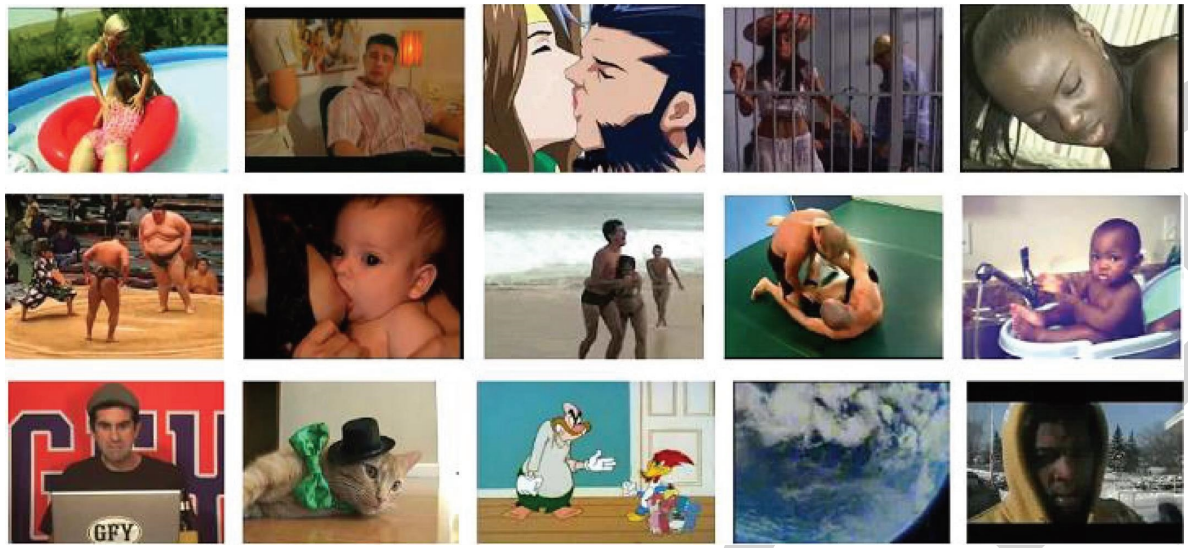


Figure 4: Pornography database.

Table 1: Performance metrics of the classification process - 1.

	MAP	AcR	ROC curve	F-measure
Porn	92.532	91.912	95.004	92.301
noPorn	88.024	89.031	88.558	89.065

Table 2: Performance metrics of the classification process - 2.

	MAP	AcR	ROC curve	F-measure
Porn	95.871	95.568	95.997	96.163
noPorn	92.958	91.597	91.869	91.733

Table 3: Performance metrics of the classification process - 3.

	MAP	AcR	ROC curve	F-measure
Porn	99.220	99.118	99.672	99.258
noPorn	98.487	98.596	98.604	98.599

As can be seen from the table above, the results look pretty satisfactory. In some cases, the model finds it challenging to locate the noPorn category, slightly reducing its overall performance. This is because the vector representations are identical. Although experimentally, this did not reduce the performance for the problems tested, there may be other problems with a drop. Even more importantly, this limits its use to situations where the number of classes is multiple.

For this problem, a simple solution was used to replace the imaging function to group vectors with short Euclidean distances or large internal products to have data located in some lower norm sphere or even data without geometric constraints. The results of the procedure are shown in Table 2 below.

As can be seen, alternative display schemes achieve much better results without imposing such strong constraints on the nature of the input data. The main problem of the proposed solution is that it requires network retraining and, therefore, cannot operate on pretrained networks. This significantly reduces its usefulness as retraining costs are vast, and the chances of mastering sparse attention mechanisms are slim.

The groups are created randomly in random attention, and the attention occurs within the group. To increase the probability of success of the method, we repeat the process a few times. For this reason, we propose a comparison model, the randomization, which can be used to create sparse models that do not require retraining. As shown in Table 3 below, the model in question achieves impressive results.

It seems that this model can begin the search to find attention mechanisms that do not require retraining.

5. Conclusions

In this work, we proposed and studied solutions for efficient attention mechanisms. The methods presented are based on either predetermined sparse patterns or dynamic dilution. The advanced technique first introduced in the literature suggests a GAN assisted by attention mechanisms, which can speed up and even be more efficient, allowing for faster processing and fewer memory requirements. The methodology is used in a case study to deal with incidents of fair or unfair exposure to offshore content to underage students during distance learning in online education video conference applications. A significant disadvantage of the proposed method is that it requires an extensive bandwidth network.

Changes that can lead to simpler variants of attention that operate without imposing restrictions on attention inputs are critical future developments in this work. Also, the search for even more efficient computing methods and, in general, the solutions that can significantly improve the performance of solving complex real-time problems like the one studied. Finally, it is crucial to investigate how an external classification scheme can be implemented that can achieve high acceleration for a sufficiently large input size.

Data Availability

The data are available at https://sites.google.com/site/pornographydatabase/

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] A. Gangwar, E. Fidalgo, E. Alegre, and V. González-Castro, "Pornography and child sexual abuse detection in image and video: a comparative evaluation," in *Proceedings of the 8th International Conference on Imaging for Crime Detection and Prevention (ICDP 2017)*, pp. 37–42, Madrid, Spain, December 2017.

[2] M. Moustafa, "Applying Deep Learning to Classify Pornographic Images and Videos," 2015, https://arxiv.org/abs/1511.08899.

[3] M. Perez, S. Avila, D. Moreira et al., "Video pornography detection through deep learning techniques and motion information," *Neurocomputing*, vol. 230, pp. 279–293, 2017.

[4] J. Wehrmann, G. S. Simões, R. C. Barros, and V. F. Cavalcante, "Adult content detection in videos with convolutional and recurrent neural networks," *Neurocomputing*, vol. 272, pp. 432–438, 2018.

[5] P. Vitorino, S. Avila, M. Perez, and A. Rocha, "Leveraging deep neural networks to fight child pornography in the age of social media," *Journal of Visual Communication and Image Representation*, vol. 50, pp. 303–313, 2018.

[6] X. Ou, H. Ling, H. Yu, P. Li, F. Zou, and S. Liu, "Adult image and video recognition by a deep multicontext network and fine-to-coarse strategy," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 5, pp. 1–25, 2017.

[7] K. Yuan, D. Tang, X. Liao et al., "Stealthy porn: understanding real-world adversarial images for illicit online promotion," *2019 IEEE Symposium on Security and Privacy (SP)*, in *Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP)*, pp. 952–966, San Francisco, CA, USA, May 2019.

[8] D. Li, R. Wang, P. Chen, C. Xie, Q. Zhou, and X. Jia, "Visual feature learning on video object and human action detection: a systematic review," *Micromachines*, vol. 13, no. 1, p. 72, 2021.

[9] L. Xing, K. Demertzis, and J. Yang, "Identifying data streams anomalies by evolving spiking restricted Boltzmann machines," *Neural Computing & Applications*, vol. 32, no. 11, pp. 6699–6713, 2020.

[10] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4037–4058, 2021.

[11] S. T. Ali, K. Goyal, and J. Singhai, "Moving object detection using self adaptive Gaussian Mixture Model for real time applications," in *Proceedings of the 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, pp. 153–156, Bhopal, India, October 2017.

[12] S. P. Kasthuri Arachchi, T. K. Shih, and N. L. Hakim, "Modelling a spatial-motion deep learning framework to classify dynamic patterns of videos," *Applied Sciences*, vol. 10, no. 4, p. 1479, 2020.

[13] B. Vishwanath and K. Rose, "Spherical video coding with geometry and region adaptive transform domain temporal prediction," in *Proceedings of the ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2043–2047, Barcelona, Spain, May 2020.

[14] I. Duboyskii, A. Shabanova, O. Sivchenko, and E. Usina, "Architecture of cross-platform videoconferencing system with automatic recognition of user emotions," *IOP Conference Series: Materials Science and Engineering*, vol. 918, no. 1, Article ID 012086, 2020.

[15] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[16] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "MoCoGAN: Decomposing Motion and Content for Video Generation," 2017, https://arxiv.org/abs/1707.04993.

[17] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, Cambridge, MA, USA, 2012.

[18] J. Feng, X. Feng, J. Chen et al., "Generative adversarial networks based on collaborative learning and attention mechanism for hyperspectral image classification," *Remote Sensing*, vol. 12, no. 7, p. 1149, 2020.

[19] D. Moreira, S. Avila, M. Perez et al., "Pornography classification: the hidden clues in video space-time," *Forensic Science International*, vol. 268, pp. 46–61, 2016.

[20] I. Corley, J. Lwowski, and J. Hoffman, "DomainGAN: Generating Adversarial Examples to Attack Domain Generation Algorithm Classifiers," 2020, http://arxiv.org/abs/1911.06285.

[21] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications," 2020, http://arxiv.org/abs/2001.06937.

[22] A. Dash, J. Ye, and G. Wang, "A review of generative adversarial networks (GANs) and its applications in a wide variety of disciplines -- from medical to remote sensing," 2021, http://arxiv.org/abs/2110.01442.

[23] K. S. and M. Durgadevi, "Generative Adversarial Network (GAN): a general review on different variants of GAN and applications," *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, in *Proceedings of the 2021 6th International Conference on Communication and Electronics Systems (ICCES)*, pp. 1–8, Coimbatre, India, July 2021.

[24] G.-D. Hu, "Observers for one-sided Lipschitz non-linear systems," *IMA Journal of Mathematical Control and Information*, vol. 23, no. 4, pp. 395–401, 2006.

[25] I. Loeb, "Lipschitz functions in constructive reverse mathematics," *Logic Journal of IGPL*, vol. 21, no. 1, pp. 28–43, 2013.

[26] C. Caetano, S. Avila, W. R. Schwartz, S. J. F. Guimarães, and A. d. A. Araújo, "A mid-level video representation based on binary descriptors: a case study for pornography detection," *Neurocomputing*, vol. 213, pp. 102–114, 2016.

[27] Z. Cai, H. Huang, W. Wu, X. Ma, and X. Hu, "Detecting gathering incident of video surveillance based on plane geometry," in *Proceedings of the 2010 International Conference on Machine Vision and Human-machine Interface*, pp. 323–325, Kaifeng, China, Apr. 2010.

[28] Y. He, Y. Ye, P. Hanhart, and X. Xiu, "Motion compensated prediction with geometry padding for 360 video coding," in *Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–4, St. Petersburg, FL, USA, September 2017.

[29] H. Oh and S. Lee, "Visual presence: viewing geometry visual information of UHD S3D entertainment," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3358–3371, 2016.

[30] S. Avila, N. Thome, M. Cord, E. Valle, and A. Araújo, "Pooling in image representation: the visual codeword point of view,"

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 1701266, 8 pages
https://doi.org/10.1155/2022/1701266

Research Article

Cyberattacks Defense in Digital Music Streaming Platforms by Mobile Distributed Machine Learning

Guoxu Fan

Pingdingshan University, Conservatory of Music, Pingdingshan, Henan 467000, China

Correspondence should be addressed to Guoxu Fan; fanguoxu309@126.com

Received 29 March 2022; Accepted 19 April 2022; Published 6 May 2022

Academic Editor: Konstantinos Demertzis

Given the massive popularity of digital music industry repositories and their corresponding targeting by cybercriminals, this paper presents an intelligent model for cyberattacks defense in digital music streaming platforms by mobile distributed machine learning. The basic idea of machine learning is to use large data sets to create a model that responds well to inputs it has never processed before. With the increase in data volume and complexity of models, it becomes increasingly challenging to complete machine learning processes in a single machine. Distributed ML was developed to solve this problem, and a standard procedure is completed through the collaboration of multiple servers. With the evolution of mobile devices and the increase in their number, it is possible to create an integrated and compact mobile distributed machine learning (MDML) system that could reduce the workload of servers. A distributed logit polynomial function model is proposed, which is used to model options in distributed binary regression accounting units, which are of low complexity and high stability in noisy environments.

1. Introduction

One of the current challenges associated with the pandemic is the massive increase in content consumption through pay-TV and other streaming services. Users are forced to isolate themselves and have trapped many people at home [1]. From the first day of the enforcement of the measures, the music platforms available on the market and platforms for streaming content saw an increase in the number of their subscribers and the content they consume. For this reason, international streaming service providers have made recommendations for the need to reduce data rates to ensure the proper operation of networks, as the requirements have been maximized [2].

At the same time, cybercrime increased dramatically, especially the account takeover attack (ATA) [3] against music streaming platforms. Specifically, ATAs are a specialized type of attack in which criminals take ownership of online accounts using stolen access criteria to similar services. The basic techniques of these attacks are social engineering, brute force data breaches, and phishing campaigns. Botnets usually use the results of collecting these data to identify any other services that may use the same standards. In many cases, lists with similar data are sold on the dark web [4].

To further address these cases of breach and interception of credentials in a music industry service, this study proposes a highly efficient and easy-to-use MDML [5] that uses a polynomial logit function to model options in distributed binary regression units to timely detect cyberattacks in music content streaming services [6].

2. Related Literature

Mobile distributed machine learning and distributed computing [7] are concepts that the research community is currently trying to exploit in the best possible way the resources offered in each scenario [8, 9].

In 2015, Taddy [10] proposed a prototype method for the idea of distributed multinomial regression. Their research was fueled by the use of high-dimensional response multinomial models to analyze a large number of random counts. Content research, as texts are tokenized, and token counts are described as generating from a multinomial

depending on document properties, was one of its inspiring uses. They developed such algorithms using text projected onto a broad collection of explanatory factors using a publicly accessible data set of Yelp reviews. The fitted models may investigate the relationship among phrases and variables of concern, reduce dimensions into supervised component scores, and predict outcomes. We suggest that the technique presented here is an appealing choice for social engineers and other textual researchers who want to use regression tools they are acquainted with on text data.

Shamili et al. [11] developed a dispersed support vector machine technique for detecting computer viruses on a system of portable devices in the context of network security. The lightweight design uses a statistical classification system developed via training with instances of both regular and exceptional use patterns to monitor cellular user behavior in a dispersed and privacy-preserving manner. They claim that the distributed learning technique has many benefits, including being lightweight regarding Internet use, maintaining the privacy of participating users, and automatically generating a generic behavioral signature of virus based on typical user usage patterns. The system was tested employing the MIT reality data set, and the results were positive.

Gu et al. [12] conducted a literature study on the server-based to client-based machine learning transition. They again went through a few popular server-based and client-based deep learning methodologies and applications. They also spoke about the obstacles and potential future developments in this field. They have described their goals and showed how client-based machine learning is both sufficient and necessary. They have highlighted the limitations of client-based inference and illustrated recent achievements, particularly in the disciplines of machine vision and natural language interpretation. Finally, they have identified future research paths in academia and business for client-based machine learning. In conclusion, implementing client-based machine learning in real-world applications is still in progress.

Shakarami et al. [13] presented a study of ML-based computation offloading techniques in the mobile edge computing ecosystem in the shape of a classical taxonomy to identify current mechanisms and unresolved concerns in this critical field. Reinforcement training, guided learning, and uncontrolled learning were the three primary categories in their proposed taxonomy. Then, depending on various characteristics, the applicable methodologies were compared to one another. Finally, they discussed several crucial research problems as outstanding topics in the ML-based offloading mechanisms, considering the existing literature gap.

In a cognitive eavesdropping context, Guo et al. [5] examined a distributed machine learning strategy for a multiuser mobile edge computing system in a perceptual eavesdropping context, where several secondary devices have specific tasks to compute with varied priorities. They looked at the federated learning methodology for the system architecture of a multiuser mobile edge computing platform. Various users had distinct computational tasks that needed to be calculated by different computational access points. Finally, simulation results showed that the suggested strategy might successfully minimize system costs in both delay and energy usage while also ensuring that the user with the highest job priority receives greater bandwidth and processing capabilities. In future research expanding from this study, they will investigate some fairness disparities between consumers due to aspects such as the network state, job size, and processing capabilities. Furthermore, since most mobile devices have a limited battery life, it is difficult to maintain all mobile devices online.

3. Methodology

The proposed implementation, based on distributed multinomial logistic regression (DMLR) [10], is a classification method that extends the capabilities to multiclass issues, such as those with more than two possible unique outcomes [12]. In statistics, it is a model used to forecast the probability of several possible effects of a categorically distributed dependent variable, given a set of independent variables that might have varied values such as real, binary, category, and other types [5, 14].

The model follows the same technique as the accounting regression, with the sole variation being that the dependent variables are categorical rather than binary. In particular, there are K alternative outcomes rather than just two. The proposed solution uses a linear prediction function $f(k, i)$ to predict the probability that the observation i will have an effect k, of the following form [8, 15]:

$$f(k, i) = \beta_{0,k} + \beta_{1,k}x_{1,i} + \beta_{2,k}x_{2,i} + \cdots + \beta_{M,k}x_{M,i}, \tag{1}$$

where $\beta_{m,k}$ is a regression coefficient associated with the m-th explanatory variable and the k-th effect. Explanatory variables and coefficients are organized into vectors of magnitude $M+1$ so that the prediction function can be written in its most elaborate form [5]:

$$f(k, i) = \beta_k \cdot \mathbf{x}_i, \tag{2}$$

where β_k is the set of coefficients related to the result k, and x_i (vector line) is the set of explanatory variables related to the observation i.

To arrive at the polynomial logit model, we use the logic that by running $K-1$ independent regression accounting models, one result is selected as constant, and the other $K-1$ results are regressed against the fixed result. If the result K (the last result) is chosen as the constant, then [5, 10]

$$\begin{aligned} \ln\frac{\Pr(Y_i = 1)}{\Pr(Y_i = K)} &= \beta_1 \cdot \mathrm{X}_i, \\ \ln\frac{\Pr(Y_i = 2)}{\Pr(Y_i = K)} &= \beta_2 \cdot \mathrm{X}_i, \\ \ln\frac{\Pr(Y_i = K-1)}{\Pr(Y_i = K)} &= \beta_{K-1} \cdot \mathrm{X}_i. \end{aligned} \tag{3}$$

For this reason, we defined different sets of regression constants, one for each possible result. Raising the power of e

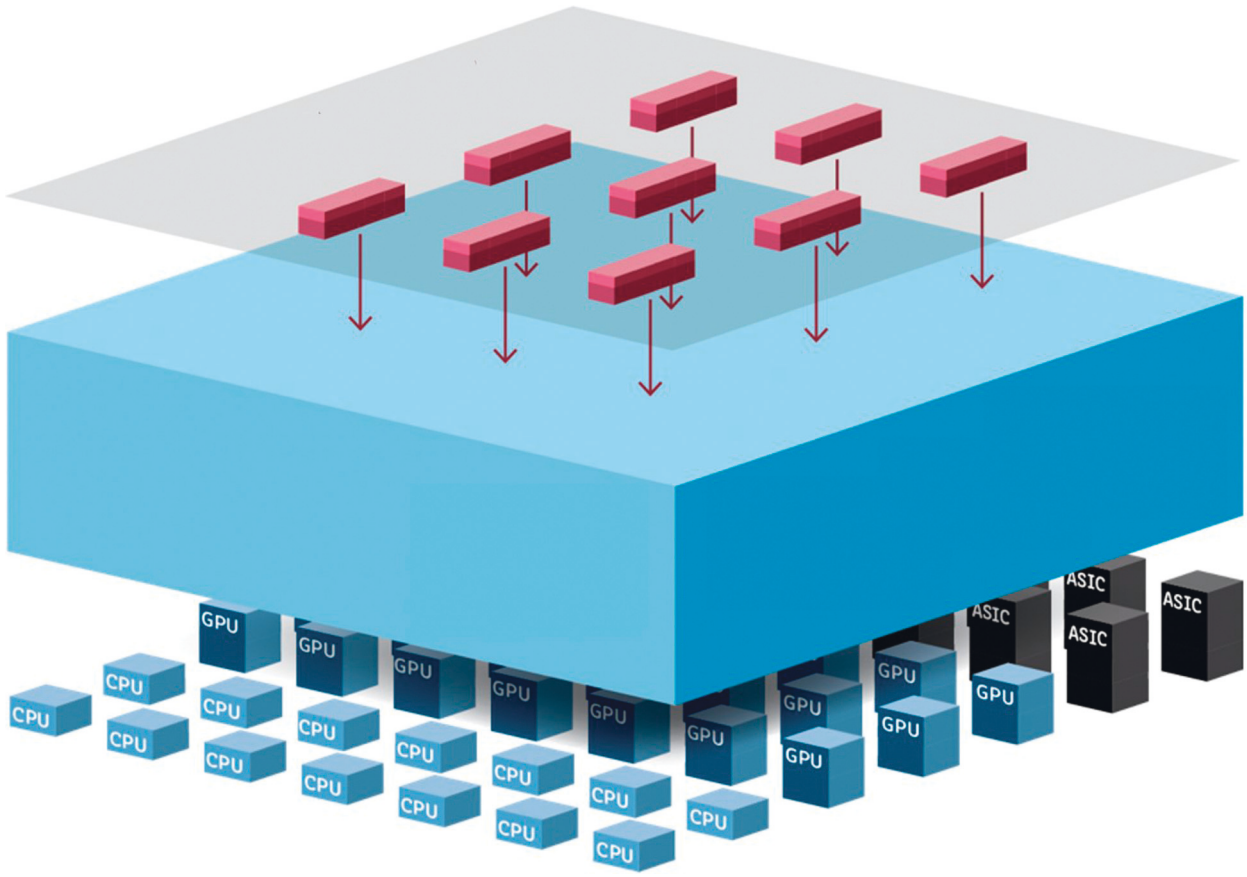


Figure 1: An abstract depiction of the proposed architecture.

on both sides of each equation and solving in terms of probabilities, we have [16, 17]

$$\begin{aligned} \Pr(Y_i = 1) &= \Pr(Y_i = K)e^{\beta_1 \cdot X_i}, \\ \Pr(Y_i = 2) &= \Pr(Y_i = K)e^{\beta_2 \cdot X_i}, \\ &\cdots\cdots \\ \Pr(Y_i = K-1) &= \Pr(Y_i = K)e^{\beta_{K-1} \cdot \mathbf{x}_i}. \end{aligned} \tag{4}$$

Using the fact that the sum of all K probabilities must make 1, then

$$\begin{aligned} \Pr(Y_i = K) &= 1 - \sum_{k=1}^{K-1} \Pr(Y_i = k) \\ &= 1 - \sum_{k=1}^{K-1} \Pr(Y_i = K)e^{\beta_k \cdot X_i} \Rightarrow \Pr(Y_i = K) \\ &= \frac{1}{1 + \sum_{k=1}^{K-1} e^{\beta_k \cdot X_i}}. \end{aligned} \tag{5}$$

Respectively, we use the above to find the other possibilities [11, 18, 19].

$$\begin{aligned} \Pr(Y_i = 1) &= \frac{e^{\boldsymbol{\beta}_1 \cdot \mathbf{x}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k \cdot \mathbf{X}_i}}, \\ \Pr(Y_i = 2) &= \frac{e^{\boldsymbol{\beta}_2 \cdot \mathbf{X}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k \cdot \mathbf{X}_i}}, \\ &\cdots\cdots \\ \Pr(Y_i = K-1) &= \frac{e^{\boldsymbol{\beta}_{K-1} \cdot \mathbf{X}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k \cdot \mathbf{X}_i}}. \end{aligned} \tag{6}$$

The fact that we performed multiple regressions proves why we assumed the independence of irrelevant alternatives. Thus, the estimation of the desired distributed solution is feasible. An abstract illustration of the proposed architecture, based on how the schema can work in MDML, is shown in Figure 1.

By this logic, this model is a distributed machine learning technique as the proposed learning algorithm is implemented in multiple nodes to improve performance, increase accuracy, and distribute the input data of a learning model. This distributed nature of the proposed algorithm allows substantiated decisions to be made from large data sets [20, 21].

Table 1: Features of dataset.

ID	Feature name	Type	ID	Feature name	Type
1	Age_Of_Domain	{1,−1}	2	Having_Ip_Address	{−1,1}
3	HTTPS_Token	{−1,1}	4	URL_Length	{1,0,−1}
5	Shortining_Service	{1,−1}	6	Having_At_Symbol	{1,−1}
7	Double_Slash_Redirecting	{−1,1}	8	Prefix_Suffix	{1,1}
9	Having_Sub_Domain	{−1,0,1}	10	Sslfinal_State	{−1,0,1}
11	Domain_Registeration_Length	{−1,1}	12	Favicon	{−1,1}
13	Port	{−1,1}	14	Request_Url	{−1,1}
15	URL_Of_Anchor	{−1,0,1}	16	Links_In_Tags	{−1,0,1}
17	Sfh	{−1,0,1}	18	Submitting_To_e-mail	{−1,1}
19	Abnormal_URL	{−1,1}	20	Redirect	{−1,1}
21	On_Mouseover	{−1,1}	22	Rightclick	{−1,1}
23	Popupwidnow	{−1,1}	24	Iframe	{−1,1}
25	Dnsrecord	{−1,1}	26	Web_Traffic	{−1,0,1}
27	Page_Rank	{−1,1}	28	Google_Index	{−1,1}
29	Links_Pointing_To_Page	{−1,0,1}	30	Statistical_Report	{−1,1}
31	Char_Freq_;	Real	32	Char_Freq_(	REAL
33	Char_Freq_[	Real	34	Char_Freq_!	REAL
35	Char_Freq_$	Real	36	Char_Freq_#	REAL
37	"Is_Host_Login"	{−1,1}	38	"Is_Guest_Login"	{−1,1}
39	"Num_Failed_Logins"	Real	40	"Logged_In"	{−1,1}
41	"Root_Shell"	Real	42	"Su_Attempted"	REAL
43	"Num_Root"	Real	44	Credit_Amount	REAL
45	Credit_History	"No credits," "All paid," "Existing paid," "Delayed paid," "Critical"	46	Class	Attack/ Normal

4. Use Case

For the modeling of the proposed system, a specialized ATA scenario was implemented, a threat that is one of the most critical risks of music content streaming applications today. Cybercriminals have developed a substantial criminal interest in the music ecosystem. The influx of visitors combined with the vast amounts of music content spent daily creates a new landscape of threats [22]. By this logic, they often utilize advanced techniques, even zero-day attacks, to steal credentials and generally launch ATA on music streaming platforms. The most common tactic is to use bot infrastructure to capture the accounts of unsuspecting users and exploit them for financial gain [23, 24].

Given the distributed nature of these applications and their use typically by mobile applications, this scenario implements a complete and compact DMLR cyber security system. The data in use are about an innovative clickstream dataset inspired by how phishing campaigns are detected in conjunction with credit card fraud detection techniques [24–28]. The features used are presented in detail in Table 1.

The DMLR presumes that the information is case-specific. For each scenario, each independent variable has a unique value. The model also implies that the independent variables can predict the dependent variable properly. There is no requirement for the independent variables to be statistically independent, as with other types of regression. However, coliteracy is regarded as relatively low because it becomes difficult to distinguish between the effects of several variables if the contrary is true [8, 16].

When used to model alternatives, the polynomial logit function relies on the independence of irrelevant alternatives (IIA), which is not always desirable. This implies that the likelihood of selecting one class over another is independent of the presence or absence of other irrelevant choices [29]. For example, if a bicycle is offered as an additional choice, the relative odds of choosing a car or bus to get to work do not vary. This enables the modeling of a set of $K-1$ independent binary options as a collection of selected K alternatives. Each separate is chosen as a constant, and the other $K-1$s are compared to it one by one. Although hypothesis IIA is a core hypothesis in rational choice theory, several psychological investigations reveal that people frequently breach this criterion when making decisions [30, 31].

When the polynomial logit is used to model options, it might produce too much confusion between the meaningful choices and between different alternatives in some instances. This is critical if the study's goal is to forecast how options would change if one of them disappeared. In comparable cases, other models, such as the nested logit or the polynomial probit, can be utilized because they enable the violation of the IIA.

As a log-linear model, binary accounting regression can be extended to multiple models. The logarithm of the separation function is obtained by modeling the logarithm of the probability of seeing a specific output with the linear classifier and a normalizing factor [32–35].

$$\begin{aligned} \ln \Pr(Y_i = 1) &= \beta_1 \cdot X_i - \ln Z, \\ \ln \Pr(Y_i = 2) &= \beta_2 \cdot X_i - \ln Z, \\ &\cdots \\ \ln \Pr(Y_i = K) &= \beta_K, \\ \ln \Pr(Y_i = K) &= \beta_K \cdot X_i - \ln Z. \end{aligned} \tag{7}$$

As in the binary case, we need an additional term $-\ln Z$ to ensure that all probabilities form a probability distribution so that they have a sum of 1.

$$\sum_{k=1}^{K} \Pr(Y_i = k) = 1. \tag{8}$$

We need to add a condition to ensure normalization, in addition to the usual multiplication, because we have taken the logarithm of the probabilities [36]. By increasing the power of each member of the equations, we convert the prosthetic part into a multiplier, and the probability becomes the measure of Gibbs [37].

$$\begin{aligned} \Pr(Y_i = 1) &= \frac{1}{Z} e^{\boldsymbol{\beta}_1 \cdot \mathbf{x}_i}, \\ \Pr(Y_i = 2) &= \frac{1}{Z} e^{\boldsymbol{\beta}_2 \cdot \mathbf{x}_i}, \\ &\cdots\cdots \\ \Pr(Y_i = K) &= \frac{1}{Z} e^{\boldsymbol{\beta}_K \cdot \mathbf{x}_i}. \end{aligned} \tag{9}$$

The quantity Z is known as the distribution partition function. We can compute the value of this function by applying the preceding constraint, which demands that all probabilities have a sum of 1.

$$\begin{aligned} 1 &= \sum_{k=1}^{K} \Pr(Y_i = k) = \sum_{k=1}^{K} \frac{1}{Z} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i} = \frac{1}{Z} \sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}, \\ Z &= \sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}. \end{aligned} \tag{10}$$

The coefficient is a constant because it is not a function of Y_i which is the variable thanks to which the probability distribution is defined. However, it is not variable about the explanatory variables or unknown β_k coefficients of regression, which should be determined through an optimization process. The final equations of probability are [38–40]

$$\begin{aligned} \Pr(Y_i = 1) &= \frac{e^{\boldsymbol{\beta}_1 \cdot \mathbf{X}_i}}{\sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}}, \\ \Pr(Y_i = 2) &= \frac{e^{\boldsymbol{\beta}_2 \cdot \mathbf{X}_i}}{\sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}}, \\ &\cdots \\ \Pr(Y_i = K) &= \frac{e^{\boldsymbol{\beta}_K \cdot \mathbf{X}_i}}{\sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}}, \\ \Pr(Y_i = c) &= \frac{e^{\boldsymbol{\beta}_c \cdot \mathbf{X}_i}}{\sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{X}_i}}, \\ \operatorname{softmax}(k, x_1, \ldots, x_n) &= \frac{e^{x_k}}{\sum_{i=1}^{n} e^{x_i}} \end{aligned} \tag{11}$$

is called a softmax function. The reason is that the exposure of the variables $(x_1, \ldots, x_n)$ magnifies the differences between them. As a result, softmax $(\mathrm{k}, x_1, \ldots, x_n)$ [41, 42] will return a value close to 0 when x_k is much less than the maximum of all values and will return 1 when applied to the maximum value, unless very close to the next highest price.

This function may build a weighted average that acts like a smooth function (and can be easily separated) and values the index function.

$$f(k) = \left\{ \begin{matrix} 1, k = \arg\max(x_1, \ldots, x_n) \\ 0, \text{otherwise} \end{matrix} \right\}. \tag{12}$$

Thus, we can write the probability equations as follows:

$$\Pr(Y_i = c) = \operatorname{soft\,max}(c, \beta_1 \cdot X_i, \ldots, \beta_K \cdot X_i). \tag{13}$$

This function is the equivalent of the accounting function in binary accounting regression.

In general, there are only $k-1$ individually differentiable probabilities, and therefore, there are $k-1$ separately distinct coefficient vectors. One way to look at this is to see that if we add a constant vector to all factor vectors, then the equations are identical.

$$\frac{e^{(\boldsymbol{\beta}_c + C) \cdot \mathbf{x}_i}}{\sum_{k=1}^{K} e^{(\boldsymbol{\beta}_k + C) \cdot \mathbf{x}_i}} = \frac{e^{\boldsymbol{\beta}_c \cdot \mathbf{x}_i} e^{C \cdot \mathbf{x}_i}}{\sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i} e^{C \cdot \mathbf{x}_i}} = \frac{e^{C \cdot \mathbf{x}_i} e^{\boldsymbol{\beta}_c \cdot \mathbf{x}_i}}{e^{C \cdot \mathbf{x}_i} \sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}} = \frac{e^{\boldsymbol{\beta}_c \cdot \mathbf{x}_i}}{\sum_{k=1}^{K} e^{\boldsymbol{\beta}_k \cdot \mathbf{x}_i}}. \tag{14}$$

As a result, it is common to set $C = -\beta K$ (or someone other than the coefficient vector). Essentially, we set the variable so that one of the vectors becomes 0, and the remaining vectors are changed into the difference between these vectors and the vectors we chose. The coefficients are transformed mathematically as follows [43]:

$$\begin{aligned} \beta_1^{'} &= \beta_1 - \beta_K, \\ &\cdots\cdots \\ \beta_{K-1}^{'} &= \beta_{K-1} - \beta_K, \\ \beta_K^{'} &= 0. \end{aligned} \tag{15}$$

This leads to the following equations [13, 35]:

$$\begin{aligned} \Pr(Y_i = 1) &= \frac{e^{\boldsymbol{\beta}_1^{'} \cdot \mathbf{x}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k^{'} \cdot \mathbf{x}_i}}, \\ &\cdots \\ \Pr(Y_i = K-1) &= \frac{e^{\boldsymbol{\beta}_{K-1}^{'} \cdot \mathbf{x}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k^{'} \cdot \mathbf{x}_i}}, \\ \Pr(Y_i = K) &= \frac{1}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k^{'} \cdot \mathbf{x}_i}}. \end{aligned} \tag{16}$$

The distributed configuration, which is based on the configuration server and is adopted here, was acquired in 8 different contexts, and the results of the process are presented in Table 2.

Table 2: Model performance.

ID	Model Part	ROC Curve	F-score	Recall	Precision
1	Part-1	0.9777	0.9741	0.9770	0.9770
2	Part-2	0.9789	0.9735	0.9745	0.9776
3	Part-3	0.9663	0.9621	0.9610	0.9650
4	Part-4	0.9839	0.9818	0.9885	0.9861
5	Part-5	0.9747	0.9761	0.9735	0.9714
6	Part-6	0.9630	0.9635	0.9677	0.9660
7	Part-7	0.9786	0.9743	0.9750	0.9779
Summary		**0.9741**	**0.9719**	**0.9734**	**0.9738**

The proposed method attempts to approach the amount of $\Pr(Y_i = 1)$ by performing the sampling for a few repetitions. Binary regression units are initialized to a sample of actual data and perform N iterations, taking some data due to the model contribution [10, 44, 45]. Essentially, the method undertakes to give lower importance to the real data and much higher energy in the cases where a marginal distribution comes from, thus helping the model to approach the actual data distribution and determine the nature of the ATA.

In conclusion, the proposed architecture can accurately learn N samples from $N-1$ models, while its learning speed can be even thousands of times faster than conventional methods. It is noteworthy that all βk coefficient vectors are uniquely recognizable. This has to do with the fact that all the odds must be added to 1, making one of them wholly decided while all the others are unknown. This can be translated, in combination with the very stable ability to predict, as the absence of contradictions between the models that act uniformly and respectively the existence of complete information and information about all the elements that make up the problem, so that the purpose of the decision comes from the choice of the optimal solution that maximizes the objective function and at the same time satisfies specific criteria.

5. Conclusions

In this work, we proposed an innovative MDML system, which significantly reduces the workload of servers in distributed environments and undertakes to perform machine learning tasks on large-scale data. Specifically, a distributed logit polynomial function model was proposed, which is used to model options in distributed binary regression accounting units. It is a highly efficient and low-demand distributed machine learning system tested to solve a highly demanding cybersecurity problem associated with distributed music repositories. Given the massive popularity of digital music repositories, targeted ATA attacks are carried out with malicious intent, such as stealing and removing personal information from user accounts.

As it turned out experimentally, the proposed system managed to categorize with great success the elements that lead to ATA attacks patterns. The high accuracy combined with the generalization of the results of the extensive tests indicates that this system is suitable for distributed environments and for solving highly complex problems. It is essential to say that the proposed standard had very stable predictability (0.9635 <F-score <0.9818), which proves the great coherence of the models that act uniformly in recognition of standards and respectively the existence of complete information and information for all elements that make up the problem, even if they come from a distributed environment.

In the future, the proposed method can be extended by developing a mobile application that will work in an even more efficient and flexible federated style, allowing several users to contribute to the model training process. By merging unsupervised and supervised learning methodologies, the distributed training algorithm may be optimized and operated at an abstract level to handle more complex and multidimensional data in more extensive data sets. Furthermore, some bioinspired optimization approaches, such as the particle optimization methodology, can be utilized to develop solutions that maximize or reduce some study parameters, such as the cost function. Finally, it would be fascinating to use the polynomial logit function to create a distributed neural network in which kernel technology would be used to address issues with nonlinearity but time continuities.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] C. Egbert, F. Alhenaki, and D. Johnson, "Leveraging a music streaming platform in establishing a novel storage covert channel," in *Proceedings of the 2020 IEEE 45th Conference on Local Computer Networks (LCN)*, pp. 437–440, Sydney, Australia, November. 2020.

[2] S. Barua and D. Talukder, "A blockchain based decentralized video streaming platform with content protection system," in *Proceedings of the 2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6, Dhaka, Bangladesh, December. 2020.

[3] R. Kawase, F. Diana, M. Czeladka, M. Schüler, and M. Faust, "Internet fraud: the case of account takeover in online marketplace," in *Proceedings of the 30th ACM Conference on Hypertext and Social Media*, pp. 181–190, Hof, Germany, September. 2019.

[4] P. Mulinka, P. Casas, and J. Vanerio, "Continuous and adaptive learning over big streaming data for network security," in *Proceedings of the 2019 IEEE 8th International Conference on Cloud Networking (CloudNet)*, pp. 1–4, Coimbra, Portugal, August. 2019.

[5] Y. Guo, R. Zhao, S. Lai, L. Fan, X. Lei, and G. K. Karagiannidis, "Distributed machine learning for multiuser mobile edge computing systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2022, Article ID 3140660, 1 page, 2022.

[6] F. Guo and S. Cao, "Research on the service guarantee strategy based on streaming media platform," in *Proceedings of the 2015 6th IEEE International Conference on Software*

Engineering and Service Science (ICSESS), pp. 371–374, Article ID 7339077, Beijing, China, September. 2015.

[7] T. Tuor, S. Wang, T. Salonidis, B. J. Ko, and K. K. Leung, "Demo abstract: distributed machine learning at resource-limited edge nodes," in *Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1-2, Article ID 8406837, Honolulu, HI, USA, April. 2018.

[8] Z. Gu and Y. Yang, "Detecting malicious model updates from federated learning on conditional variational autoencoder," in *Proceedings of the 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 671–680, Portland, OR, USA, May 2021.

[9] P. Lin, "Research on optimization of distributed big data real-time management method," in *Proceedings of the 2018 3rd International Conference on Smart City and Systems Engineering (ICSCSE)*, pp. 626–630, Article ID 00134, Xiamen, China, September. 2018.

[10] M. Taddy, "Distributed multinomial regression," *Annals of Applied Statistics*, vol. 9, no. 3, 2015.

[11] A. S. Shamili, C. Bauckhage, and T. Alpcan, "Malware detection on mobile devices using distributed machine learning," in *Proceedings of the 2010 20th International Conference on Pattern Recognition*, pp. 4348–4351, Article ID 1057, Istanbul, Turkey, August. 2010.

[12] R. Gu, C. Niu, F. Wu et al., "From server-based to client-based machine learning," *ACM Computing Surveys*, vol. 54, no. 1, pp. 1–36, 2022.

[13] A. Shakarami, M. Ghobaei-Arani, and A. Shahidinejad, "A survey on the computation offloading approaches in mobile edge computing: a machine learning-based perspective," *Computer Networks*, vol. 182, Article ID 107496, 2020.

[14] M. Dhingra, M. Jain, and R. S. Jadon, "Role of artificial intelligence in enterprise information security: a review," in *Proceedings of the 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pp. 188–191, Article ID 7913142, Waknaghat, India, December. 2016.

[15] J. Wang, B. Cao, P. Yu, L. Sun, W. Bao, and X. Zhu, "Deep learning towards mobile applications," in *Proceedings of the 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1385–1393, Article ID 00139, Vienna, Austria, July. 2018.

[16] Y. Chen, Y. Mao, H. Liang, S. Yu, Y. Wei, and S. Leng, "Data poison detection schemes for distributed machine learning," *IEEE Access*, vol. 8, pp. 7442–7454, Article ID 2962525, 2019.

[17] U. Mohammad, S. Sorour, and M. Hefeida, "Dynamic task allocation for mobile edge learning," *IEEE Transactions on Mobile Computing*, vol. 2021, Article ID 3137017, 1 page, 2021.

[18] M. Nind and H. Vinha, "Doing research inclusively: bridges to multiple possibilities in inclusive research," *British Journal of Learning Disabilites*, vol. 42, no. 2, pp. 102–109, 2014.

[19] H. Zheng, H. Hu, and Z. Han, "Preserving user privacy for machine learning: local differential privacy or federated machine learning?" *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 5–14, Article ID 3010335, 2020.

[20] J. Wu, S. Guo, J. Li, and D. Zeng, "Big data meet green challenges: greening big data," *IEEE Systems Journal*, vol. 10, no. 3, pp. 873–887, Article ID 2550538, 2016.

[21] J. Wu, S. Guo, J. Li, and D. Zeng, "Big data meet green challenges: big data toward green applications," *IEEE Systems Journal*, vol. 10, no. 3, pp. 888–900, Article ID 2550530, 2016.

[22] K. Rastogi, D. Lohani, and D. Acharya, "Context-aware monitoring and control of ventilation rate in indoor environments using internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9257–9267, Article ID 3057919, 2021.

[23] D. Sarma, W. Alam, I. Saha, M. N. Alam, M. J. Alam, and S. Hossain, "Bank fraud detection using community detection algorithm," in *Proceedings of the 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 642–646, Coimbatore, India, July. 2020.

[24] Z. Song, "A data mining based fraud detection hybrid algorithm in E-bank," in *Proceedings of the 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pp. 44–47, Fuzhou, China, June. 2020.

[25] S. Huda, Aripin, M. F. Naufal, V. M. Yudianingtias, and Anisti, "Fraud patterns classification: a study of fraud in business process of Indonesian online sales transaction," in *Proceedings of the 2020 3rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT)*, pp. 212–217, Medan, Indonesia, June. 2020.

[26] Y. Pristyanto and A. Dahlan, "Hybrid resampling for imbalanced class handling on web phishing classification dataset," in *Proceedings of the 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pp. 401–406, Yogyakarta, Indonesia, November. 2019.

[27] S. P. Ripa, F. Islam, and M. Arifuzzaman, "The emergence threat of phishing attack and the detection techniques using machine learning models," in *Proceedings of the 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, pp. 1–6, Rajshahi, Bangladesh, July. 2021.

[28] J. Rashid, T. Mahmood, M. W. Nisar, and T. Nazir, "Phishing detection using machine learning technique," in *Proceedings of the 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)*, pp. 43–46, Riyadh, Saudi Arabia, August. 2020.

[29] R. Atat, L. Liu, H. Chen, J. Wu, H. Li, and Y. Yi, "Enabling cyber-physical communication in 5G cellular networks: challenges, spatial spectrum sensing, and cyber-security," *IET Cyber-Physical Systems: Theory & Applications*, vol. 2, no. 1, pp. 49–54, 2017.

[30] B. Jiang, "Two-party secure computation for any polynomial function on ciphertexts under different secret keys," *Security and Communication Networks*, vol. 2021, Article ID e6695304, 11 pages, 2021.

[31] G. Castaneda, P. Morris, and T. M. Khoshgoftaar, "Maxout neural network for big data medical fraud detection," in *Proceedings of the 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)*, pp. 357–362, Article ID 00064, Newark, CA, USA, April. 2019.

[32] J. Yu, Y. Lee, K. C. Yow, M. Jeon, and W. Pedrycz, "Abnormal event detection and localization via adversarial event prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 2021, Article ID 3053563, 15 pages, 2021.

[33] N. Gedik, "Least squares Support vector mechanics to predict the stability number of rubble-mound breakwaters," *Water*, vol. 10, no. 10, Article ID 10, 2018.

[34] T. W. Frick, R. D. Myers, and C. Dagli, "Analysis of patterns in time for evaluating effectiveness of first principles of instruction," *Educational Technology Research & Development*, vol. 70, 2022.

[35] M. Ahmadlou and H. Adeli, "Enhanced probabilistic neural network with local decision circles: a robust classifier,"

Integrated Computer-Aided Engineering, vol. 17, no. 3, pp. 197–210, 2010.

[36] M. McCord, P. Davis, J. McCord, M. Haran, and K. Davison, "An exploratory investigation into the relationship between energy performance certificates and sales price: a polytomous universal model approach," *Journal of Financial Management of Property and Construction*, vol. 25, no. 2, pp. 247–271, 2020.

[37] A. J. M. Garrett, "Review: probability theory: the logic of science, by E. T. Jaynes," *Law, Probability and Risk*, vol. 3, no. 3–4, pp. 243–246, 2004.

[38] S. Guopan, "The effect of probability on risk perception and risk preference in decision making," in *Proceedings of the 2010 International Conference on Education and Management Technology*, pp. 690–693, Article ID 5657565, Cairo, Egypt, November. 2010.

[39] J. Gawlikowski, J. Feng, and P. Jung, "A survey of uncertainty in deep neural networks," Accessed: Nov. 06, 2021. [Online]. Available: http://arxiv.org/abs/2107.03342, 2021.

[40] H. Worthington, R. S. McCrea, R. King, and R. A. Griffiths, "Estimation of population size when capture probability depends on individual states," *Journal of Agricultural, Biological, and Environmental Statistics*, vol. 24, no. 1, pp. 154–172, 2019.

[41] M. A. Hussain and T.-H. Tsai, "An efficient and fast softmax hardware architecture (EFSHA) for deep neural networks," in *Proceedings of the 2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pp. 1–4, Washington, DC, USA, June. 2021.

[42] Q. Rao, B. Yu, K. He, and B. Feng, "Regularization and iterative initialization of softmax for fast training of convolutional neural networks," in *Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Article ID 8852459, Budapest, Hungary, July. 2019.

[43] P. Akubathini, S. Chouksey, and H. S. Satheesh, "Evaluation of Machine Learning approaches for resource constrained IIoT devices," in *Proceedings of the 2021 13th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pp. 74–79, Changsha, China, July. 2021.

[44] O. O. Koyejo, N. Natarajan, P. K. Ravikumar, and I. S. Dhillon, "Consistent binary classification with generalized performance metrics," *Advances in Neural Information Processing Systems*, vol. 27, 2014 Accessed: Oct. 24, 2021. [Online]. Available: https://papers.nips.cc/paper/2014/hash/30c8e1ca872524fbf7ea5c519ca397ee-Abstract.html.

[45] S. Raschka, "An overview of general performance metrics of binary classifier systems," Accessed: Nov. 09, 2021. [Online]. Available: http://arxiv.org/abs/1410.5330, 2014.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9854650, 1 page
https://doi.org/10.1155/2023/9854650

Retraction

Retracted: Privacy-Preserving Sports Wearable Data Fusion Framework

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] J. Li and J. Zhang, "Privacy-Preserving Sports Wearable Data Fusion Framework," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6131971, 7 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6131971, 7 pages
https://doi.org/10.1155/2022/6131971

Research Article

Privacy-Preserving Sports Wearable Data Fusion Framework

Jia Li and Jie Zhang

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Jia Li; lijia19900117@163.com

Received 31 March 2022; Revised 11 April 2022; Accepted 13 April 2022; Published 4 May 2022

Academic Editor: Konstantinos Demertzis

When the sports industry has access to advanced training and preparation techniques, the sports sector is entering a new era, where real-time data processing services have a crucial priority in improving physical fitness and avoiding injuries to athletes. The primary sports support methodology is based on multiple sensors, mainly wearables, often of different types and technology, which collect somatometric data in real time and are usually analyzed with deep learning technologies. And while modern athletes train and prepare intelligently using the innovative techniques of available technology, there is considerable concern about the use of personal data. There is great concern about cyberattacks and possible data leaks that could affect the sports industry and sports in general. To secure the personal data of athletes collected and analyzed by sports wearables, this paper presents a privacy-preserving sports wearable data fusion framework. This is an advanced methodology based on Lagrange's relaxation method for the problem of multiple assignments and synthesis of information by numerous sensors and the use of differential privacy to access databases with personal information, ensuring that this information will remain personal without a third entity may disclose the identity of the athlete who provided the data.

1. Introduction

To overcome the competition, the modern athlete must train and prepare intelligently and take advantage of innovative techniques. Its training program must be fully personalized, incorporating advanced tools of particular precision and functionality, based on the latest scientific innovations, advanced training systems, multidisciplinary medical positions, sports researchers, and people working in advanced sports [1]. The implementation of such a program includes unique tools for monitoring the athlete's health, ergonomic characteristics, and ways to manage training load and avoid injuries [2, 3].

A key innovation used by the entire sports industry is the athlete's involvement in capturing valuable information daily [4]. The time that the athlete must devote is usually identified with the hours of his daily training, and the data recorded is generally divided into three main categories [5].

(1) Wellness: The athlete's well-being is recorded daily based on his answer to seven critical questions.

(2) Training load: At the end of each training unit, the athlete registers the subjective sense of effort, which leads to valuable conclusions compared to the training load designed by the coach. With these data, documented indicators are calculated, such as the weekly load change, the ratio of current and chronic load, and the monotonicity index, to capture whether the athlete is in the ideal training zone, in a subtraining zone, or has entered a zone with a high risk of injury. For example, when the current and chronic load ratio is high (>1.5), the risk of injury increases, and the training load must be corrected. Also, the weekly increase in load is an essential indicator for injury prevention. A 15% increase in load compared to the previous week causes a 50% increase in the probability of injury.

(3) Health: The athlete can record extraordinary changes in his health due to illness or injury. In the purely training field, the coach undertakes the detailed planning of the training, which can be individualized and adapted, by category, for athletes who are rested, injured, absent, etc. All kinds of evaluations are collected and recorded from tests such as blood, platelets, fasting glucose, iron, creatinine, total

cholesterol, up to weekly vertical jump markers, maximal oxygen uptake, but also special tests such as substance abuse control, amphetamines, cocaine, cannabinoids, barbiturates, opioids (heroin, codeine, morphine), ethanol (alcohol), benzodiazepines, and evaluation of an acceptable or nonacceptable creatinine sample.

Because it is information on an identified or identifiable physical person, all of the above information constitutes personal data. It is subject to the status of personal data legislation ("data subject") [6]. An identifiable physical person is one whose identity can be determined, directly or indirectly, through the use of an identifier such as a name, identity number, location data, online identifier, or one or more identifiable factors such as that physical person's physical, physiological, genetic, psychological, economic, cultural, or social identity [7].

Data fusion [8] of personal data from multiple sensors for the rational use of various information is a highly complex research problem without identifying an effective solution for the functional and practical expansion of advanced personal data usage applications [9]. This view is particularly noticeable because the nature of these applications is constantly changing towards more centralized and demanding applications, where the management of incoming information is not as apparent as it was in the usually single-sensory systems of the first generation of data acquisition and management applications [10, 11]. These applications are also evolving and growing in number, incorporating increasingly sensitive information, which requires more advanced security techniques [12, 13]. All of the above introduce different types and topologies of sensors, increasing the need for a common and effective intermediate level of security between sensors and applications [14, 15].

Data mining privacy preservation entails concealing output knowledge of data through various approaches when the output data are valuable and private. This is mainly accomplished by employing two techniques: input privacy, in which information is changed using multiple styles, and output privacy, in which data are transformed to conceal the rules. Privacy preservation is critical in data mining because when data are moved or communicated between different parties, it is required to offer security to that data so that other parties do not know what information is displayed between the original parties.

Managing this coming from multiple sources different from each other complementary or surplus personal information has been recognized as a significant and critical factor in the development of sport and, in general, in preserving the prestige and credibility of the sports industry. The intermediate level of security between sensors and applications is the position occupied by the proposed privacy-preserving sports wearable data fusion framework [16, 17]. The proposed methodology ensures the integrity of the data synthesis from heterogeneous sources while guaranteeing the anonymity and reliability of the data even in cases of the use of the data in question by third-party analysts.

2. Related Literature

Because of the expansion of wearable devices and the fact that they manage personal data [14, 18], the research community focuses on privacy-preserving frameworks, as seen in the literature shown below.

Banerjee et al. [16] investigated the appropriateness of the Health Insurance Portability, and Accountability Act (HIPAA) concerns created by wearable technology in the IoT ecosystem, identifying legislative gaps and variables that promote health data exposure. They developed a partnership-identity risk model, showed the ramifications in four distinct settings, and offered privacy protection advice. They classified industrial self-regulation from "pure" self-regulation to "mixed" self-regulation. There is no government involvement or any other stakeholder in the private regulating mechanism, public standard setting, pricing, or output setting. There is a high level of close federal monitoring. They noted that many of the issues with health data sharing would be addressed by the business itself. However, a hybrid of industry rule-making and government monitoring has the most potential for industry self-regulation.

Zarepour et al. [19] proposed a privacy-aware architecture for wearable cameras that might safeguard all sensitive topics such as persons, objects, and places. It identifies the likely sensitive issues in each picture using contextual information acquired from the wearable sensors and stored photos. Various techniques are used to identify sensitive items after detecting the surroundings and the user's behavior. The sensitive items are first placed and then obscured or erased using image editing methods. Their findings indicated that the suggested system could identify and blur sensitive objects with sufficient precision in both an interior and an outside setting.

In 2014, Safavi et al. [20] proposed a theoretical model for wearable medical systems, which included ten concepts and nine tests capable of delivering a comprehensive privacy protection bundle to wearable device users, and which could be implemented on any wearable OS. They built this framework by examining current mobile technology, which was then coupled with current security norms and assessed using strict information security principles. They have also recommended a detailed checklist that might aid both designers and manufacturers improve the quality of their products' privacy measures. Finally, they acknowledged that these frameworks would be impossible to execute without law compliance that integrates security and confidentiality with regulation.

Chen et al. [21] introduced FedHealth, a distributed transition knowledge architecture for wearable healthcare, to address the difficulties of user data being stored in isolated islands and cloud-based models failing to personalize. FedHealth is a broad and extendable system that conducts data aggregation using federated learning and then creates reasonably tailored models using transfer learning in various healthcare applications. Their tests and applications have shown that accurate and individualized healthcare may be provided without jeopardizing privacy and security. They want to expand this technique with incremental learning in the future to provide more tailored and adaptable treatment.

Psychoula et al. [17] examined privacy resilience and methods for preserving and integrating privacy into present frameworks. As customers grow more conscious of privacy threats and demand greater privacy control from service providers, the privacy environment will evolve. Frameworks that include privacy risks might affect how data are kept, processed, and shared. They argue that data collection, management, and sharing will become even more fragmented, with each service provider having to subscribe to a user's info instead of the other side around. As a result, addressing the privacy protection dilemma requires focusing on privacy knowledge and risk. Methods for understanding and learning user preferences and negotiating to satisfy their expectations should be researched. Finally, developing algorithmic privacy risk indicators can reliably determine a person's privacy risk based on data acquired and provided about the user.

Finally, Poore et al. [22] introduced the Lagrangian relaxation method that we use, stating that these techniques have proven to be particularly helpful in solving these issues to the interference level in real time, particularly for dense scenarios and numerous scans of data from various sensors. Their research introduced a new family of creative Lagrangian relaxation methods that address some of the shortcomings of prior approaches. The efficiency and efficacy of their technique class are shown by various numerical investigations.

From the above literature, we can say that privacy-preserving [23] frameworks are under the research community's focus because of the explosion of these devices and the fact that they handle personal data [16].

3. Methodology

Data merging occurs when data from many sources are merged to reflect a single reference point. Although it appears to be a simple goal, data merging is a complex procedure because most databases suffer from redundancy, inconsistency, and inaccuracy. To derive significant insights from the data obtained, it is necessary to consolidate all of these data sources and get a single point of reference. The requirement for database compliance with data privacy legislation had far-reaching consequences for database management methods. However, various obstacles must be overcome to ensure database compliance with data privacy rules.

Differential privacy is a technique for publicly disclosing information about a dataset by defining the patterns of groups within the dataset while maintaining the privacy of individuals. The assumption behind differential privacy is that if the effect of a single arbitrary database modification is small enough, the query result cannot be used to infer much about any one individual, hence ensuring privacy. Differential privacy can also be defined as a constraint placed on the algorithms used to publish aggregate information about a statistical database that prevents publishing private information about individual records whose data are contained in the database. For example, some government agencies use differentially private algorithms to publish demographic data or other statistical aggregates while maintaining the confidentiality of survey responses. Businesses use them to collect information about user behavior while limiting what is visible to even internal analysts.

Differential privacy is usually considered when identifying persons whose information may be saved in a database. Although it does not explicitly address issues of identification and reidentification, differentially private algorithms are expected to be immune to such attacks. A differentially secret algorithm is one in which the observer who sees the output has no way of knowing if the computation utilizes the information of a specific individual.

The proposed methodology ensures the integrity of the data synthesis from heterogeneous sources while guaranteeing the anonymity and reliability of the data even in cases of the use of the data in question by third-party analysts [12, 24].

Specifically, having the problem of data fusion from N sensors, its modeling turns into the following optimization problem [13, 25, 26]:

$$u(z) = \max_{z_{i1}i_2i_3} \sum_{i_1=0}^{M_1} \sum_{i_2=0}^{M_2} \sum_{i_3=0}^{M_3} \mathscr{C}_{i_1i_2i_3} z_{i_1i_2i_3}, \tag{1}$$

if the following restrictions apply

$$\begin{aligned} &\sum_{i_2=0}^{M_2} \sum_{i_3=0}^{M_3} z_{i_1i_2i_3} = 1, \quad i_1 = 1, 2, \ldots, M_1, \\ &\sum_{i_1=0}^{M_1} \sum_{i_3=0}^{M_3} z_{i_1i_2i_3} = 1, \quad i_2 = 1, 2, \ldots, M_2, \\ &\sum_{i_1=0}^{M_1} \sum_{i_2=0}^{M_2} z_{i_1i_2i_3} = 1, \quad i_3 = 1, 2, \ldots, M_S. \end{aligned} \tag{2}$$

According to Lagrange's relaxation method [22], a set of constraints is subtracted and expressed with the help of Lagrange multipliers in the objective function of the above equation. The motivation for this approach is that a proper selection of Lagrange multipliers will tend to satisfy the inherent limitations typically found in a similar problem [27]. Thus, the three-dimensional assignment problem becomes a two-dimensional assignment problem [28].

We assume that we have S sets of measurements from N_S sensors, which monitor an athlete and detect target points. Still, the number is not necessarily equal to the number of actual targets set in training. The S-dimensional problem is presented as follows [4, 9, 15]:

$$\max \sum_{i_1=0}^{M_1} \cdots \sum_{i_S=0}^{M_S} \mathscr{C}_{i_1\ldots i_S} z_{i_1\ldots i_S}. \tag{3}$$

Given the fact that

$$\begin{aligned} &\sum_{i_2=0}^{M_2} \cdots \sum_{i_s=0}^{M_S} z_{i_1\cdots i_S} = 1, \quad i_1 = 1, 2, \ldots, M_1, \\ &\sum_{i_1=0}^{M_1} \cdots \sum_{i_s=0}^{M_S} z_{i_1\cdots i_S} = 1, \quad i_2 = 1, 2, \ldots, M_2, \\ &\sum_{i_1=0}^{M_1} \cdots \sum_{i_{s-1}=0}^{M_{S-1}} z_{i_1\cdots i_S} = 1, \quad i_S = 1, 2, \ldots, M_S. \end{aligned} \tag{4}$$

The multipliers Lagrange u_r, $r = S, S-1, \ldots, 3$ and the constraints of the above equations are defined in relation to the cost function. So, the following r "loose" subproblem arises [29]:

$$\begin{aligned} w^r_{i_1 \cdots i_r} &= \sum_{i_{r+1}=0}^{M_{r+1}} \cdots \sum_{i_S=0}^{M_S} z_{i_1 \ldots i_S} = \sum_{i_{r+1}=0}^{M_{r+1}} w^{r+1}_{i_1 \cdots i_{r+1}}, \\ d^{i_1 \ldots i_r} &= \max_{i_{r+1}} \cdots \max_{i_s} \left(c_{i_1 \cdots i_S} + u_{(r+1)_{i_{r+1}}} \cdots + u_{S_{i_s}} \right) \\ &= \max_{i_{r+1}} \left(d^{r+1}{}_{i_1 \ldots i_{r+1}} + u_{(r+1)_{i_{r+1}}} \right). \end{aligned} \tag{5}$$

Obviously, we have

$$d^S_{i_1 \cdots i_S} = c_{i_1 \ldots i_S}. \tag{6}$$

The r subproblem can be written as follows:

$$\max_{w_{i_1 - i_r}} \sum_{i_1=0}^{M_1} \sum_{i_2=0}^{M_2} \cdots \sum_{i_r=0}^{M_r} d^{i_{i_1 \ldots i_r}} w^{r_r}_{i_1 \ldots i_r} - \sum_{i_{r+1}=0}^{M_{r+1}} u_{(r+1)_{i_{r+1}}} \cdots - \sum_{i_s=0}^{M_S} u_{S_{i_S}}. \tag{7}$$

Given the fact that [30]

$$\begin{aligned} \sum_{i_2=0}^{M_2} \cdots \sum_{i_s=0}^{M_S} w^r_{i_1 \ldots i_r} &= 1, \quad i_1 = 1, 2, \ldots, M_1, \\ \sum_{i_1=0}^{M_1} \cdots \sum_{i_s=0}^{M_S} w^r_{i_1 \ldots i_r} &= 1, \quad i_2 = 1, 2, \ldots, M_2, \\ \sum_{i_1=0}^{M_1} \cdots \sum_{i_{r-1}=0}^{M_{S-1}} w^r_{i_1 \ldots i_r} &= 1, \quad i_r = 1, 2, \ldots, M_r. \end{aligned} \tag{8}$$

So for a given set of Lagrange multipliers, the r subproblem is a generalized assignment problem, where $r \leq S$. We defined the binary problem because Lagrange multipliers will impose a kind of "punishment" on the relaxed constraints violated by the solution.

Because anonymizing the data set several times is not enough to protect the data from a solid and well-prepared attacker, for example, in an n-element database, a specific feature knower of n−1 objects can easily infer the value of the individual attribute that remains, and in this research, we use differential privacy, which is an interactive method that protects data, even from attackers with prior knowledge of it [31].

Given $e > 0$, a randomized function M yields e-differential privacy, if for every data set x, x' with $x \sim x'$ and every S ⊆ RM, where RM is the set of values of M [32, 33].

$$P[M(x) \in S] \leq e^{\varepsilon} \cdot P[M(x') \in S]. \tag{9}$$

As e we consider a small, not negligible, positive number, usually in the interval (0.01, ln2), the lower the price, the greater the protection of records. The definition ceases to be useful if $e < 1/n$. We also consider n as universally known information. We observe that the relation can be written equivalently as follows:

$$P[M(x') \in S] \leq e^{\varepsilon} \cdot P[M(x) \in S], \tag{10}$$

due to the symmetry resulting from the definition of the proximity of the bases. The concept of differential privacy assures us that the attacker cannot deduce from the image of M, most likely, if the data from a single record have changed. In some cases, it is helpful to consider a generalization of the definition.

$$P[M(x) \in S] \leq e^{\varepsilon} \cdot P[M(x') \in S] + \delta. \tag{11}$$

The higher the $d > 0$, the easier it is for an attacker to distinguish which base is x' and x. The initial definition (with $d = 0$) is safer. In short, term d represents the possibility that some people may lose more privacy than others and that the multiplication barrier does not apply to everyone. If d is too small, this risk is too small [34]. An overview of how differential privacy is used is shown in Figure 1.

In general, it is true that even a mechanism $M : X \text{ -} \longrightarrow B$ provides e-differential privacy. Then, for each function f, the composition $f \circ M$ maintains e-differential privacy. And this is true whether we have a sequential or adaptive composition as they will maintain $(\varepsilon_1 + \varepsilon_2)$-differential privacy. Even in the case of advanced composition, for each ε, δ, $\delta' \geq 0$, the mechanism created by the adaptive synthesis of k mechanisms with (ε, δ)-differential privacy provides $(\varepsilon', k\delta + \delta')$-differential privacy with

$$\varepsilon' = \sqrt{2k \log\left(\frac{1}{\delta'}\right)}\varepsilon + k\varepsilon\left(e^{\varepsilon} - 1\right). \tag{12}$$

The above synthesis theorems cover both the repetitive application of differential privacy mechanisms in the same database and their repetitive application in different databases that may, however, contain information related to a specific record.

4. Use Case

For the modeling of the proposed system, a specialized differential privacy scenario was implemented with data derived from sensor fusion. It should be emphasized that the architecture of the models managing the randomization mechanisms is entirely different from that of the generalization mechanisms. Most algorithms use the technique to accept a set of data and return an anonymized version of it. However, the use of interactive techniques requires different modeling, and, for a question posed by a third party in the athlete's information fusion database, the administrator chooses the amount of privacy they wish to convey. The data are processed, noise is added, and the analyzer returns the result.

In the following modeling, we mainly use the Laplace mechanism, which satisfies a dynamic differential privacy criterion to implement different levels of privacy in the information distributed to third parties [35].

Let q be a set of values R, and let Δ be its l_1-sensitivity. Then, the mechanism [20, 33]

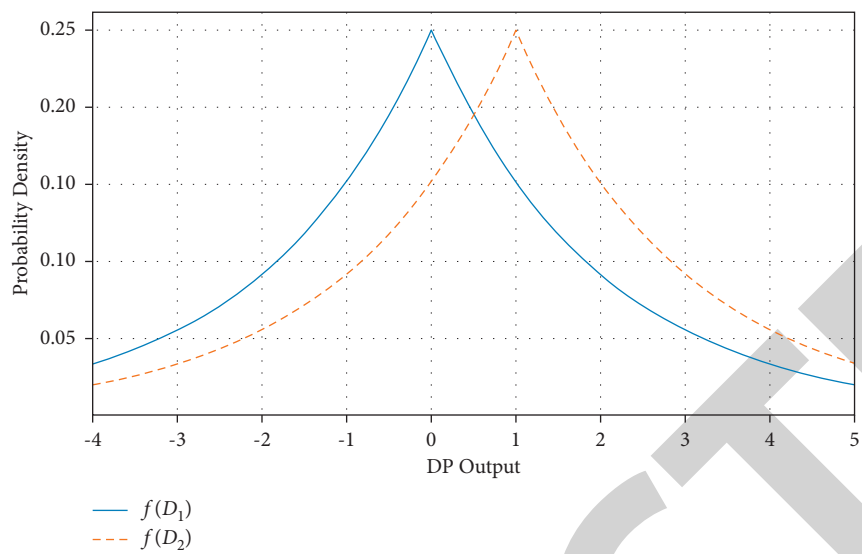


Figure 1: Differential privacy (probability density vs. differential privacy output).

$$M(x) = q(x) + z, \tag{13}$$

with $z \sim \mathrm{Lap}(\Delta|\varepsilon)$, provides e-differential privacy.

The size of the noise depends on the type of query and the selection of e. So for a counting query we want order noise $\sim \mathrm{Lap}(1|e)$, while the smaller the value of ε, the more inaccurate the result. We introduce the x-database with the athlete's heart rate and query the average heart rate [36].

$$q(x) = \frac{\sum_{i=1}^{n} x_i}{n}, \tag{14}$$

with $x_i \in [0, x_{\max}]$. If we use a neighborhood relation of type $|x_i - x_i'| \leq x_{\max}$, then the sensitivity of the query will be

$$\Delta = \max_{x \sim x'} |q(x) - q(x\prime)| = \frac{1}{n} \max_{x_i, x_i'} |x_i - x_i'| i \in [1, n]. \tag{15}$$

So we have

$$\Delta = \frac{x_{\max}}{n}. \tag{16}$$

According to the above, the mechanism becomes

$$M(x) = \frac{\sum_{i=1}^{n} x_i}{n} + \mathrm{Lap}\left(\frac{x_{\max}}{n\varepsilon}\right), \tag{17}$$

and will maintain e-differential privacy. We observe that the magnitude of the noise resulting from the Laplace mechanism is inversely proportional to the number n of recordings, which is to be expected since, intuitively, we expect better privacy if the size of the base is large.

The probability density function for mean $\mu = 0$ is

$$f(x|0, b) = \frac{1}{2b} e^{-|x|/b}. \tag{18}$$

So we have the cumulative distribution function

$$\begin{aligned} F(x) &= \int_{-\infty}^{x} f(u) \mathrm{d}u, \\ &= \int_{-\infty}^{x} \frac{1}{2b} e^{-|u|/b} \mathrm{d}u, \\ &= \begin{cases} \frac{1}{2} e^{x/b}, & x < 0, \\ 1 - \frac{1}{2} e^{-x/b}, & x \geq 0. \end{cases} \end{aligned} \tag{19}$$

The inverse function is

$$F^{-1}(x) = \begin{cases} b \cdot \ln(2x), & 0 < x < \frac{1}{2}, \\ -b \cdot \ln(2 - 2x), & \frac{1}{2} \leq x \leq 1. \end{cases} \tag{20}$$

Setting $u = x$ - 1/2, we end up with a generator of random variables [32].

$$X = -b \cdot \mathrm{sgn}(u) \ln(1 - 2|u|) u \in \left(-\frac{1}{2}, \frac{1}{2}\right]. \tag{21}$$

Thus, by selecting random variables u from the uniform distribution in the interval $[-0.5, 0.5]$, the random variable X will belong to the Laplace distribution with scale parameter b. We construct two different functions. The relation that connects e with the parameter b is

$$b = \frac{\Delta f}{\varepsilon}, \tag{22}$$

with Δf denoting the sensitivity of a function $f\colon X \longrightarrow \mathbb{R}^k$.

$$\Delta f = \max_{x \sim x'} f(x) - f(x'), \tag{23}$$

where x, x' are two adjacent databases.

To prove the above, we apply the proposed framework to the classic query experiment to find the mean value of a sensitive, numerical attribute. Specifically, we present in detail the methodology for finding the mean value of the heartbeat feature.

The heartbeat can be an indication of a person's physical condition. The average resting heart rate is between 70 and 75 beats per minute. People who do regular aerobic exercise reach 50 to 60 beats per minute. Professional athletes can have only 30 to 35 beats per minute, while people with poor fitness can go 90 or 100 beats per minute.

So we will have

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} b_i. \tag{24}$$

Athlete A's score values range from $[30, b_{\max}]$. We notice that

$$\left|b_i - b_i'\right| \le b_{\max} - 30. \tag{25}$$

Therefore, the sensitivity can be calculated as follows:

$$\Delta_f = \max_{x\sim x'}\left|q(x) - q(x')\right| = \frac{1}{n}\max_{x_i, x_i'}\left|b_i - b_i'\right| i \in [1, n]. \tag{26}$$

So we have

$$\Delta_f = \frac{b_{\max} - 30}{n}. \tag{27}$$

We know that the mechanism

$$M(x) = \frac{\sum_{i=1}^{n} b_i}{n} + \mathrm{Lap}\left(\frac{b_{\max} - 5}{n\varepsilon}\right), \tag{28}$$

will maintain e-differential privacy. Applying the formula to our calculations, we get the results for the average grade.

When $e = 1$, the average grade = 33.74; when $e = 0.1$, the average grade = 33.73; and when $e = 00.1$, the average grade = 33.81, etc.

Another option is to add Laplace noise to each point and then calculate their average value.

5. Conclusions

To secure athletes' data collected and analyzed by sports wearables, this paper presents an innovative and highly flexible privacy-preserving sports wearable data fusion framework. It is an advanced methodology for protecting privacy in synthesized databases. Specifically, the procedure is based on the Lagrange relaxation method for the problem of multiple assignments and the synthesis of information from numerous sensors. Data are secured using a flexible, adaptive differential privacy system. Using Laplace noise allows access to databases with personal information, ensuring that this information will remain personal without a third party being able to reveal the identity of the athlete who provided the data in question.

This technique is an initial privacy-preserving framework for maintaining data mining confidentiality. When data are transferred or shared between different parties, it is mandatory to provide security so that other parties do not know what information is being shared between the original parts, identifying the users. In general, this methodology helps hide the knowledge of sports data output as the output data are valuable and private, thus contributing to the shielding of defense mechanisms related to the sports industry.

Data Availability

The data used in this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] B. Ma, S. Nie, M. Ji, J. Song, and W. Wang, "Research and analysis of sports training real-time monitoring system based on mobile artificial intelligence terminal," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID e8879616, 10 pages, 2020.

[2] C. Lu, "Design of information system using visual studio for auxiliary sports under computer big data," in *Proceedings of the 2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, pp. 757–760, Dalian, China, August 2021.

[3] G. Qin, N. Ding, and J. Yan, "Research on the intelligent system of computer big data technology to guide athletes," in *Proceedings of the 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*, pp. 203–207, Changchun, China, February 2022.

[4] D. Patel, D. Shah, and M. Shah, "The intertwine of brain and body: a quantitative analysis on how big data influences the system of sports," *Annals of Data Science*, vol. 7, no. 1, pp. 1–16, 2020.

[5] T. Aira, K. Salin, T. Vasankari et al., "Training volume and intensity of physical activity among young athletes: the health promoting sports club (HPSC) study," *Advances in Physical Education*, vol. 09, no. 04, pp. 270–287, 2019.

[6] R. Jiang, "Application of computer sports big data informatization construction comprehensive intelligent information system," in *Proceedings of the 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*, pp. 212–216, Changchun, China, February 2022.

[7] U. Granacher and R. Borde, "Effects of sport-specific training during the early stages of long-term athlete development on physical fitness, body composition, cognitive, and academic performances," *Frontiers in Physiology*, vol. 8, p. 810, 2017.

[8] Y. Zheng, "Methodologies for cross-domain data fusion: an overview," *IEEE Transactions on Big Data*, vol. 1, no. 1, pp. 16–34, 2015.

[9] D. M. El-Din, A. E. Hassanien, and E. E. Hassanien, "Information integrity for multi-sensors data fusion in smart mobility," in *Toward Social Internet of Things (SIoT): Enabling Technologies, Architectures and Applications*, A. E. Hassanien, R. Bhatnagar, N. E. M. Khalifa, and M. H. N. Taha, Eds., Springer International Publishing, New York, NY, USA, 2020.

[10] L. Jia and Q. Wang, "Establishment and application of comprehensive evaluation model for athletes," in *Proceedings*

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9869278, 1 page
https://doi.org/10.1155/2023/9869278

Retraction

Retracted: Online Anomaly Detection of Industrial IoT Based on Hybrid Machine Learning Architecture

Computational Intelligence and Neuroscience

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] J. Guo and Y. Shen, "Online Anomaly Detection of Industrial IoT Based on Hybrid Machine Learning Architecture," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 8568917, 10 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 8568917, 10 pages
https://doi.org/10.1155/2022/8568917

Research Article

Online Anomaly Detection of Industrial IoT Based on Hybrid Machine Learning Architecture

Jia Guo[1] **and Yue Shen**[2]

[1]*Zhengzhou Sias University, Department of Electronics and Information, Zhengzhou 451100, China*
[2]*Henan Geology Mineral College, Zhengzhou 451464, China*

Correspondence should be addressed to Jia Guo; guojia_edu@126.com

Received 25 February 2022; Revised 3 March 2022; Accepted 7 March 2022; Published 30 April 2022

Academic Editor: Konstantinos Demertzis

Industrial IoT (IIoT) in Industry 4.0 integrates everything at the level of information technology with the level of technology of operation and aims to improve Business to Business (B2B) services (from production to public services). It includes Machine to Machine (M2M) interaction either for process control (e.g., factory processes, fleet tracking) or as part of self-organizing cyber-physical distributed control systems without human intervention. A critical factor in completing the abovementioned actions is the development of intelligent software systems in the context of automatic control of the business environment, with the ability to analyze in real-time the existing equipment through the available interfaces (hardware-in-the-loop). In this spirit, this paper presents an advanced intelligent approach to real-time monitoring of the operation of industrial equipment. A hybrid novel methodology that combines memory neural networks is used, and Bayesian methods that examine a variety of characteristic quantities of vibration signals that are exported in the field of time, with the aim of real-time detection of abnormalities in active IIoT equipment are also used.

1. Introduction

The industry sector within Industry 4.0 introduces and uses the Internet of Things in all its functions, which has contributed to the implementation of severe innovative leaps [1, 2]. Each part of the industrial ecosystem that participates in the production process is accompanied by a massive volume of generated data, which describe its regular operation while also containing frequent anomalies related to every day or improper use [3]. The ability to detect abnormalities in the equipment process in real-time has a significant impact on the overall operation of the industrial environment while offering stability and reliability during its management. [4].

This possibility, especially when attempted without human supervision, is complex as it depends on many factors. Initially, the key to this process is to determine the exact boundary between normal and abnormal operations, which presents the most significant difficulty in this analysis [5]. To select the appropriate anomaly detection technique, several factors must be weighed. The most important is the nature of the data produced, i e., binary, continuous in time, or discrete values and their relationship. The availability of data and the determination of the type of deviant behavior should also be specified. More specifically, it should be determined whether it refers to an anomaly of a point pattern or whether the specific behavior is under conditions [6].

Point anomaly refers to detecting a point whose value differs from the rest of the data set. The type of data set to which this anomaly refers is the one where its values have a specific range and, in regular operation, do not exceed the maximum and minimum value that it determines. Therefore, a sharp increase or decrease in a value that simultaneously exceeds these limits can be described as an abnormal behavior of the equipment or operation. Such anomalies are best detected before processing the data set and analyzing it. [7].

Another category of data anomalies is the one that presents a specific and repetitive pattern over time, including fluctuations in their values. Any deviation from this pattern can be considered an anomaly [8].

Finally, some data behaviors are considered anomalies under conditions. A typical example is a bottleneck in an industrial network regarded as normal behavior during industrial working hours, such as in the morning or noon, but late at night is not normal and is an anomaly for network traffic flow. So, in this type of anomaly, the data values and the conditions that characterize them are essential [9, 10].

The industrial systems that are usually involved in the production process of a unit show a gradual deviation from the regular operation and not instantaneous failure. This makes it possible to predict deviant behavior, as there is a time interval between faults. As it is easily understood, the development of methods to detect anomalies is significant. They reduce the chances of an unexpected failure of the industrial system, which can have huge costs. [11].

All the abovementioned reasons led to the orientation of many research works on methods based on data analysis to detect abnormalities in the operation of industrial equipment. Based on the literature, detecting anomalies in a function can be done with statistical methods, but machine learning algorithms are also of particular interest.

2. Literature Review

The research community is continuously trying to implement machine learning technologies to take advantage of its unique characteristics related to anomaly detection, especially in environments that tend to produce extremely high amounts of data, like IIoT [1]. This section presents some recent studies in this field, some of which propose the relatively new Federated Learning framework.

Shah et al. [3] in 2018 investigated various machine learning algorithms for detecting anomalies in IIoT data from motor equipment. They analyzed the sensor data on fuel consumption, engine load, and oil pressure to determine when a certain engine exhibits anomalous behavior and may fail. To discover aberrations in machine behavior, they used multivariate linear regression, Gaussian mixture models, and time-series data analysis. To conclude their research, they employed simple statistical analysis to answer some of these scenarios, while machine learning algorithms were applied in others.

Zhou et al. [4] offered an overview of existing network anomaly detection algorithms as well as a brief description of the requirements and obstacles in IIoT network security. They also presented alternative anomaly detection approaches specifically suitable to IIoT networks. Such methods take advantage of the physical world's deterministic properties to find anomalies in observed behavior. The approaches based on specification descriptions and physical process modeling consider the operating dynamics of the underlying physical system and discover abnormalities from essential system characteristics. These techniques can be used in conjunction with cyber security techniques that detect anomalies caused by data modification. Finally, they proposed that for IIoT network anomaly detection, integrated cyber security and physical state estimate techniques would be more successful.

Four popular SCADA IIoT protocols, as well as their security flaws, were described by Zolanvari et al. [12]. Following that, they conducted a risk audit of the most significant and common security issues in IIoT systems and how machine learning-based solutions could assist in mitigating them. They showed a use case that included a real-world testbed constructed to perform cyber-attacks and designed an intrusion detection system. They used real-world cyber-attacks against this system to demonstrate how machine learning could address the identified gap by effectively handling them and measured the performance using representative measures to provide a fair assessment of the methods' effectiveness. Finally, feature priority ranking was investigated to emphasize the most important characteristics in separating malicious from normal traffic.

To fight against malicious actors, Yan et al. [13] presented a Hinge classification algorithm based on mini-batch gradient descent with an adaptive learning rate and momentum (HCA-MBGDALRM) in 2020. They stated that their method outperformed established approaches in terms of scalability and speed for deep network training. They also fixed the data skew issue during the shuffle phase. They developed a parallel framework for HCA-MBGDALRM to speed up the analysis of large traffic data volumes and enable IIoT safety improvements. Finally, they found that their technique increased the model's training efficiency and accuracy, ensuring the reliability of big data networking in IIoT.

Liu et al. [14] proposed the federated learning (FL) framework, which allows for decentralized edge devices to collaborate to train a deep anomaly detection (DAD) model, improving its generalization capabilities. They also developed a convolutional neural network long short-term memory (CNN- LSTM) model for detecting anomalies. They used their units to capture fine-grained characteristics, while maintaining the LSTM unit's benefits in forecasting time series data. The findings confirmed that their approach could maintain appropriate precision across all data sets and that the technique could enhance information flow 300 times without making any mistakes by reducing gradients. They even proposed a gradient compression mechanism to lower communication expenses and increase communication efficiency in order to accomplish real-time and lightweight anomaly detection.

Finally, Wang et al. [15] in 2021 suggested an anomaly detection system for IIoT, which utilized federated learning to improve confidentiality for various IIoT applications. They used this method to create a universal anomaly detection concept, training each local model using the deep reinforcement learning technique without aggregating local data sets to protect confidentiality. Their solution had the advantage of not requiring local data sets during federated learning, which decreased the risk of privacy compromise. The federated deep reinforcement learning algorithm then adequately identified anomalous users. The proposed

technique demonstrated positive outcomes in diverse IIoT scenarios, according to the validation studies.

The proposed approach of this work aims at detecting signs of equipment behavior change in real-time to identify anomalies before the collapse of a system, which may be due to damage or cyber-attacks [16].

3. Proposed Methodology

The primary idea of the proposed methodology is based on the logic of the P–F curve, which gives a representative picture of the behavior of equipment in regular operation and in that which presents abnormalities [8]. Therefore, the aim is to predict the point at which the behavior of the equipment begins to change (point P) much earlier than would be perceived by the person in charge of the operation of the equipment from indications that would appear. The technological innovations related to sensors contribute to achieving the abovementioned goal for monitoring the production process and then recording the data necessary for the subsequent analysis by the proposed hybrid machine learning system [7].

More specifically, the proposed approach contains three basic algorithms. The first implements a long short-term memory (LSTM) neural network [17], the next implements the time-domain feature extraction process, and the last is the Bayesian online changepoint detection [18]. Initially, the measurements recorded in real-time by the sensors are taken as input from the LSTM neural network, and the predicted values for the exact quantities are generated in a period predetermined by the network composition. These values then feed the algorithm to extract the features needed for the upcoming data analysis. It is also important to note that the data taken as input to this process are not the sum of the data of each size but are selected from a time-varying fixed-size window [19]. Finally, the signal resulting from the output of the power supplies the change point detection algorithm, which indicates through a graph the probability that a point is a change point. At the same time, through the probabilities that it calculates, it predicts the future state of the equipment in the period above determined by the neural network. This procedure is followed because the change point detection algorithm is sensitive to noise and should be subtracted from the signal to be inserted into it to output information at a higher level than the original data and reduce the uncertainty it initially includes [20].

3.1. Long Short-Term Memory Neural Network. LSTM neural networks can retrieve information from a significant number of past time steps, providing satisfactory results in problems with serial data and especially time series. It is a chain of similar neural devices, but each consists of four interact levels. The synthesis and function of the basic structures of an LSTM cell can be attributed to steps as follows [17, 21, 22]:

The output of the last cell and the current input to the cell are combined in one vector, where it concerns all the data to be processed:

$$[h_{t-1} + x_t]. \tag{1}$$

The above vector goes through the "forget gate," and the following function is generated:

$$f_t = \sigma\left(W_f * [h_{t-1}, x_t] + b_f\right). \tag{2}$$

This function is multiplied by the previous memory state so that we obtain the following:

$$[C_{t-1} * f_t]. \tag{3}$$

The vector $[h_{t-1} + x_t]$ passes through the "input gate," and the following function is generated:

$$i_t = \sigma\left(W_i * [h_{t-1}, x_t] + b_i\right). \tag{4}$$

The same vector passes through the tanh function and produces the following:

$$\widetilde{C}_t = \tanh\left(W_C * [h_{t-1}, x_t] + b_C\right). \tag{5}$$

The results of the previous two steps are multiplied by each other to obtain the following:

$$\left[i_t * \widetilde{C}_t\right]. \tag{6}$$

Adding the results produced by the multiplications in the above steps results in a new memory state:

$$C_t = C_{t-1} * f_t + i_t * \widetilde{C}_t. \tag{7}$$

The vector $[h_{t-1} + x_t]$ passes through the "output gate," and the following function is generated:

$$o_t = \sigma\left(W_o * [h_{t-1}, x_t] + b_o\right). \tag{8}$$

The new memory state operates as a tanh function, and the result is multiplied by the above function producing the new cell output:

$$h_t = o_t * \tanh\left(C_t\right). \tag{9}$$

3.2. Feature Extraction. The data collected by sensors usually require a pre-treatment to remove the noise contained in the signal due to their use and minimize the uncertainty caused by them. In addition, in this way, information is produced with sufficient accuracy so that the analysis of the data subsequently produces reliable results. Especially in vibration signals, such as those made mainly in industry, the extraction of characteristic quantities is necessary when the data analysis involves the detection of errors and the prediction of various quantities [23].

The method used to extract feature sizes is done in the time-domain features extraction. These characteristics refer to the mean, which calculates the quotient resulting from the sum of the signal values and their number. The root mean square (RMS) calculates the square root of the mean value of the signal raised to the square. This size increases gradually as an error develops within the signal to be studied, but it cannot provide information at the initial stage of error development [20, 24, 25]. In addition, variance is a quantity

that indicates the scatter of the signal using the mean as a reference. In contrast, standard deviation (std) determines the square root of the signal variance. More specific sizes, capable of delivering more information, are kurtosis and skewness, which process the probability density function of the signal. More specifically, kurtosis calculates the maximum value of the function and indicates whether the signal can respond immediately to a change. Under normal conditions, the kurtosis emitted by a vibration signal is approximately equal to a pre-agreed value, e g., three. At the same time, if there are errors in it, then the probability density function changes, and therefore, the value of the curvature is greater than that of regular operation. Accordingly, skewness is a quantity obtained from the mean value of the probability density function and is used to indicate whether the vibration signal is negatively or positively skewed. In a signal with normal distribution, the skewness has zero value. Still, if it is disturbed due to errors, then it will receive either a negative or a positive value depending on the skewness it will present. In addition, it is worth noting that the abovementioned two values can be applied to signals that are not purely continuous in time (stationary) in contrast to characteristics such as mean and standard deviation [26, 27].

Another quantity that can be deduced from the probability density function for vibration signals is entropy, which calculates the histogram of the above function and indicates the magnitude of the randomness and uncertainty of the signal. Finally, the lower and upper bound histograms belong to the same category, which calculates the maximum and minimum values of the probability density function, respectively [18, 28].

The abovementioned are the appropriate characteristics for processing vibration signals coming from the industrial sector. Also, the attributes in question result in a value calculated from the total of the data studied. However, in this approach, the feature extraction process is rolling, and the result obtained is a curve with the values calculated at each step. In other words, this process uses a window of a specific size so that the calculation of features is not done from the whole data set but from a subset of a fixed size that moves over time. Therefore, the analysis of the exported characteristics includes values from previous times and the current one. The amount of these historical data is determined by the window set for calculating each attribute. The use of the window and therefore, the historical data result in the extraction of information at a higher level and greater computational efficiency [7, 24, 29].

The work used sliding windows, and the exact window size for each feature was determined after testing to achieve the best result in terms of information to be extracted from the feature display.

3.3. Bayesian Online Changepoint Detection. A time series is a collection of observations in chronological order. These data are large, so they take up more memory, are multidimensional, and are constantly updated. Another characteristic of them is abrupt changes in their structure, such as a jump in a much higher value than the previous one or a different behavior in data distribution. Those points that change behavior is called change points and essentially split the data into homogeneous parts. Detection abnormalities in a state of operation is a process in which abrupt changes in serial data are identified and performed in real-time or afterward. Most algorithms with "Bayesian" logic focus on the fragmentation of the data set and on techniques that produce results from their subsequent analysis. Still, the algorithm used in this study focuses on identifying the cause of the problem. During execution, it creates a distribution of the next value in the data sequence, taking into account only the values that have been recorded so far. This approach is suitable for detecting points of change in time series due to its ease in quantifying the probability that a position is a point of difference [10, 21, 30].

The quantities to be studied are a series of time-determined observations divided into various dissimilar and non-overlapping areas with a specific length. At the same time, the boundaries between them are the changepoints. It is also considered that these observations are independent and uniformly distributed random variables with probability distribution $P(x_t|n_\rho)$ where n_ρ are independent and similarly distributed random variables. In addition, a grouping of observations between time instant a and b is denoted by $x_{\mathrm{a\,b}}$ and the preset probability distribution in the space between two change points with $P_{\mathrm{gap}}(g)$.

This approach estimates the subsequent probability distribution at the current data length r_t at time t. Data length r_t is a time-dependent function, which is zeroed when a state change occurs, i e., it encounters a change point and refers to the data set from the most recent change point to that time point. In addition, the observations concerning a data length r_t are denoted by (r), while if the data length is zero $(r = 0)$, then they are denoted by $x^{(r)}$.

To predict the probability distribution at the current data length, the ex-post probability distribution must first be calculated retrospectively, and the marginal prediction distribution through the following formulas must be integrated into it [24, 31, 32]:

$$P(x_{t+1} \mid x_{1:t}) = \sum_{r_t} P\left(x_{t+1} \mid x_t^{(r)}, r_t\right) P(r_t \mid x_{1:t}) P(r_t \mid x_{1:t}) = \frac{P(r_t, x_{1:t})}{P(x_{1:t})} P(r_t, x_{1:t}) = \sum_{r_{t-1}} P(r_t, r_{t-1}, x_{1:t}). \tag{10}$$

The formula $P(r_t, x_{1:t})$ calculates the probability density function in the current data length retrospectively to calculate the ex-post probability distribution. Finally, it is worth noting that the forecast distribution depends only on

recent data (r). Therefore, the probability density function can be calculated retrospectively based on the current data length r_t, given r_{t-1}, and the predictive distribution results from the new value observed, given the values observed so far.

To calculate the abovementioned quantities and formulate retrospective formulas, it is necessary to define the limit conditions based on two considerations. In the first case, a point change has occurred before the first value of the data to be studied, and therefore, the probability function is zeroed for the initial data length. In the second case, on the other hand, the study is done on a recent subset of data, and the boundary condition is formed by the normalized survival function, which indicates the time at the end of which one or more events occur. The conditions in the mathematical form are shown below [25, 33, 34]:

$$P(r_0 = 0) = 1$$
$$P(r_0 = \tau) = \frac{1}{Z}\widetilde{S}(\tau), \tag{11}$$

where Z is the normalized constant and $\widetilde{S}(\tau) = \sum_{t=r+1}^{\infty} P_{\text{gap}}(g = t)$.

The computational efficiency of the algorithm is due to the form of the probability function that there is a point of change based on previous data. This function is zero everywhere except when the data length increases as a new value are added to it and when a unique change point is observed. The function of this probability is shown below [18], [35]:

$$P(r_t|r_{t-1}) = \begin{cases} H(r_{t-1}+1), r_t = 0 \\ 1 - H(r_{t-1}+1), r_t = r_{t-1}+1, \\ 0, \text{elsewere} \end{cases} \tag{12}$$

where H(t) is the hazard function and represents which pieces of data have a higher or lower probability of an event occurring and is equal to

$$H(\tau) = \frac{P_{gap}(g = \tau)}{\sum_{t=\tau}^{\infty} P_{gap}(g = t)}. \tag{13}$$

The exponential models are a handy tool for detecting anomalies and essentially for the algorithm described in this work. They are easy to use because they can offer a set of parametric probability distributions and statistical quantities, which can be calculated during data collection. The probability format based on these models is reported for completeness and is shown in the following equations [36], [37]:

$$H(\tau) = \frac{P_{gap}(g = \tau)}{\sum_{t=\tau}^{\infty} P_{gap}(g = t)}$$
$$P(x|n) = h(x)\exp\left(n^T U(x) - A(n)\right) \tag{14}$$
$$A(n) = \log \int \mathrm{d}n h(x)\exp\left(n^T U(x)\right)$$

4. Data Set, Scenarios, and Results

Data from an industrial plant that performs cold rolling on metals were used for the present study. This process substantially reduces the thickness of the metal to the optimum smaller thickness with a perfectly smooth surface or reshapes it through two rollers, which rotate in the opposite direction from that of the metal. The metal temperature must be lower than where the metal recrystallizes to do this process.

Ten sensors have been installed in the cold rolling equipment to collect data on vibrations to implement the experiments. Also, in the cold rolling unit, there is a sensor that measures the speed of the motor and another that measures its current. In this study, we are only interested in the data collected by the former. In more detail, these sensors record values for four different variables every ten seconds for vibration data. These are the acceleration, the state of the rollers in terms of resistance and vibration (overall bearing), the abrupt change of state (shock), and the speed (velocity). These measurements cover ten months, and the number of files created by different sensors was ten in number, with a size ranging from 870,000 to 990,000 particles of data.

Each of the re-created files contains six columns, the first of which refers to the name of the roller, depending on the position of the sensor in it. The second column has the date and time in timestamp UTC so that it is expressed globally and not locally, and the other four columns refer to the values of the quantities produced by the sensors, and these data are time series. From the statistical analysis of the data, it initially appears that the magnitude "shock" describes the existence or not of a significant disturbance in the operation of the device. Its non-existence is illustrated with a zero value while the opposite state with a positive value. The acceleration takes only positive values with a minimum value close to zero while the maximum does not exceed 7. Similar behavior is shown by the second size overall bearing, with its maximum value not exceeding 8. Finally, velocity indicates that the minimum price is close to 0, but its maximum value is much higher than the two previous sizes. An indicative representation of the sizes in question is presented in Figure 1.

The diagrams above show the data for each of the quantities recorded by the sensor. From the figure regarding acceleration, we observe a value that differs from the rest but allows us to see the variation of the other values, as there is no considerable difference between the maximum and the rest. If we capture some values before this maximum value, as shown in the velocity figure on the right, we will notice a difference between the speed data, and they are not zero. In the case of velocity, on the other hand, it is evident that the fluctuation of the values does not exist, and they are all presented as a straight line very close to zero, which is interrupted by a vertical that represents the maximum value. The high-velocity value probably comes from a disturbance in the sensor environment. It is not interpreted in a natural way to be related to the behavior of the equipment [9, 29, 33].

The experimental process aims to detect anomalies in the industrial data. The diagrammatic representation of the experiment is shown in Figure 2.

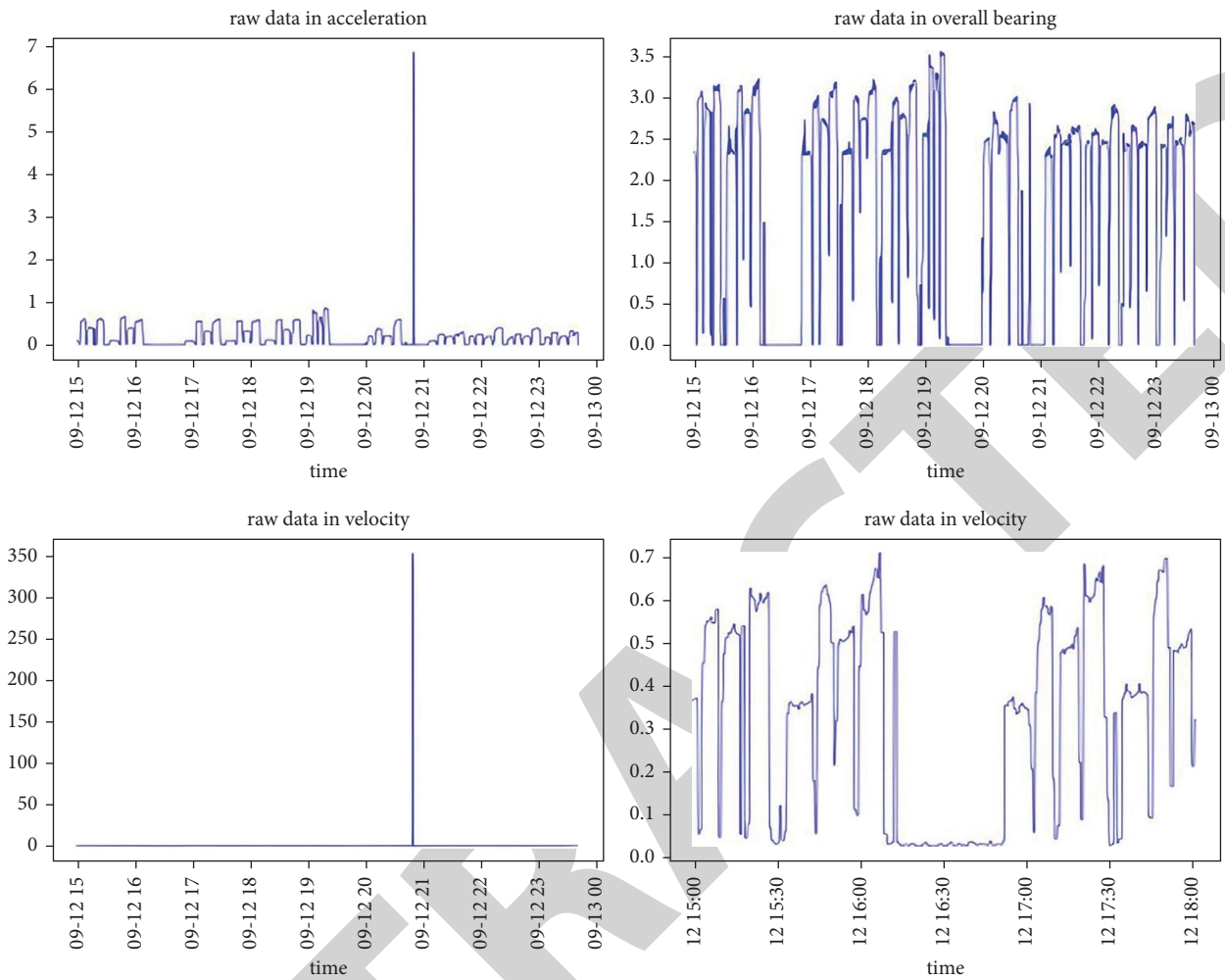


Figure 1: Depiction of the data set (random state).

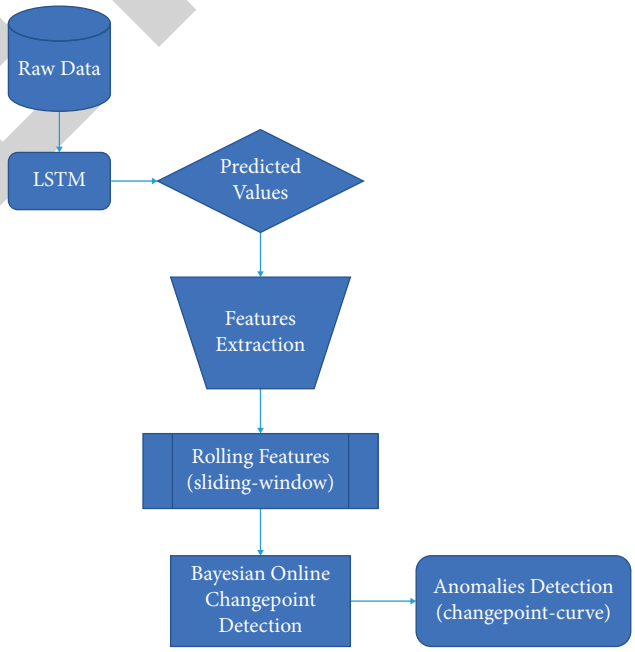


Figure 2: Flowchart of the proposed anomaly detection process.

Acceleration, overall bearing, and velocity values are predicted via the LSTM neural network during the experimental process. In the intermediate stage of the process, the characteristics are extracted from vibration signals (raw data) in the time domain. The procedure concerns the following sizes [17, 20, 21, 27]:

$$\begin{aligned}
\text{mean} &= \frac{\sum_{i=1}^{N} x_i}{N} \\
\text{RMS} &= \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \\
\text{STD} &= \sqrt{\frac{\sum_{i=1}^{N}(x_i - m)^2}{(N-1)\sigma^2}} \\
\text{var} &= \frac{\sum_{i=1}^{N}(x_i - m)^2}{(N-1)\sigma^2} \\
Ku &= \frac{\sum_{i=1}^{N}(x_i - m)^4}{(N-1)\sigma^4} \\
Sk &= \frac{\sum_{i=1}^{N}(x_i - m)^3}{(N-1)\sigma^3} \\
e(p) &= -\sum_{i=-1}^{n} p(z_i)\log_2 p(z_i) \\
h_U &= \max(x_i) + \frac{\Delta}{2} \\
h_L &= \max(x_i) - \frac{\Delta}{2},
\end{aligned} \tag{15}$$

which are used to detect differences between vibration signals.

Then, the feature extraction algorithm is applied, producing five rolling features. Specifically, let $\tau_0 \in T$ be the time of submission of the relevant duration query, then the scope of a rolling window with width ω and step δ for each $\tau \in T$ (with $\tau \geq t0$) extends [19, 30]:

$$\text{scopes}_s(\tau, \omega, \delta) = \begin{cases} [\tau - \omega + 1, \tau], \alpha v \tau \geq \tau_0 + \omega \wedge \text{mod}((\tau - \tau_0), \delta) = 0 \\ \text{scope}_s(\tau - 1, \omega, \delta), \alpha v \text{mod}((\tau - \tau_0), \delta) \neq 0 \\ [\tau_0, \tau], \alpha v \tau_0 \leq \tau < \tau_0 + \omega \wedge \text{mod}((\tau - \tau_0), \delta) = 0 \end{cases}, \tag{16}$$

where the magnitudes τ_0, $\tau \in T$ are expressed in time landmarks and ω, $\delta \in \mathbb{N}$ are expressed in a range of time intervals (ω, $\delta > 0$). For the sake of simplicity, the abovementioned definition implies that all-time quantities are expressed as natural numbers so that the function is calculated at distinct time points of T. Then, the window multiples result from the relation [19, 38]:

$$W_s(S, \tau, \omega, \delta) = \{s \in S(\tau) \colon s.A_\tau \in \text{scope}_s(\tau, \omega, \delta)\}. \tag{17}$$

Usually, step δ is the same size as the unit of time (e g., second) so that the window's progress is ideally in line with the corresponding time. Because δ<? is generally valid, the contents of two consecutive snapshots of the popup window overlap. Meanwhile, its contents remain unchanged until the function is applied again to the next pulse, after δ time. This is expressed by the retroactive expression in the middle branch of the function: the edges of the window change only at times specified in step δ. The third part of the function provides the possibility of initial "missing" windows immediately after the query when the range exceeds the time range of the current contents.

Since the range function is monotonous due to the evolution of time implies a homologous passing of the intervals and can be defined even for future moments. All the following elements of the current are covered, regardless of when and if they finally appear. For example, suppose the kurtosis attribute is exported with window = 25. In that case, the first value of this attribute will be extracted from the data to be studied in positions 1 to 25, while the second value from the data in positions 2 to 26 and so on. The term position means the order in which the data from the sensor have been recorded in this case.

Finally, for each time $\tau \in T$, the window connection operator returns the joining of pairs of blocks that appear in the respective window snapshots [19]:

$$\begin{aligned} S_1(\tau) \infty S_2(\tau) = \{(s_1, s_2, \tau_m, : s_1 \in W_1(S_1(\tau)), s_2 \in W_2(S_2(\tau)) \wedge \\ \wedge E(s_7, s_2) \wedge \tau_m = max(s_1, A_\tau, s_2, A_\tau)\}. \end{aligned} \tag{18}$$

It is essentially a sliding-window join process, separately for each stream. Each new input block of current S_1 is checked for connection condition E with all existing rolling window W_2 of the stream S_2. The exported results receive the most recent timeline displayed in the primary tuples pair if matching elements are found. This is, after all, the time indicated that restores the order of the final results of the generated data stream.

The Bayesian online changepoint detection algorithm [20] is then applied to each of the characteristics generated for each size studied to detect a change in equipment operation through feature analysis. This process can be captured in the following steps [17, 20, 39]:

Step 1. Initialization through following marginal conditions:

$$\begin{aligned} P(r_0) &= \widetilde{S}(r) \text{or} P(r_0 = 0) = 1 \\ v_1^{(0)} &= v_{\text{prior}} \\ X_1^{(0)} &= X_{\text{prior}}. \end{aligned} \tag{19}$$

Step 2. Observation of the following data value.

Step 3. Predictive probability estimation as follows:

$$\pi_t^{(r)} = P\left(x_t | v_t^{(r)}, X_t^{(r)}\right). \tag{20}$$

Step 4. Probability calculation as the current data length increases as shown below:

$$P(r_t = r_{t-1} + 1, x_{1:t}) = P(r_{t-1}, x_{1:t-1})\pi_t^{(r)}(1 - H(r_{t-1})). \tag{21}$$

Step 5. Calculation of probability for the existence of a point of change as follows:

$$P(r_t = 0, x_{1:t}) = \sum_{r_{t-1}} P(r_{t-1}, x_{1:t-1})\pi_t^{(r)} H(r_{t-1}). \tag{22}$$

Step 6. Probability calculation in the observation group, from the first to the time t:

$$P(x_{1:t}) = \sum_{r_t} P(r_t, x_{1:t}). \tag{23}$$

Step 7. The data length distribution is defined as follows:

$$P(r_t \mid x_{1:t}) = \frac{P(r_t, x_{1:t})}{P(x_{1:t})}. \tag{24}$$

Step 8. The parameter value as shown below:

$$\begin{aligned} v_{t+1}^{(0)} &= v_{\text{prior}} \\ X_{t+1}^{(0)} &= X_{\text{prior}} \\ v_{t+1}^{(r+1)} &= v_t^{(r)} + 1 \\ X_{t+1}^{(r+1)} &= X_t^{(r)} + u(x_t). \end{aligned} \tag{25}$$

Step 9. Calculation of the marginal forecast distribution:

$$P(x_{t+1} \mid x_{1:t}) = \sum_{r_t} P\left(x_{t+1} \mid x_t^{(r)}, r_t\right) P(r_t \mid x_{1:t}). \tag{26}$$

Step 10. Return to observe the next value.

The temporal and spatial complexity of the algorithm is linear about the amount of data to be processed and is calculated at each time point based on the data observed so far. The abovementioned description gives a clear picture of the operation and purpose of the algorithm, which calculates the quantities required to calculate the probability that each point is a change point (log-likelihood). In particular, the algorithm accepts as input the data in which it is desirable to detect anomalies and, as a result, is initially extracted for each time corresponding to the data set, the value of the probability to be a point of change. These values are used to create the diagram that presents the above information graphically. More specifically, the values of the probabilities that are calculated refer to the size log-likelihood, which has a logarithmic character and a negative sign. The vertical axis is on a logarithmic scale in the diagram formed, and the horizontal one represents time.

Another quantity that results from applying the algorithm is the position of the point most likely to be a point of change. Although the generated curve may have enough points with a sufficient probability value to show a state change, the algorithm returns the highest probability value, i e., the point where the curve is at the highest position on the chart. This approach returns to the time when behavior change is most likely. Finally, it returns the average values found in all previous time points from the algorithm's most probable for a state change. Similarly, the average size values studied and found after the aforementioned time are calculated.

The results of the above experiment are shown in the following diagrams. The first row of the figures shows the predicted values of the interest quantities compared to the actual values of these data. The mode of each feature is different, having the same input. Of course, there are several similarities in the attributes mean, std and rms. Therefore, the result of the algorithm shows several common points for the specific features. In addition to all the diagrams showing the result of the experiment, a clear picture of the process of detecting anomalies is demonstrated as the peaks of the curves are clear.

Below are the results for some data sets where the procedure of this experiment was applied. In each figure, the first column presents the initial data of each size from acceleration, overall bearing, and velocity for a data set. The following columns show the algorithm's result for each kurtosis feature, skewness, mean, std, and rms. These curves are the result to be evaluated in the experiment. The results show the recording of the time where the point of change of the equipment operates according to the algorithm is considered. The value forecast was made with a time limit of half an hour.

Finally, an important observation is that applying the algorithm to the characteristics mean, standard deviation, and rms gives better results about kurtosis and skewness. While they exhibit similar behavior, they are different for each data set, and this can be perceived from the time they identify as the most likely to occur abnormalities. In addition, the mean feature realizes earlier the malfunction of the equipment as it presents an anomaly at a previous time from the other elements for all the sizes studied.

5. Conclusions

The detection and timely assessment of abnormalities in the operation of the industrial ecosystem allows the detection of incidents and the corresponding identification of correlations and causal relationships with security incidents, which can significantly mitigate the effects of sophisticated cyber-attacks. In this spirit, a hybrid system of deep machine learning architecture was used to predict anomalies in the operation of industrial equipment. Specifically, the Bayesian online changepoint detection algorithm was used to detect anomalies, the time-domain features extraction process for feature extraction, and the long-short term memory neural network to predict the values of the magnitudes that reflect the correct or not the operation of the equipment. Sensors in

real-time record the data used. Because they contain a lot of noise, detecting anomalies without prior pre-treatment produces excellent uncertainty. If the detection is done after the proper pre-processing of the data, the results are distinguished with great accuracy.

Future developments of the work concern the development of more complex deep learning architectures to model the problem more fully in question. Procedures should also be studied to add other parameters as input and improve the model's accuracy and efficiency. At the same time, predictive analysis procedures for new methods that will fully automate extracting data characteristics and detecting anomalies should be studied.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Banafa, "2 the industrial Internet of Things (IIoT): challenges, requirements and benefits," in *Secure and Smart Internet of Things (IoT): Using Blockchain and AI*, pp. 7–12, River Publishers, Denmark, 2018.

[2] S. Schneider, "THE INDUSTRIAL INTERNET OF THINGS (IIoT)," in *Internet of Things and Data Analytics Handbook*, pp. 41–81, Wiley, Hoboken, New Jersey, U.S, 2017.

[3] G. Shah and A. Tiwari, "Anomaly detection in IIoT," in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, pp. 295–300, Goa India, January. 2018.

[4] L. Zhou and H. Guo, "Anomaly detection methods for IIoT networks," in *Proceedings of the 2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, pp. 214–219, Singapore, July. 2018.

[5] S. Mumtaz, A. Alsohaily, Z. Pang, A. Rayes, K. F. Tsang, and J. Rodriguez, "Massive Internet of Things for industrial applications: addressing wireless IIoT connectivity challenges and ecosystem fragmentation," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 28–33, 2017.

[6] S. Sulaiman, A. Aldeehani, M. Alhajji, and F. A. Aziz, "Development of integrated supply chain system in manufacturing industry," *Journal of Computational Methods in Science and Engineering*, vol. 21, no. 3, pp. 599–611, 2021.

[7] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: a survey," 2019, http://arxiv.org/abs/1901.03407.

[8] K. Al Jallad, M. Aljnidi, and M. S. Desouki, "Anomaly detection optimization using big data and deep learning to reduce false-positive," *Journal of Big Data*, vol. 7, no. 1, p. 68, 2020.

[9] A. Cuzzocrea, "Big data lakes: models, frameworks, and techniques," in *Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1–4, eju Island, Korea (South), January. 2021.

[10] L. Xing, K. Demertzis, and J. Yang, "Identifying data streams anomalies by evolving spiking restricted Boltzmann machines," *Neural Computing & Applications*, vol. 32, no. 11, pp. 6699–6713, 2020.

[11] M. Boubekeur, "Industrial applications for cyber-physical systems," in *Proceedings of the 2017 First International Conference on Embedded Distributed Systems (EDiS)*, p. 59, Oran, Algeria, December. 2017.

[12] M. Zolanvari, M. A. Teixeira, L. Gupta, K. M. Khan, and R. Jain, "Machine learning-based network vulnerability analysis of industrial Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6822–6834, 2019.

[13] X. Yan, Y. Xu, X. Xing, B. Cui, Z. Guo, and T. Guo, "Trustworthy network anomaly detection based on an adaptive learning rate and Momentum in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6182–6192, 2020.

[14] Y. Liu, N. Kumar, Z. Xiong, W. Y. B. Lim, J. Kang, and D. Niyato, "Communication-efficient federated learning for anomaly detection in industrial Internet of Things," in *Proceedings of the GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pp. 1–6, Taipei, Taiwan, December. 2020.

[15] X. Wang, S. Garg, H. Lin et al., "Towards accurate anomaly detection in industrial internet-of-things using hierarchical federated learning," *IEEE Internet of Things Journal*, no. 1, p. 1, 2021.

[16] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, "Cyber threats to industrial IoT: a survey on attacks and countermeasures," *IoT*, vol. 2, no. 1, pp. 163–186, 2021.

[17] D. Wu, Z. Jiang, X. Xie, X. Wei, W. Yu, and R. Li, "LSTM learning with bayesian and Gaussian processing for anomaly detection in industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5244–5253, 2020.

[18] I. M. del Águila and J. del Sagrado, "Bayesian networks for enhancement of requirements engineering: a literature review," *Requirements Engineering*, vol. 21, no. 4, pp. 461–480, 2016.

[19] S. u. R. Baig, W. Iqbal, J. L. Berral, and D. Carrera, "Adaptive sliding windows for improved estimation of data center resource utilization," *Future Generation Computer Systems*, vol. 104, pp. 212–224, 2020.

[20] R. van de Schoot, S. Depaoli, R. King et al., "Bayesian statistics and modelling," *Nature Reviews Methods Primers*, vol. 1, no. 1, 2021.

[21] Y. Guo, "Stock price prediction based on LSTM neural network: the effectiveness of news sentiment analysis," in *Proceedings of the 2020 2nd International Conference on Economic Management and Model Engineering (ICEMME)*, pp. 1018–1024, Chongqing, China, August. 2020.

[22] C. I. Orozco, M. E. Buemi, and J. J. Berlles, "Towards an attention mechanism LSTM framework for human action recognition in videos," in *Proceedings of the 2020 IEEE Congreso Bienal de Argentina (ARGENCON)*, pp. 1–6, Resistencia, Argentina, September. 2020.

[23] K. T. Chitty-Venkata and A. Somani, "Impact of structural faults on neural network performance," in *Proceedings of the 2019 IEEE 30th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, p. 35, July. 2019.

[24] T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*, Wiley, Hoboken, New Jersey, U.S, 2003.

[25] P. Bevington and D. K. Robinson, *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill Education, midtown Manhattan, 2003.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4482217, 8 pages
https://doi.org/10.1155/2022/4482217

Research Article

Combination of Blockchain and AI for Music Intellectual Property Protection

Na Li

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Na Li; lina198404@126.com

Received 28 March 2022; Accepted 12 April 2022; Published 28 April 2022

Academic Editor: Konstantinos Demertzis

In the last two years, due to the pandemic and restrictive measures, the dependence of music creators and artists on the Internet, where they could promote their work, organize live streaming concerts, and talk to the public, has increased and expanded even more and seeks higher revenue from digital music platforms. An important issue that arises from the above statement is protecting the authors' copyright regarding the uses and sharing in the digital services of their works with protected content. Although circulated in digital information, the protected content is not information but a product of ethical and commercial value. While it has an intangible owner and it owes its existence to the creative idea of its creator, it is not an idea. The imposition of legal and commercial conditions on its movement cannot be associated with any restrictions on the free movement of information, as it is not related to them. In general, the unauthorized exchange of digital music files via peer-to-peer violates copyright law. The exchange of files is unauthorized, as it does not have the relevant permission from the creators and beneficiaries and is therefore illegal. With this in mind, this paper proposes a highly effective way of protecting the copyright of music technology, which is based on the widespread use of artificial intelligence, blockchain, and cryptography technologies. Specifically, an advanced blockchain model based on Hyperledger Fabric is introduced, which, however, uses Quantum Homomorphic Encryption and Quantum Zero-Knowledge Arguments. Music files are implemented as Nonfungible Tokens (NFTs), which activate smart contracts. Finally, an advanced collaborative filtering algorithm provides recommendations for effectiveness in securing the copyrights of music industry creators. A specialized scenario was built to model the proposed system to verify the degree of protection on music intellectual property in developing a security simulation with an innovative consensus-based zero knowledge and the quantum fully homomorphic encryption technique. Experiment results show that this technique can aid in implementing a technologically aware system capable of providing a powerful answer to a current real-world problem.

1. Introduction

Developments in the digital realm are rapid and have a strong imprint on the livelihood of songwriters and musicians. Through digital music services where their music is played, artists are now looking for income, especially in the wake of the pandemic and the restrictive social distancing measures that have been imposed. It was an alternative artistic outlet for the music industry. It turned to the Internet and music and movie/series streaming services as the dominant entertainment solution in combination with the public. These platforms have made huge profits, while on the contrary, the income of musicians from live concerts and, in general, the income from digital uses is still limited. In addition, piracy, and the sharing of music archives in general, without respect for copyright, is a massive escape from profits. Under copyright and related rights law, copying, translating, performing, or presenting a piece of music to the public, or posting a piece of music on the Internet, is illegal unless there are specific exceptions to the law or permission from all beneficiary parties [1, 2]. In this context, various organizations have moved dynamically in all directions, with business agreements, beneficial collaborations, political means and institutional demands, communication campaigns, and actions to promote the interests of the creators and to highlight their constant need for respect for the copyrights of musicians by all without any exception [1, 3].

The unauthorized peer-to-peer exchange of digital music files violates copyright law. File exchange is unauthorized since it does not have the necessary consent from the creators and beneficiaries and hence is prohibited. The best solution for protecting the musical rights of creators and beneficiaries in the rapidly evolving landscape of the global digital world that affects the field of copyright imposes advanced and technologically up-to-date solutions [4]. New specialized tools are required using advanced algorithms, which will help to identify, in addition to the original declared uses, the musical works in the digital services and the Internet. This will contribute firstly to protecting their copyrights and secondly to the further increase of musicians' income from their works' online and multidisciplinary uses [5, 6]. Multiterritorial online licensing involves licensing music works on the Internet and, to giant users, the so-called digital or online music service providers covering more than one national and geographical territory with various legal approaches [7].

Wanting to overcome the severe weaknesses that characterize traditional mechanisms, this work proposes a highly effective way of protecting the copyright of music technology, which is based on the widespread use of artificial intelligence, blockchain, and cryptography technologies [8]. This template provides a complete cycle of services that start by lending the represented musical works to the online services, locating, and identifying them in the usage reports, collecting the corresponding rights, up to the detailed liquidation of rights that will be in the hands of the creators and beneficiaries of the temporary or total assignment of their copyright. The above stages of this integrated service are encoded in Nonfungible Tokens (NFTs) [9], activated by smart contracts in the Hyperledger fabric [10, 11]. They use advanced quantum encryption methods in the consent mechanisms and the blockchain protection itself. Specifically, an enhanced blockchain paradigm based on Hyperledger Fabric is presented, but with the addition of Quantum Homomorphic Encryption and Quantum Zero-Knowledge Arguments. NFTs are used to implement music files and to activate smart contracts. Finally, a specialized collaborative filtering algorithm makes intelligent recommendations for efficiency and transparency in the use of music content, the detection of violations of preagreed copyright protection rules of music works, the possibility of correcting errors by modifying smart contracts, and the distribution based on actual use [12]. This is a sophisticated system that is proposed for the first time in the literature.

The research is structured as follows. Section 2 provides a complete overview of the topic's associated review. The proposed prototype is shown in Section 3. Section 4 contains a concept of proof in a relevant scenario, and the final section draws conclusions and suggests future study directions.

2. Related Literature

With the development of the Internet, online content has been tremendously increased. In recent years, the research community has focused on the practical use of blockchain technology [13–15].

Li et al. [16] introduced a blockchain-based decentralized music copyright operation management system that disconnects the link between activities and property. They created a distributed licensing management system utilizing the blockchain's shared ledger system and smart contracts as the Ethereum platform's main structure [17]. The platform also brings together the concerns of artists, producers, owners, and customers, which benefits the audio sector's future growth and provides users with more meaningful content and a greater outcome. According to them, the platform is still in the development and upgrading stages.

Bakhytzhan et al. [8] looked at innovative ways to legal control audio media in particular and difficulties and answers. They emphasized a common flaw in the law: it does not follow up with technological advancements and possibilities to manage these relationships. They claimed that the deployment of blockchain technology confronts significant challenges and that many legal concerns must be overcome for blockchain to operate and grow as a property control instrument. The preservation of legal items in digital format is one of the possible applications of blockchain [4]. However, there are still specific issues in the digital music business relating to licensing and distribution, royalties paid by the patent owner, and illicit copying. They concluded that blockchain and smart contracts might help with author rights management in two ways: first, by automating the whole process of creating a licensing agreement, and second, as interim measures of the parties' duties, such as royalties' payment.

Ito and O'Dair [18] outlined the critical obstacles to managing artistic ownership in the digital age before looking at how advocates of the ledger and public blockchain technologies have proposed that these technologies help solve these problems. They also looked at the most critical technical terminology. They concluded that blockchain is a cutting-edge technology that has the potential to revolutionize intellectual property management. A tight emphasis on the technology itself, on the other hand, might lead to a lack of awareness of significant operational and implementation issues. The right design of incentives, at both the operational and implementation levels, will be critical to the efficient use of blockchain software for intellectual asset administration.

Gurkaynak et al. [19] concentrated on the potential benefits of blockchain for the development of intellectual property law and its implications for the registration, maintenance, and protection of creative intellectual assets. They extensively explained the idea of blockchain, examined the hurdles and impediments, detailed the legal position of blockchain and its possible use in intellectual asset law, and outlined the issues. They finished by making recommendations to help progress blockchain technology and raise the number of people aware of it, as well as its effective integration into different services and registration/transaction scenarios.

Amelin et al. [4] looked at employing blockchain technology to keep track of intellectual property items. One of blockchain's main drawbacks is the absence of full-fledged systems for settling creative ownership issues. This issue

necessitates not just technological solutions but also institutional and legal ones. According to the authors, separating such registers into two categories has implications for the creation of blockchain systems for registering and maintaining intellectual property. The first is the permissionless blockchain platform's registers of intellectual property items. The information about entities and their users will not be authorized. The second kind is IP object registries based on license blockchain technology, in which government agencies with IP protection authority will be given superuser powers. They will have formal entries with validating rights value. They claim that the benefits of adopting blockchain-based registers outweigh the disadvantages of centralized registries and that they will be in high demand for securing intellectual property objects in general.

From the above literature, we conclude that even though the research community has conducted a significant study, the practical solutions provided for the effective implementation of blockchain are still limited [7, 8].

3. Proposed Prototype

Hyperledger is a connected P2P network system for developing decentralized ledgers based on blockchain technology. The proposed implementation is based on the architecture of Hyperledger Fabric (HyFa) [10, 20] to establish a system that will contribute to collaboration between various companies, organizations, and stakeholders without the use of public blockchain applications. The commonly accepted HyFa platform combines individual independent sections, which can be selected and combined to provide a solution to protect the music rights of authors and beneficiaries [14, 21, 22].

Consensus in the proposed framework is divided into the following:

(1) In Endorsement, guided by a policy pursued by the participants to approve a transaction

(2) In Ordering, which accepts the approved transactions and agrees on whether the proposal can be registered in the ledger

(3) In Validation, which receives a block of orders and validates the correctness of the result, including the control of the validation policy and the double charge

Each transaction or update of the ledger follows the flow of Figure 1 [20].

The client uses the interface programming environment to form a transaction proposal, which includes the channel name, the chained code name, and the input parameters of the code to be executed. The customer then sends a transaction proposal to all prospective escorts to satisfy the validation policy of that chain code [5, 23]. The peers simulate the transaction based on the parameters received by the customer, interacting with the chain code to record the updates and produce results in the form of a read and write set following the set signature and returning the results to the customer. The client collects the answers from all the peers and confirms that the results are consistent, e.g., all candidates who have subscribed to the same payload. The following is the merging of all the peers' signatures together with the registration readings and the creation of a transaction that is submitted to the ordering service [4, 7, 14].

The proposed ordering service innovates using a consensus algorithm based not on the standard Byzantine Fault Tolerance mechanism but on a zero-knowledge quantum protocol for the QMA class. To achieve zero knowledge, we use the nonblack-box extraction technique that allows the simulator to "imitate" the honest prover without knowing the witness. The protocol consists of a fixed number of rounds (>4) and achieves computational zero-knowledge. We try to attain statistical zero-knowledge while at the same time reducing the number of rounds to four. The proposed implementation requires a fully homomorphic quantum system, which allows homomorphic calculations in quantum circuits and messages [24, 25]. We also use compute-and-compare obfuscation. A compute-and-compare program CC $[f, s, z]$, where f is a function and the s, z strings, has as output the value z for each input x where $f(x) = s$ while rejecting any other entrance. A compute-and-compare obfuscator converts the CC program to the obfuscate CCg program, where it is computationally indistinguishable from a simulated "dummy" program that rejects all inputs. Finally, we use a Conditional Disclosure of Secrets (CDS) protocol. A CDS protocol consists of two rounds and has a sentence z and a message m from the sender as input. The recipient receives m only if the sentence is true; otherwise, the message remains hidden. At the same time, the witness w of the recipient's motion z remains hidden from the sender [26, 27].

Then, we use two rounds of Witness-Indistinguishable Arguments (W-IA), which is the basis of the following results. The protocol consists of a commitment α, a challenge β, and a response γ. The valuable property for us is the calculation of β and γ with an additional message (from the verifier to the prover) to achieve statistical zero-knowledge. The main idea of the protocol is to use a (leveled) fully homomorphic cryptosystem with maliciously circuit private security to reduce the rounds as the verifier sends an encrypted challenge β and the prover first calculates the binding α and then the answer γ is homomorphically encrypted. Knowing the private key of the homomorphic cryptosystem, the verifier can decrypt the ciphertext it receives and confirm they are the right ones (a, b, c). All the above can be constructed considering the quasi-polynomial difficulty of the LWE problem [28, 29].

Cryptography plays a significant role in the proposed model, providing tools that enable secure communication between two or more individuals and securing copyright protection. In particular, the proposed system bypasses the widely known cryptographic protocols, such as public-key encryption, using a quantum fully homomorphic rate-1 encryption scheme, which allows the calculation of functions with encrypted data input for its secure communication via public channels [30, 31].

A fully homomorphic encryption system allows one party to send its encrypted message m under a public key so

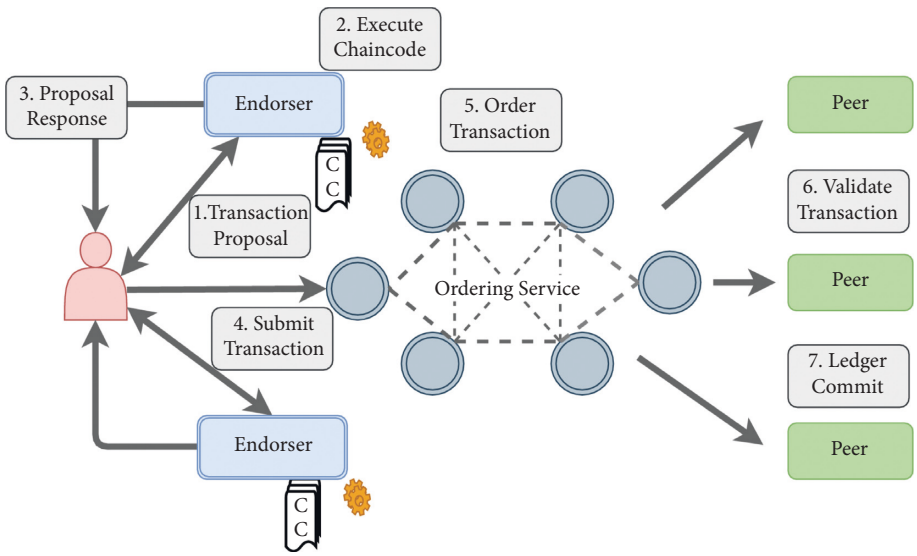


Figure 1: Prototype-high level transaction flow.

that the other can then, having a C circuit, calculate and send [32]:

$$\mathrm{Enc}(m) \longrightarrow^{\mathrm{Eval}(C,\cdot)} \mathrm{Enc}(C(m)), \tag{1}$$

without learning any new information about the message m. Computations on encrypted data apply when a computer-weak client wants to upload his data to a more powerful server that can run complex circuits while maintaining his privacy. The proposed template goes a step further and uses a quantum fully homomorphic rate-1 encryption scheme while ensuring that the communication complexity introduced by the protocol does not negate the enhanced performance offered by the server.

Before delving into our proposal, it is essential to understand why existing protocols fail and show a more significant rate. The rate is generally defined as the fraction of the magnitude of the result of the calculations in non-encrypted information concerning the magnitude of the impact of the measures in encrypted information. In quantum protocols, a cryptocurrency that encrypts a ℓ-qubit quantum state $|\psi\rangle$ is in the following form [30, 32, 33]:

$$\mathrm{QOTP}((x_1, z_1, \ldots, x_\ell, z_\ell), |\psi\rangle), \mathrm{QEnc}(\mathrm{pk}, (x_1, z_1, \ldots, x_\ell, z_\ell)), \tag{2}$$

where QOTP (Quantum One-Time Pad) is applied separately to each qubit and the string otk = $(x_1, z_1, \ldots, x_\ell, z_\ell)$ is encrypted bit by bit. It is easy to notice that this cryptosystem has an inverse polynomial rhythm due to its classic homomorphic rate. An obvious solution would be to adopt a hybrid approach and sample the QOTP key using a pseudorandom generator. More specifically, we could improve the pace by calculating [34, 35]

$$QOTP(PRG(seed), \vee\psi\rangle), QEnc(pk, seed), \tag{3}$$

for some uniformly sampled seed $\leftarrow\$ \{0, 1\}^\lambda$. Then, we can still calculate a function with encrypted input since there is the possibility of converting the ciphertext to its original form, homomorphically calculating the pseudorandom generator. Although this approach works for ciphertext that has just been encrypted, the same is not valid after a homomorphic application of functions; depending on the gateway to be applied to the quantum input, the QOTP otk key changes to a different otk′ string. Although there is a way to refresh the classic part of the ciphertext with each calculation, it does not fit with the hybrid approach we are considering. This is because the modified otk′ probably does not belong to the arrival set of the pseudorandom generator PRG; i.e., there may not be a seed′ string such that PRG (seed′) = otk′. Therefore, we observe that we cannot eliminate the classic QEnc (pk, otk) encryption. Even in the ideal case where the classical homomorphic shape has an optimal rate since at least two classical bits are required to encrypt a qubit, we will still have more complexity than desired and reach a dead end. So, the proposed quantum fully homomorphic rate-1 encryption scheme is ideal in terms of both security and performance [36–38].

The use of copyright will be based on Nonfungible Tokens (NFTs), which will be activated using smart contracts. Each NFT is a digital asset stored on a secure but transparent global blockchain. It is also nonexchangeable, which means unique. Thus, an NFT is a digital component associated with a distinct and unique component. NFTs contain security, transparency, and unchanged encryption storage, are indivisible, and can store significant amounts of data, including individual information. This is what makes a particular distinctive nonexchangeable and stored in a Smart Contract, which is code that executes automatically when a set of conditions is displayed [39, 40].

Combining a Smart Contract with other unique identifying metadata—such as the owner's identity and secure file links—along with the security provided by the blockchain provides virtually unquestionable proof of ownership and authenticity to potential buyers. Smart Contracts can prevent someone from transferring an NFT or accessing an underlying asset unless all the conditions set out in the

contract are met, including the possible payment of royalties for the resale of the NFT. The NFTs in this standard are also coded to enforce Smart Contract copyright clauses. When reselling an NFT, automatically pay a fee to the seller, usually a fixed percentage of the resale price [41, 42].

In conclusion, NFTs and smart contracts act as certificates of authenticity to the underlying asset and as a valuable representation of the ownership of a tangible asset as a stock. The NFT holder has access to the underlying asset but may not have exclusive access to or control of the asset, let alone the asset owner or any intellectual property. The default rule is that the patent holder or copyright holder retains all intellectual property rights unless it is clear from the terms of the market (e.g., the Smart Contract in our case) that ownership of a copyright is being transferred [24, 42].

Finally, the template uses specialized collaborative recommendation mechanisms to make intelligent recommendations for efficiency and transparency in the use of music content, detect breaches of preagreed copyright rules, and correct errors by modifying smart contracts and based on actual usage. Specifically, a memory-based collaborative filtering methodology is used, which calculates the usefulness of each object for a user by directly processing all the evaluations contained in the system. For example, if I_{au} is the set of objects shared by users a and u, then their degree of similarity can be deduced from the Minkowski distance [19, 43, 44]:

$$w_{a,u} \equiv d_{a,u} = \left[\sum_{i \in I_{au}} \left| r_{a,i} - r_{u,i} \right|^k \right]^{1/k}, \tag{4}$$

where k is the class of the distance. For various values of k, the general formula of the equation takes specific forms. The cosine factor is also used:

$$w_{a,u} = \frac{\overrightarrow{I_a \cdot \overrightarrow{I_u}}}{\left\| \overrightarrow{I_a} \right\|_2 \cdot \| \|_u} = \frac{\sum_{i \in I_{au}} r_{a,i} r_{u,i}}{\sqrt{\sum_{i \in I_{au}} r_{a,i}^2} \sqrt{\sum_{i \in I_{au}} r_{u,i}^2}}, \tag{5}$$

and the Pearson correlation coefficient:

$$w_{a,u} = \frac{\sigma(I_a, I_w)}{\sigma_{I_a} \times \sigma_{I_w}} = \frac{\sum_{i \in I_{au}} \left(r_{a,i} - r_a \right) \left(r_{u,i} - r_u \right)}{\sqrt{\sum_{i \in I_{au}} \left(r_{a,i} - \overline{r_a} \right)^2 \sum_{i \in I_{au}} \left(r_{u,i} - \overline{r_u} \right)^2}}, \tag{6}$$

where $\sigma(I_a, I_w)$ is the covariance of the user's scores a and u and σ_{Ia}, σ_{Ib}, the corresponding standard deviations.

In the final stage, the recommendation type estimates a particular object's benefit for a given user. Its general form is shown in the following equation:

$$\widehat{r_{a,i}} = \underset{u \in U_{a,i}}{\text{aggr}}\, r_{u,i}, \tag{7}$$

where $\widehat{r_{a,i}}$ is the value of the benefit of the object i for the user a or the prediction of the evaluation that the user a would make for the object i, if he had known in the past. $U_{a,i}$ is a set of "similar" to a users (or objects) who have already evaluated object i and which has been created by the filtering process. The aggr notation describes how peer-to-peer ratings are processed. The relative formulas used are the simple average:

$$\widehat{r_{a,i}} = \frac{1}{\left| U_{a,i} \right|} \sum_{u \in U_{a,i}} r_{u,i}, \tag{8}$$

the weighted average:

$$\widehat{r_{a,i}} = \frac{\sum_{u \in U_{a,i}} w_{a,u} r_{u,i}}{\sum_{u \in U_{a,i}} w_{a,u}}, \tag{9}$$

and the Resnick type:

$$\widehat{r_{a,i}} = \overline{r_a} + \frac{\sum_{u \in U_{a,i}} w_{a,u} \left(r_{u,i} - \overline{r_u} \right)}{\sum_{u \in U_{a,i}} \left| w_{a,u} \right|}. \tag{10}$$

The main advantages of the process are related to the ease of their algorithmic implementation and the possibility of immediate update of the forecasts as soon as new (or existing) ratings are added to the system. Also, the generated forecasts are constantly improving with the increase of the available ratings in the system.

4. Proof of Concept

For the modeling of the proposed system, a specialized scenario was implemented to verify the degree of protection on music intellectual property in implementing a security simulation with an innovative consensus-based zero knowledge and the quantum fully homomorphic encryption scheme [12, 27, 45].

Specifically, the validity of the scenario is a direct consequence of the security of the homomorphic cryptosystem, to prove which of the following changes are required:

(1) The prover calculates a random commitment used in the homomorphic calculation. Thus, the verifier can (in proof of correctness) confirm the validity of the protocol without knowing the private key of the homomorphic cryptosystem. While maintaining WI statistics, a particular commitment scheme is used, the Sometimes Binding Statistically Hiding (SBSH) commitment. There is a (negligibly small) chance of having a perfect commitment in such a commitment scheme. In this case, the verifier can export the blocked message. The leveraging technique proves the correctness.

(2) All the above procedure is repeated twice, and the prover proves that in at least one of the two cases, the calculations were done correctly, using a WI statistic (for the NP class). This is enough to prove the nondiscrimination of the witness of the total protocol since in the proof we can "exchange" the witness in each step separately.

Since this protocol is the basis for the following, those, in turn, will be based on the quasi-polynomial difficulty of the problem.

In the simulation technique, for simplicity, we first consider verifiers that do not interrupt communication and are explainable; i.e., honest verifier algorithms support the messages. The essence of the protocol is the exportable commitment scheme which works as follows [19, 46]:

1 The sender samples two random strings s, td, and

(a) A public and a private key (pk, sk) of a homomorphic QFHE cryptosystem and the string encryption td, $c_{\text{td}} = \text{QFHE.Enc}(\text{pk,td})$.

(b) The obfuscated program $\widetilde{\mathbf{CC}} \leftarrow \text{Obf}\mathbf{CC}[f, s, (\text{sk}, m)]$, with f being the QFHE decryption function; the sender sends pk, $c_{\text{td},}$ and $\widetilde{\mathbf{CC}}$ to the recipient.

(2) The recipient guesses a value of y and sends it encoded via the CDS protocol.

(3) The sender responds with a message encrypted via the CDS protocol so that if $y = t$, it returns the value s. Alternatively returns $\perp$.

Intuitively, the above procedure offers commitment since the message in the obfuscated program is uniquely defined and secret since the receiver cannot correctly guess the td value, despite a negligible probability. The simulator can also output (sk, m) and simulate the sender's optics. After receiving the first message, it homomorphically calculates the sender's last message using its circuit, entering the encrypted td value and the sender's internal state. The result of the homomorphic calculation is the message encrypted via CDS, whose proposition is correct and returns the value of s, encrypted via QFHE. This value is exactly the input required for $\widetilde{\mathbf{CC}}$ to return the message m. In addition, the CC program returns the private sk key together so that the simulator can decrypt the QFHE encrypted messages and generate a valid copy of the T communication without using rewinding.

Respectively, the zero knowledge in 4 rounds is proved. We can upgrade the WI protocol to zero knowledge using the above binding technique. In the first round, the verifier sends a zero-value commitment with random r (same as the randomness used to generate the keys of the QFHE cryptosystem). Then, we implement the above extraction technique by setting as m the randomness r. At the end of their interaction, the prover uses the WI protocol to prove that it knows the randomness r or $x \in L$ [36, 47].

To defend from malicious attacks where the prover falsifies a valid CDS control witness from the td encryption, we add an SBSH y value binding, which can be extracted with low probability, allowing the attack to be reduced QFHE security. At the same time, we check through the CDS protocol that the binding pattern is correctly defined (i.e., the prover includes the randomness of the SBSH binding as part of the witness). For the solution, we use the 3-round postquantum CDS protocol with statistical security on the part of the recipient.

The only issue is that we thought that the verifiers do not interrupt communication and are explainable. For the first problem, we consider two simulators, one for the case that interrupts and one that does not interrupt the interaction. Then, we build a combination simulator that randomly chooses which of the two to use. Watrous' rewinding allows the simulator to rewind until it guesses correctly, without affecting the verifier. On the other hand, to confirm that the verifier is explicable, we add to our protocol a proof of zero knowledge (from the verifier to the prover) that ensures that its messages are honest since the verifier is classical in our protocol, as long as the proof of zero-knowledge is for the NP class [31, 48].

To ensure statistically zero knowledge, however, we must have statistical correctness in the new ZK protocol, and therefore, we need proof of delayed-entry zero knowledge (with statistical correctness). At the same time, the receipt must not exceed three rounds so as not to add extra around the total communication. However, we do not know of any 3-round zero-knowledge protocol (postquantum). However, in our case, a less powerful tool is enough. We can sometimes use simulated (sometimes simulatable) zero knowledge (SSim ZK), where the simulation is possible with a negligible low probability. To use the above tool, we need to configure the security parameters of the other protocols appropriately to compensate for this exponential loss, similar to the SBSH binding pattern. SSim resembles zero knowledge with super polynomial simulation (SPS), with the main difference that in SPS zero knowledge, the simulator runs in hyperpolynomial time, in contrast to SSim zero knowledge, where the simulator runs in polynomial time. Still, there is an exponentially slight chance of success. This difference is of significant importance for our protocol since, basically, we cannot rewind the state of the verifier, and therefore, we require the simulation to be linear.

5. Conclusions

This work introduced an innovative blockchain model based on Hyperledger Fabric, which uses Quantum Homomorphic Encryption and Quantum Zero-Knowledge Arguments. The music files are implemented as NFTs, which activate smart contracts. At the same time, through intelligent artificial intelligence algorithms, recommendations are made for the effectiveness in securing the copyrights of the creators of the music industry. As it turned out experimentally, this process can assist in implementing a technologically aware system that can provide a powerful solution to a real modern problem. This is a sophisticated system that is proposed for the first time in the literature.

And while the proposed system presents exceptionally high levels of security, its implementation requires processing all available data before making recommendations, which is not immediately feasible, especially in substantial digital music repositories. Another problem is that they cannot generate recommendations for users who do not have standard ratings with other users and, respectively, cannot suggest items that have not yet been rated by someone (cold start). Finally, they are vulnerable to the

problem of overspecialization because they cannot generalize the data they process.

The above weaknesses are also immediate issues for future investigation, which will promote the system's reliability to a much higher degree. The aim is to protect the interests of music producers and the general protection of the music industry's copyright.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] M. Orzan, G. Orzan, G. Spiridon, and T. Neacsu, "Informatic online system for dissemination of scientific research results for intelectual property rights," in *Proceedings of the 2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, pp. 1–6, Cluj-Napoca, Romania, May 2010.

[2] E. J. S. Gonzalez and K. McMullen, "The design of an algorithmic modal music platform for eliciting and detecting emotion," in *Proceedings of the 2020 8th International Winter Conference on Brain-Computer Interface (BCI)*, pp. 1–3, Gangwon, Korea (South), April. 2020.

[3] Z.-Y. Zhao, C.-D. Wang, P.-J. Zheng, Q. Gong, K.-W. Huang, and J.-H. Lai, "Music sharing platform based on sina app engine," in *Proceedings of the 2015 Ninth International Conference on Frontier of Computer Science and Technology*, pp. 298–303, Dalian, China, August 2015.

[4] R. Amelin, V. Arkhipov, S. Channov, M. Dobrobaba, and V. Naumov, "Prospects of blockchain-based information systems for the protection of intellectual property," in *Communications in Computer and Information Science*, D. A. Alexandrov, A. V. Boukhanovsky, A. V. Chugunov, Y. Kabanov, O. Koltsova, and I. Musabirov, Eds., vol. 1038, pp. 327–337, Springer International Publishing, Cham, 2019.

[5] Y. Kim, K.-H. Kim, and J.-H. Kim, "Power trading blockchain using hyperledger fabric," in *Proceedings of the 2020 International Conference on Information Networking (ICOIN)*, pp. 821–824, Barcelona, Spain, January 2020.

[6] F. Leal, B. Veloso, B. Malheiro, J. C. Burguillo, A. E. Chis, and H. González-Vélez, "Stream-based explainable recommendations via blockchain profiling," *Integrated Computer-Aided Engineering*, vol. 29, no. 1, pp. 105–121, 2021.

[7] Y. Yue and X. Fu, "Research on medical equipment supply chain management method based on blockchain technology," in *Proceedings of the 2020 International Conference on Service Science (ICSS)*, pp. 143–148, Xining, China, August 2020.

[8] B. Bakhytzhan, A. Magauiya, M. Tuktibayeva, and T. Gaukhar, "The use of blockchain technology in the field of digital music," in *Proceedings of the 2021 16th International Conference on Electronics Computer and Computation (ICECCO)*, pp. 1–3, Kaskelen, Kazakhstan, November. 2021.

[9] U. W. Chohan, "Non-fungible Tokens: blockchains, scarcity, and value," *SSRN Electronic Journal*, 2021.

[10] E. Androulaki, A. Barger, V. Bortnikov et al., "Hyperledger fabric," in *Proceedings of the 30th EuroSys Conference*, pp. 1–15, Porto Portugal, April. 2018.

[11] Q. Nasir, I. A. Qasse, M. Abu Talib, and A. B. Nassif, "Performance analysis of hyperledger fabric platforms," *Security and Communication Networks*, vol. 2018, pp. 1–14, 2018.

[12] H. Takahashi and U. Lakhani, "Voting blockchain for high security NFT," in *Proceedings of the 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE)*, pp. 358–361, Kyoto, Japan, October. 2021.

[13] X. Yu, Z. Shu, Q. Li, and J. Huang, "BC-BLPM: a multi-level security access control model based on blockchain technology," *China Communications*, vol. 18, no. 2, pp. 110–135, Oct. 2021.

[14] A. Hassan, M. I. Ali, R. Ahammed, M. M. Khan, N. Alsufyani, and A. Alsufyani, "Secured insurance framework using blockchain and smart contract," *Scientific Programming*, vol. 2021, pp. 1–11, 2021.

[15] S. Yousuf and D. Svetinovic, "Blockchain technology in supply chain management: preliminary study," in *Proceedings of the 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, pp. 537-538, Granada, Spain, August 2019.

[16] Y. Li, J. Wei, J. Yuan, Q. Xu, and C. He, "A decentralized music copyright operation management system based on blockchain technology," *Procedia Computer Science*, vol. 187, pp. 458–463, 2021.

[17] S. Casale-Brunet, P. Ribeca, P. Doyle, and M. Mattavelli, "Networks of Ethereum non-fungible Tokens: a graph-based analysis of the ERC-721 ecosystem," in *Proceedings of the 2021 IEEE International Conference on Blockchain (Blockchain)*, pp. 188–195, Melbourne, Australia, December 2021.

[18] K. Ito and M. O'Dair, "A critical examination of the application of blockchain technology to intellectual property management," in *Business Transformation through Blockchain*, H. Treiblmaier and R. Beck, Eds., Springer International Publishing, Cham, pp. 317–335, 2019.

[19] G. Gürkaynak, İ. Yılmaz, B. Yeşilaltay, and B. Bengi, "Intellectual property law and practice in the blockchain realm," *Computer Law & Security Report*, vol. 34, no. 4, pp. 847–862, Aug. 2018.

[20] Hyperledger – Open Source Blockchain Technologies, 2022, https://www.hyperledger.org/.

[21] B. Ampel, M. Patton, and H. Chen, "Performance modeling of hyperledger sawtooth blockchain," in *Proceedings of the 2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 59–61, Shenzhen, China, July. 2019.

[22] N. Živić, "Distributed ledger technology for automotive production 4.0," in *Proceedings of the 2020 28th Telecommunications Forum (TELFOR)*, pp. 1–3, Belgrade, Serbia, November 2020.

[23] Y. Chang, C. Fang, and W. Sun, "A blockchain-based federated learning method for smart healthcare," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–12, Article ID 4376418, 2021.

[24] V. Aleksieva, H. Valchanov, and A. Huliyan, "Implementation of smart-contract, based on hyperledger fabric blockchain," in *Proceedings of the 2020 21st International Symposium on Electrical Apparatus Technologies (SIELA)*, pp. 1–4, Bourgas, Bulgaria, June. 2020.

[25] T. Alshalali, K. M'Bale, and D. Josyula, "Security and privacy of electronic health records sharing using hyperledger fabric," in *Proceedings of the 2018 International Conference on*

Computational Science and Computational Intelligence (CSCI), pp. 760–763, Las Vegas, NV, USA, December 2018.

[26] M. C. Xenya and K. Quist-Aphetsi, "Decentralized distributed blockchain ledger for financial transaction backup data," in *Proceedings of the 2019 International Conference on Cyber Security and Internet of Things (ICSIoT)*, pp. 34–36, Accra, Ghana, May 2019.

[27] W.-S. Park, D.-Y. Hwang, and K.-H. Kim, "A TOTP-based two factor Authentication scheme for hyperledger fabric blockchain," in *Proceedings of the 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 817–819, Prague, Czech Republic, July. 2018.

[28] T. Zhang, "Association rules," *Knowledge Discovery and Data Mining. Current Issues and New Applications*, vol. 23, pp. 245–256, Berlin, Heidelberg, 2000.

[29] C. Velasco, R. Colomo-Palacios, and R. Cano, "Neural distributed ledger," *IEEE Software*, vol. 37, no. 5, pp. 43–48, 2020.

[30] S. Behera and J. R. Prathuri, "Application of homomorphic encryption in machine learning," in *Proceedings of the 2020 2nd PhD Colloquium on Ethically Driven Innovation and Technology for Society (PhD EDITS)*, pp. 1-2, Bangalore, India, November 2020.

[31] M. Mohan, M. K. K. Devi, and V. J. Prakash, "Homomorphic encryption-state of the art," in *Proceedings of the 2017 International Conference on Intelligent Computing and Control (I2C2)*, pp. 1–6, Coimbatore, India, June. 2017.

[32] J. Kim and A. Yun, "Secure fully homomorphic authenticated encryption," *IEEE Access*, vol. 9, pp. 107279–107297, 2021.

[33] H. Ahmed and J. Glasgow, "Swarm intelligence : concepts , models and applications technical report 2012-585," 2012, https://www.semanticscholar.org/paper/Swarm-Intelligence-%3A-Concepts-%2C-Models-and-Report-Ahmed-Glasgow/116b67cf2ad2c948533e6890a9fccc5543dded89.

[34] N. Abdelgaber and C. Nikolopoulos, "Overview on quantum computing and its applications in artificial intelligence," in *Proceedings of the 2020 IEEE Third International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, pp. 198-199, Laguna Hills, CA, USA, December. 2020.

[35] D. Ferrari, A. S. Cacciapuoti, M. Amoretti, and M. Caleffi, "Compiler design for distributed quantum computing," *IEEE Transactions on Quantum Engineering*, vol. 2, pp. 1–20, 2021.

[36] H. Zhang, Z. Ji, H. Wang, and W. Wu, "Survey on quantum information security," *China Communications*, vol. 16, no. 10, pp. 1–36, 2019.

[37] J. Singh and M. Singh, "Evolution in quantum computing," in *Proceedings of the 2016 International Conference System Modeling Advancement in Research Trends (SMART)*, pp. 267–270, Moradabad, India, November. 2016.

[38] J. Vuckovic, T. Yoshie, M. Loncar, H. Mabuchi, and A. Scherer, "Nano-scale optical and quantum optical devices based on photonic crystals," in *Proceedings of the 2nd IEEE Conference on Nanotechnology*, pp. 319–321, Washington, DC, USA, August. 2002.

[39] X. Zhao and Y.-W. Si, "NFTCert: NFT-based certificates with online payment gateway," in *Proceedings of the 2021 IEEE International Conference on Blockchain (Blockchain)*, pp. 538–543, Melbourne, Australia, December 2021.

[40] K. B. Muthe, K. Sharma, and K. E. N. Sri, "A blockchain based decentralized computing and NFT infrastructure for game networks," in *Proceedings of the 2020 Second International Conference on Blockchain Computing and Applications (BCCA)*, pp. 73–77, Antalya, Turkey, November 2020.

[41] E. Erturk, M. Dogan, U. Kadiroglu, and E. Karaarslan, "NFT based fundraising system for preserving cultural heritage: heirloom," in *Proceedings of the 2021 6th International Conference on Computer Science and Engineering (UBMK)*, pp. 699–702, Ankara, Turkey, September. 2021.

[42] C. Jiang and C. Ru, "Application of blockchain technology in supply chain finance," in *Proceedings of the 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*, pp. 1342–1345, Harbin, China, September. 2020.

[43] P. Tang, W. Wang, J. Lou, and L. Xiong, "Generating adversarial examples with distance constrained adversarial imitation networks," *IEEE Transactions on Dependable and Secure Computing*, no. –1, 2021.

[44] H. Worthington, R. S. McCrea, R. King, and R. A. Griffiths, "Estimation of population size when capture probability depends on individual states," *Journal of Agricultural, Biological, and Environmental Statistics*, vol. 24, no. 1, pp. 154–172, 2019.

[45] M. Ahmed, S. Reno, N. Akter, and F. Haque, "Securing medical forensic system using hyperledger based private blockchain," in *Proceedings of the 2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6, DHAKA, Bangladesh, December 2020.

[46] Y. Hui and L. Zesong, "Research on real-time analysis and hybrid encryption of big data," in *Proceedings of the 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pp. 52–55, Chengdu, China, May 2019.

[47] P. L. Knight, "Quantum communication and quantum computing," in *Proceedings of the Technical Digest. Summaries of papers presented at the Conference on Lasers and Electro-Optics. Postconference Edition. CLEO '99. Conference on Lasers and Electro-Optics (IEEE Cat. No.99CH37013)*, p. 56, Baltimore, MD, USA, May 1999.

[48] T. Shen, F. Wang, K. Chen, K. Wang, and B. Li, "Efficient leveled (multi) identity-based fully homomorphic encryption schemes," *IEEE Access*, vol. 7, pp. 79299–79310, 2019.

Hindawi
Computational Intelligence and Neuroscience
Volume 2023, Article ID 9840420, 1 page
https://doi.org/10.1155/2023/9840420

Retraction

Retracted: Cyber Risk Recommendation System for Digital Education Management Platforms

Computational Intelligence and Neuroscience

Received 26 September 2023; Accepted 26 September 2023; Published 27 September 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] X. Yin and Y. Chen, "Cyber Risk Recommendation System for Digital Education Management Platforms," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 8548534, 11 pages, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 8548534, 11 pages
https://doi.org/10.1155/2022/8548534

Research Article

Cyber Risk Recommendation System for Digital Education Management Platforms

Xiufang Yin and Yanfang Chen

Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Xiufang Yin; yinxiufang19850107@163.com

Received 26 March 2022; Accepted 8 April 2022; Published 28 April 2022

Academic Editor: Konstantinos Demertzis

Covid-19 pandemic has ushered in a new school and academic year for students in a distance learning regime. This new daily routine was unprecedented and undoubtedly unusual, especially for the younger ones. At this point and at these ages, the risk of cyber fraud is even greater. The transition from the physical environment to the Internet took place quickly without the appropriate time to control potential risks and the proper information and training of teachers and students. Some common threats that need to be addressed to protect learners and their data when using e-learning methods are malicious remote access, malware, phishing, cyber fraud, etc. Considering the above situation, this work presents an innovative cyber risk recommendation system for digital education management platforms. The system in question is a distributed two-stage algorithm based on game theory and machine learning, which is trained by the constant change in the choice of recommendations by users to maximize security. We examine the algorithm's ability to simulate a user system in which everyone independently selects a user recommendation, assesses the environment and the implications of this choice, and then concludes whether it will continue to have that recommendation fixed. The methodology with which we have represented the digital e-learning system has been done with an approach that directly corresponds with their general view as a cyber-physical-social system. We consider the digital school as an environment that brings limitations, leading us to a pretty demanding personalization problem. Users coexist in this environment, in which everyone acts voluntarily but influences and is influenced by the surrounding environment. Our results lead us to conclude that this algorithm responds in a fully effective, flexible, and efficient way to the needs of protection and risk assessment of e-learning education systems.

1. Introduction

As the pandemic continues and educational institutions follow a hybrid education model (both physical and remote), the academic realm continues to attract the attention of digital criminals [1, 2]. The number of users who encountered threats disguised as popular online training/video conferencing platforms increased for all digital platforms. About 98% of the threats belonged to the not-a-virus category, divided into riskware and adware [3]. Adware bombards users with unwanted ads, while riskware consists of various files—from browser toolbars and download managers to remote management tools—that can perform multiple actions on the remote computer without the user's consent. Although much less than 1%, other types of more dangerous digital threats such as Trojans and Ransomware have also been reported [3]. Generally, users meet threats from unofficial websites that resemble the original platforms or spam emails and phishing campaigns presented as special offers or notifications by the platform or even by the e-learning institutions themselves [4, 5].

Under these conditions, the educational organizations that are directly affected try to organize, coordinate, and finally manage their organization's risk [6, 7]. Risk Management is defined as how organizations approach the risks associated with their activities and objectives to ensure timely and smooth development. The general idea is that the effort to achieve the goals of an organization is now intertwined with the effort to "repel" the various risks that threaten to hinder this endeavor, wherein, in this case, they

concern the targeting of their educational mechanisms. Lack of confidence is eliminated, but it is constantly increasing due to the complexity of the digital world. Risk management allows organizations to prepare for this uncertainty, minimizing the risk they face before dealing with it. The ability to capture and control risks results in organizations making their decisions with greater certainty and confidence [8].

Digital risk analysis is a complicated and dynamic subject that requires a high level of skill in quantitative approaches, techniques, and instruments. It is an issue when risk management is implemented using out-of-date methods, especially in today's digital world. Cyber risk recommendation for digital education management platforms is proposed in this research, considering the above. We are dealing with a distributed two-stage method based on game theory and machine learning that continuously adapts to new proposals from users to maximize the safety of each one [6]. Replicating a user system in which each person makes their own choice, examines the consequences, and then makes their own choice again is what we are interested in. As a cyber-physical-social system, we consistently described the digital e-learning system with their overall vision [9]. We see the digital classroom as a constraining setting that forces us to confront a problematic personalization dilemma. Users cohabit in this ecosystem, where everyone acts on their own will but is influenced by their surroundings. Our findings concluded that our algorithm meets the protection and risk assessment demands of e-learning education systems as completely effective, adaptable, and efficient [10].

2. Related Literature

The danger of financial loss, interruption, or harm to an organization's reputation because of a breakdown of its information technology systems is known as cyber risk [8, 11]. A deliberate and unauthorized security breach to acquire access to information systems for espionage, extortion, or humiliation might be an example of such a danger. Also, unintended or unintentional security breaches may or may not represent an expose that has to be handled [12].

Amin [6] offered a pathway for evaluating cyber risk by creating a framework for a Bayesian network to simulate financial loss as a function of significant risk and resilience variables. Instead of taking a compliance approach to risk management, he took a valuation strategy. He employed a qualitative scorecard evaluation to analyze the extent of cyber risk exposure and the success of the organization's resilience strategies. He emphasized the uniqueness of risks and questioned the usual usage of actuarial models, which are typically used to analyze financial risk. There is a vast body of experience data.

The goal of Lassoued et al. [4] was to uncover the barriers to attaining quality in remote learning during the Coronavirus pandemic and to look into the many methods in which students continued their studies at home throughout the pandemic. They employed an exploratory, descriptive technique using a questionnaire given to both professors' and students' responses. According to the findings, teachers and students encountered self-imposed and educational, technological, budgetary, and organizational barriers. Universities blended traditional and new techniques of distance learning, as well as radio and television classes. After providing electronic publications via university websites, other institutions utilized the Internet to give lessons through different educational platforms or social networking sites to explain the lessons. When faced with similar or unanticipated obstacles in the future, the following recommendations were made to help in the study area and other areas to improve their ability to deliver high-quality distance learning.

Hakak et al. [2] looked at the harmful cyber activities linked to COVID-19 and various mitigating strategies. They also recommended an attack taxonomy to aid with risk management and mitigation in the future. They looked at COVID-19-themed hacks and divided them into four groups: service disruption, financial gain, stealing of information, and a piece of malware that took advantage of the pandemic fear, with subcategories inside each (e.g., malware, ransomware, phishing). They presented proposed mitigation methods using these categories. The cyberattack taxonomy and possible mitigation measures may also help design future pandemic cyberattack prevention efforts. They plan to expand on the suggested taxonomy and develop a risk management strategy for significant future crises.

Paris et al. [13] focused their research on the privacy problems that arose because of the pandemic's spread in higher education. They observed higher institutions' deployment of popular online learning platforms using critical informatics methodologies and theories to find numerous trends that emerged as a consequence of their implementation. They concluded their study by recommending measures to effectively manage the risks posed by higher education's usage of digital platforms, particularly those connected to privacy.

Thanou et al. [9] used quality of experience (QoE) to solve the challenge of integrating and analyzing the influence of visitor behavioral elements. User happiness has been frequently expressed using QoE, measured using correctly specified utility functions. The visitor quality of experience, as measured by a prospect-theoretic utility function, is greatly influenced by the total amount of time consumed in displays by all visitors, making their actions and choices interconnected and functioning more like a socially competitive environment. A game among visitors was designed and solved in a distributed method to identify the optimum time stay in exhibitions. The evaluation findings were provided in detail, emphasizing the functioning and superiority of the suggested framework while also giving obliging insights into visitor choices and behaviors under actual settings.

Based on the literature reviewed above, we can conclude that there is an open field for effective research in managing cyber risk in digital education management education platforms during pandemic crises, particularly in the manner presented in this study.

3. Proposed Approach

Given the uncertainty of documenting such a complex system, the proposed algorithmic approach focuses on modeling recommendations for users of educational applications, QoE, digital risk assessment, and the implementation of recommendations [14] in the user's system for their further protection [15]. More specifically, we deliberate an educational platform, where the best way to manage is to study, capture, and estimate the digital risk based on the time of each user's visit. We assume that users can choose between R different recommendations within the proposed platform.

For the present study, we will consider three different recommendations $R = \{R_A, R_B, R_C\}$, while, in general, a digital program for managing educational activities may have more. Each of these recommendations offers users a different QoE and, at the same time, involves an extra degree of user engagement, which we will analyze in detail. We also denote the set of users as $N = \{1, \ldots, i, \ldots, N\}$ of which each has its type of visit, namely ant, butterfly, fish, and grasshopper. The sets of users per type are denoted as N_a, N_b, N_f, N_g, and therefore, the group of users is $N = N_a \cup N_b \cup N_f \cup N_g$. Each of the classes has different behavior in terms of visit time, and how tolerant–careful–informed is the user in digital security about other users of the platform [16].

The time of each user's visit to the platform is symbolized as t_i, and due to the characteristics of the platform (e.g., use of HTTPS) but also the personal characteristics and preferences–knowledge of each user, it is the upper and lower block, that is, $t_i^{\min} < t_i < t_i^{\max}$. We also denote as t_{-i} the times of all other users coexisting simultaneously on the platform with user i, $i \in N$. Each different composition R_x, $x = A, B, C$, and $R_x \in R$ offers a different QoE to the user who chooses it. We call these QoE Q_x, $x = A, B$, and C provided by the R_A, R_B, and R_C recommendations. Without compromising the generality, we assume that the values of these QoEs are sorted in a truly ascending order, that is, $Q_A < Q_B < Q_C$, which indicates that the higher the recommendation identifier (A, B, C), the better the recommendation. A simple and effective metric that allows QoE verification for each user is the relative visit time ratio, which we define as [17, 18]

$$rt_i = \frac{t_i}{\sum_{j \neq i} j \in N t_j}, \tag{1}$$

where the denominator is the total time that other users devote to their visit simultaneously as i. So we can easily observe that when the sum of the times of other users increases, then either the risk becomes higher, or the same number of users spend total time on the visit, which gives us an indication of the risk of misuse of the platform. In such a case, the user's i QoE deteriorates, either because there is a vast number of users on the platform, which means, for example, many shared files, augmented shared hyperlinks, or because some security issues are sidelined, resulting in increased inherent risk, for example, many interconnected external devices, access from unsafe media [19, 20].

We also assume that recommendations with increased QoE will lead users to the necessary time of the visit and the most basic security standards required since their stay will be more guided, and they will be offered a large amount of information in a short time, and in a more structured way. It is, therefore, natural for many users to ask for the most attractive recommendation, which will lead to a long wait for this recommendation due to the increased demand. Thus, the recommendations can be characterized by an additional congestion control parameter c_x, $x = A, B, C$, representing the possible standby time for each proposal. This way, users should consider having a better recommendation, which may increase their QoE, but depending on how many have made the same choice, it can lead to a cost of waiting. The balance between the two should direct users to compromise and, therefore, to choose another recommendation. For our study, we undertake three different offers, which are the following [21–23]:

(1) Recommendation A (R_A): The platform has uploaded cyber security instructions. Users can follow these instructions in a short time without waiting for recommendations. However, they must comply with the platform's security rules without guidance and compliance checks. Thus, their perceived QoE (Q_A) is limited as their security level for their browsing will be limited and possibly with significant security vulnerabilities.

(2) Recommendation B (R_B): A virtual guide is to assist users on the platform and provide explanatory and helpful information on critical security issues to look out for. Users form groups as they enter the platform, and the guide makes group recommendations at specific intervals. Therefore, users should suppose more time before starting their education platform to coordinate the team and carry out a group implementation process of the recommendations. However, their perceived QoE (Q_B) will increase because they will have more structured information on critical security issues that concern them relatively quickly.

(3) Recommendation C (R_C): In addition to the features of the previous R_B recommendation, in R_C, the guide provides users with information to implement–controlled by the user group. Depending on the number of teams that will be created and the different security levels that will have to be met, the completion times of the process will vary. Therefore, users will have to wait in even longer queues. However, the perceived QoE increases as the users digest the information even more.

According to the above description of the recommendations, we observe that the arrangement of QoE for each one is genuinely increasing, that is, $Q_A < Q_B < Q_C$. In addition, more recommendations can be considered than the three we have proposed, following the same pattern, without compromising the generality. In addition, for the sake of

equality and to support the concept of accessibility for more users on the platforms under consideration, we assume that all the above recommendations have the exact computational cost for users [20, 24]. Finally, the negatives for each recommendation can be summarized quantitatively in the computational cost of the congestion control parameter c_x, $x = A, B, C$ that we entered so that we will have $c_A < c_B < c_C$.

The concept of a QoE function was adopted to represent the perceived satisfaction of each user as a function: the time devoted to the implementation of appropriate security measures, the recommendation chosen, and the satisfaction of their QoE prerequisites. A combined QoE function is adopted by each user and consists of the pure QoE component and the congestion control function. Pure QoE is expressed as the ratio of the QoE value achieved to the time spent using the platform. Thus, pure QoE increases if the user has achieved excellent QoE without devoting much time to using the platform. Note that the visit time is preferably for each user and is minimum and maximum blocked $t_i^{\min} < t_i < t_i^{\max}$, as already mentioned. We also note that the optimal visit time will not always be $t_i^{\min}$. Still, because QoE depends on the final recommendation choice and congestion on the platform (expressed by the total visit time of all users), QoE may be limited [25, 26].

Thus, the platform user expresses his flexibility regarding visit time (through the $t_i^{\min} < t_i < t_i^{\max}$limits) and considers his perceived QoE to determine the optimal visit time. More specifically, the ideal QoE function is signified as an accounting function with a variable relative to the rt_i visit time, that is,

$$\begin{aligned} &Q_x \cdot f_i^y(rt_i), \\ &x = A, B, C, \\ &y = a, b, f, g, \end{aligned} \tag{2}$$

where y denotes the four user categories a(nt), b(utterfly), f(ish), g(rasshopper). The function $f_i^y(rt_i)$, which we will refer to as the visiting efficiency function (VEF), represents user satisfaction and depends on the congestion on the platform at that time as well as how much time the user spends. Congestion is expressed solely by the time other users spend on the platform ($\sum_{\substack{j \in N \\ j \neq i}} t_j$). For the form of VEF, we chose the following sigmoid function [23, 24]:

$$f_i^y(rt_i) = \left(1 - e^{-A_y rt_i}\right)^{M_y}, \tag{3}$$

where A_y, M_y, $y = a, b, f, g$ are positive parameters that control the function's slope. Each of the four user types is characterized by a central relative visit time rt_i, target, y, $y = a, b, f, g$ which differs from species to species and is the turning point of the logistic function. The placement of the functions in the above order respects the characteristics of the visit types. We suppose that

$$rt_i^{\text{target},b} < rt_i^{\text{target},a} < rt_i^{\text{target},f} < rt_i^{\text{target},g}. \tag{4}$$

The *butterfly*-type user can redirect his route to the platform in case of congestion as opposed to the type of *ant* that needs to check all possible checkpoints, and therefore,

$$rt_i^{\text{target},b} < rt_i^{\text{target},a}. \tag{5}$$

The *fish* type user is waiting for the control procedures of the other group users to complete, so the satisfaction is limited. In contrast, *ant* user patiently completes all control procedures without complaint. So, we have

$$rt_i^{\text{target},a} < rt_i^{\text{target},f}. \tag{6}$$

The *grasshopper* user has specific and limited security issues to resolve during his visit to the platform, so if he is late, his QoE decreases dramatically.

In general, if a user has less QoE than $rt_i^{\text{target},y}$, then perceived satisfaction decreases rapidly. If, on the other hand, a user has a higher QoE than this, then the perceived QoE increases more slowly because the satisfaction conditions are met. Gathering the above observations, the combined QoE function can be expressed as [20, 22, 27]

$$\begin{aligned} &Q_i(t_i, t_{-i}) = \frac{Q_x \cdot f_i^y(rt_i)}{t_i} - c_x N_x^k e^{t_i}, \quad i \in N,\ x = A, B, C, \\ &y = a, b, f, g,\ N = \mathcal{N}_a \cup \mathcal{N}_b \cup \mathcal{N}_f \cup \mathcal{N}_g. \end{aligned} \tag{7}$$

N_x^k symbolizes how many users have chosen recommendation x at the time of calculation and is the population congestion factor that expresses a spatial weight concept in the weight of each recommendation. It is a way that gives us more control to adjust the cost of each recommendation according to the number of people our algorithm pretends, without having to change the cost values c_x which are inherent in the recommendations and do not make sense to change depending on how large the total number of users examined.

The penetration of a composition R_x, $x = A, B, C$ is expressed as the ratio of the total combined QoE of the users. They have chosen the composition R_x to the total achieved combined QoE of all users present on the platform for the specific observation moment τ. We define the penetration of a recommendation as [28, 29]

$$p_x(\tau) = \frac{\sum_{R_x}^{N} Q_x \cdot f_i^y(rt_i)}{\sum_{\substack{i=1 \\ \forall R_x \in R}}^{N} Q_x \cdot f_i^y(rt_i)}. \tag{8}$$

The above process allows the systematic approach of controlling and assessing the risks associated with the activities of users participating in e-learning platforms. The focus of efficient risk management is to identify and manage these risks to ensure a robust environment that can be causally assessed. The categorization and control of users in groups increase the probability of success of the educational organization's overall security and defense objectives. It should be noted that the proposed procedure can be visualized as a three-dimensional network of random walks.

A random walk is a random process in mathematics that depicts a path made up of an unpredictably large number of unexpected steps on a particular mathematical space random walk on a regular lattice is a prominent random walk model in which a probability distribution determines the position of each stage. A probability distribution determines the location of each step. If you take a simple random walk, the place can only jump to neighboring lattice sites, which results in a lattice path being formed. In an introductory symmetric random walk on a locally finite lattice, the probability of a location jumping to each of its immediate neighbors is the same as the likelihood of the place jumping to the location's close neighbors. A three-dimensional network of random walks is shown in Figure 1 [30, 31].

The properties of transience and reproducibility (true and not) in this three-dimensional network of random walks are class properties. If a state x has one of these properties, then all states of class x belong to have the same property. It is reasonable that short sequences are open (and vice versa; every open class is passing). In contrast, iterative classes are closed (the inverse is not valid–generally finite closed classes are iterative), which means that a Markov chain is created that allows them to appear in transient states with a probability of 1 and after a finite number of steps, to return to a repetitive state. Hence, they are essentially forever in the repetitive state class [32, 33].

Based on the above formulation, and assuming that the Markov chain is nondegradable, genuinely repetitive, and nonperiodic, then regardless of its initial distribution, we have [30, 32, 34]

$$P\left[\frac{V_n(i)}{n} \longrightarrow \pi(i), \forall i \in S\right] = 1. \tag{9}$$

So, if the transition from one situation to another situation entails a fee or some cost (negative fee) of the form,

$$R_n = R_n\left(X_{n-1}, X_n\right), \quad n \in \mathbb{N}. \tag{10}$$

for the n-th step, then we have a total reward in the first n steps equal to

$$C(n) = \sum_{s=1}^{n} R_s\left(X_{s-1}, X_s\right). \tag{11}$$

While for the average reward in one step it applies

$$\lim_{n \longrightarrow \infty} \frac{C(n)}{n} = \sum_j r_j \cdot \pi_j. \tag{12}$$

In conclusion, we can say that the risk assessment can reveal from the early stages of implementing a distance learning system the severe security gaps that the organization should address and the possible ways to avoid them. The developed risk management procedures described are ongoing throughout the educational project, with digital risk management applying to all individual projects, from the smallest (implemented by one person) to vast and complex. Many issues can be addressed in advance and allow the project manager to determine a specific course using the proposed cyber risk recommendation system for digital education management platforms [7].

4. Modeling

To model the proposed system, we implemented a specialized scenario to verify the location of the user's actions when using an e-learning platform. Before starting their browsing, users entering the platform select a recommendation to guide their browsing based on a machine learning framework. Users act as cellular automata who gain information and experience from their previous actions as the time of the algorithm progresses. At the end of the algorithm, they will have chosen the final recommendation, which will be the one that will guide them during their browsing on the platform.

These automata can listen to their environment while keeping a history of their past decisions to make beneficial decisions in the future that will lead them to optimize their QoE. The necessary information needed to decide on the recommendation they will make is the time of the visit and the values of the combined QoE of the previous iteration τ of the machine learning part of the algorithm. In addition to the above parameters, each user has advised the input of each recommendation R_x, $x = A, B, C$ of all users, that is, $p_x(\tau)$, to make the final decision and action $a(\tau)$. Given the actions of the users regarding the selection of the recommendation, the users take part in a distributed noncooperative game for the determination of the visit time, played in each repetition of the machine learning algorithm, to regulate the optimal visit time as well as the corresponding value of their QoE [33, 35, 36].

Therefore, the algorithm consists of these two repetitive decision parts, one for selecting the recommendation and one for choosing the visit time, and is done before the users visit the platform. In addition, we emphasize that the algorithm is executed every time a new user enters the platform, considering the history of all previous users who are already on the platform. The algorithm's execution time is relatively short, which is helpful since it is executed while users are waiting for the application to start.

The platform environment consists of R_x, $x = A, B, C$, R_x in R different recommendations, and N users can be studied as a learning system, where users act as learning machines and respond to their environment to decide which recommendation to choose. Each user/automaton is learning $i, i \in \mathcal{N}, \mathcal{N} = \mathcal{N}_a \cup \mathcal{N}_b \cup \mathcal{N}_f \cup \mathcal{N}_g$ for each iteration τ of the machine learning system and has a set of actions a_A, a_B, a_C. This set of actions represents the different choices that users can make regarding the recommendation. To adopt what action to take for each iteration τ, users consult the output $\beta(\tau) = Q(\tau), t(\tau)$ of their environment, where $Q(\tau)$ and $t(\tau)$ are vectors that contain the combined QoE and the visit time of each user for the snapshot τ. The output $\beta(\tau) = Q(\tau), t(\tau)$ is determined after the execution of the time management part.

Combining the selected actions of the users and the reaction of the system in the form of QoE, we calculate the probability of rewarding. This probability is the penetration

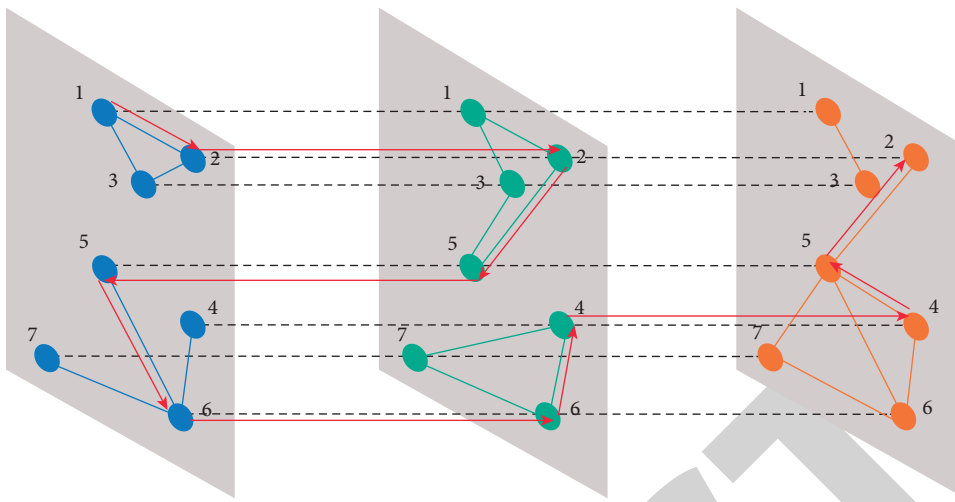


Figure 1: Random walks on multilayer networks.

of the R_x composition and is expressed by the relation [11, 37, 38]

$$p_x(\tau) = \frac{\sum_{i=1}^{N} Q_x \cdot f_i^y(rt_i)}{\sum_{\substack{i=1 \\ R_x \in R}}^{N} Q_x \cdot f_i^y(rt_i)}. \tag{13}$$

The probability of each user's action is expressed as a vector which in turn describes the probability that user i selects the R_x, $x = A, B, C$ recommendation, and following the automata learning model, this probability is updated in each round as follows:

$$\begin{aligned} \mathrm{Pr}_{i,x}(\tau+1) &= \mathrm{Pr}_{i,x}(\tau) - b \cdot p_x(\tau) \cdot \mathrm{Pr}_{i,x}(\tau), \quad x^{(\tau+1)} \neq x^{(\tau)}, \\ \mathrm{Pr}_{i,x}(\tau+1) &= \mathrm{Pr}_{i,x}(\tau) + b \cdot p_x(\tau) \cdot \left(1 - \mathrm{Pr}_{i,x}(\tau)\right), \quad x^{(\tau+1)} = x^{(\tau)}, \end{aligned} \tag{14}$$

where b, $0 < b < 1$ is a parameter that controls the convergence time of the process. The equation represents the probability that the user must choose in $(\tau+1)$ a composition different from the one in (τ). At the same time, it also describes the probability that the user continues to prefer the same composition, that is, $x(\tau+1) = x(\tau)$. As for the loading of the selection probabilities of each recommendation, in the absence of any prior knowledge of the personal preference of each user, the algorithm initially considers all possibilities equally. Finally, we point out that each user converges on the recommendation that can offset the cost of waiting in line and the limitations in the visit times that everyone has.

Given the recommendation option, each user i, $i \in N$, aims to select the optimal visit time to maximize their QoE. Therefore, the above problem can be expressed as a distributed QoE maximization problem in terms of visit time as

$$\begin{aligned} &\max_{t_i} Q_i(t_i, t_{-i}), \\ &\tau.\omega.t_i^{\mathrm{Min}} < t_i < t_i^{\mathrm{Max}}, \\ &\forall i \in N, x = A, B, C, \\ &y = a, b, f, g, N = \mathcal{N}_a \cup \mathcal{N}_b \cup \mathcal{N}_f \cup \mathcal{N}_g. \end{aligned} \tag{15}$$

Due to the distributed nature of the optimization problem and the selfish nature of the users in terms of optimizing their QoE, a game theory approach is applied to determine the optimal time for each. We denote as $G = \{N, T_i, Q_i\}$ the noncooperative visit time selection game, where N is the set of players, that is, the users of the platform, $T_i = [t_i^{\mathrm{Min}}, t_i^{\mathrm{Max}}]$ is the strategy space of the i-th user and Q_i the corresponding QoE function of this user. To decide in detail on the solution, we will recall the concept of Nash equilibrium. To ensure the existence and uniqueness of the Nash equilibrium, we must show that the Q_i function of each user is a convex function for t_i. A function Q_i can be strictly convex if for every pair of different $t_i - t_i'$ belonging to the convex set T_i, with $0 < \lambda < 1$,

$$Q_i(t_i') > Q_i(t_i) \Longrightarrow Q_i(\lambda \cdot t_i + (1-\lambda) \cdot t_i') > Q_i(t_i). \tag{16}$$

The combined QoE function of user $i, i \in N$, $N = N_a \cup N_b \cup N_f \cup N_g$ is convex in the strategy interval T_i' corresponding to the interval of the relevant time ratio [19, 33, 36, 39].

$$rt_i \in \left(\frac{\ln M}{A}, rt_B\right), \quad rt_B \in \left(\frac{\ln M_y}{A_y}, \frac{\ln 10^4 M_y}{A_y}\right). \tag{17}$$

Consequently, the Nash equilibrium point of the game $G = \{N, T_i, Q_i\}$ exists and is unique in the corresponding strategy interval.

To prove the curvature of the combined QoE function of users, we will consider the sign of the second derivative for t_i.

$$\frac{\partial^2 Q_i}{\partial t_i^2} = g(rt_i) + h(rt_i) - c_x N_x e^{t_i}, \tag{18}$$

where

$$g(rt_i) = \frac{2Q_x}{t_i^3}\left(1 - e^{A_y rt_i}\right)^{-My}\left[-M_y A_y e^{-A_y rt_i} rt_i + 1 - e^{-A_y rt_i}\right] \times \alpha,$$

$$h(rt_i) = \frac{M_y A_y^2 Q_i}{t_i^3}\left(1 - e^{-A_y rt_i}\right)^{-2M_y} e^{-A_y rt_i} rt_i^2 \left(M_y e^{-A_y rt_i} - 1\right). \tag{19}$$

Since $g(rt_i)$ is continuous, we conclude that

$$g(rt_i) < 0, \quad \forall rt_i \in \left(\frac{\ln M_y}{A_y}, rt_B\right). \tag{20}$$

By default, the Nash equilibrium of the noncooperative game should satisfy the following:

$$t_i^* = \mathrm{BR}_i(t) = \arg\max Q_i(t_i, t_{-i}). \tag{21}$$

We, therefore, conclude that the overall convergence of the noncooperative time management game at the Nash equilibrium point, under the proposed best response function, is guaranteed.

5. Results

To evaluate the algorithm, we will study the importance of the various parameters for the system and their effect on the solutions provided by the algorithm. Precisely, we must first determine the input of the algorithm, which consists of the following:

(1) The strategy field, $T_i = [t_i^{\mathrm{Min}}, t_i^{\mathrm{Max}}]$ represents the minimum and maximum time available to the student $i \in N$

(2) Cost values c_x, $x = A, B, C$ of each recommendation

(3) The values QoE Q_x $x = A, B, C$ of each composition

(4) The total number of visitors N

(5) The visit type of each visitor $y_i = a, b, f, g$.

(6) The force k of the population congestion parameter N_x^k

(7) The four visit efficiency functions f.

By executing the algorithm, we obtain as a result for each visitor the R_i recommendation chosen by the visitor, the QoE value of each visitor, and the suggested visit time of each visitor. Therefore, to evaluate the correctness of the algorithm results, we will observe these three outputs for each scenario. In addition, we will assess the time performance of the algorithm depending on the increase in the number of visitors and the change of the merging parameter b in the machine learning model [40–42].

To evaluate the algorithm's results in assigning a recommendation to each visitor, we first studied a system with ten ($N = 100$) visitors. The following two diagrams show the final QoE price for each visitor. Specifically, in Figure 2, the QoE value is depicted for each visitor and average QoE with the forced assignment of recommendations.

Figure 2 shows the result of the game if we omit the machine learning part and assign to all visitors the seemingly best recommendation *C*. Figure 3 shows the QoE value for each visitor and average QoE from the use of the proposed algorithm, with the same parameters as in the above case.

We have marked a pair (R, y) above each value in both cases. R represents the final composition selected for visitor i while y is the visitor type [a (nt), b (utterfly), f (ish), g (rasshopper)]. All tests were made in the Google Colab environment using GPU: Nvidia K80/T4, GPU Memory 12 GB; GPU memory clock: 1.59 GHz; and performance: 4.1 TFLOPS.

Comparing the two charts above, we observe that the algorithm has avoided assigning the best recommendation to all users and has set the recommendations so that all visitors are satisfied as far as possible. We also observe that the algorithm leads to QoE values that avoid excessive satisfaction and excessive dissatisfaction, bringing visitor satisfaction levels close to the average value. In addition, both in Figure 3 and in all the results we got, the visitors—*ants* and the *butterflies*—are consistent among the most satisfied. This approves that these two types of visitors show the most patience, both in the queues and during their visit.

Finally, the effect of overcrowding can be seen in Figure 2, but if we look closely, it is also evident in Figure 3. More specifically, taking as an example visitors 4 and 5 in Figure 3, we observe that a "better" composition does not imply a better QoE. Visitors 4 and 5 are of the *fish* type, and while four has received recommendation *A* which is more straightforward than *B*, it nevertheless has a better QoE than 5. This is because for recommendation *A* the tail is only three visitors while for *B* it is four. Therefore, because *fish*-type visitors are impatient, 4 has less anticipation and thus more satisfaction.

We then considered a system with one hundred ($N = 100$) visitors evenly distributed with the four types of visits (25 visitors per type). Figure 4 shows the result of the average QoE value and visit time for each of the visit types.

The results are representative of the selection of visit efficiency functions. More specifically, *butterfly* and *ant* visitors have the highest QoE, while the most difficult to satisfy *fish* and *grasshopper* types have the lowest QoE. Visit times are also typical of the distribution given by the functions in question. This is because the breaking points of the functions are progressively more right for each type of visitor. This results in the best QoE value being at a higher time value. The above result reveals the exceptional importance of the functions in the configuration of the problem and the modeling of the types of visitors. In conclusion, visit efficiency functions are the basic configuration of the algorithm that encodes the behavior of the four types of visits.

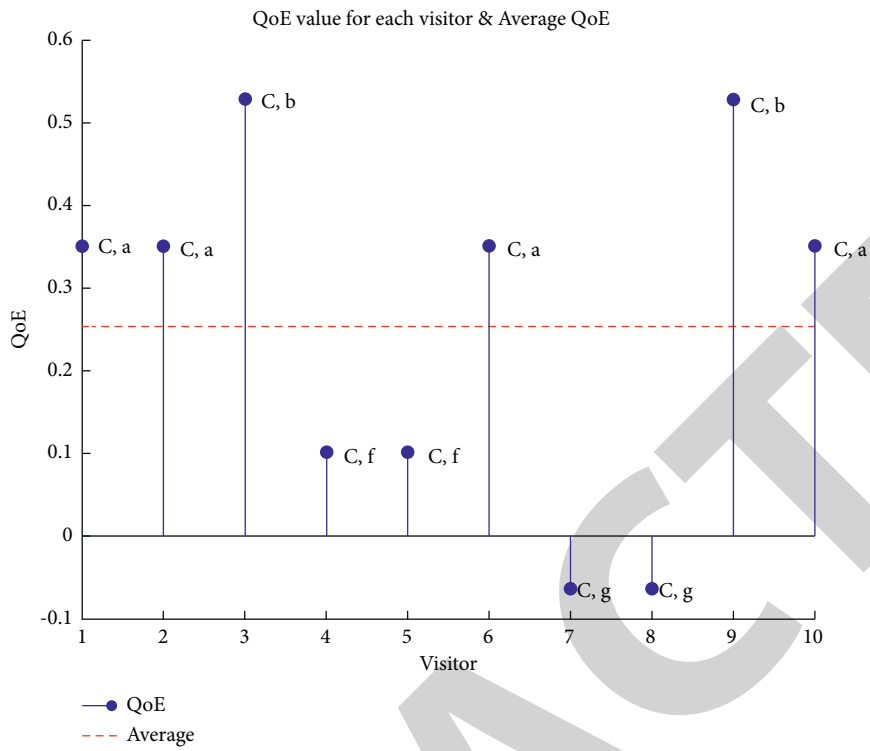


Figure 2: QoE value for each visitor and average QoE (forced assignment of recommendations).

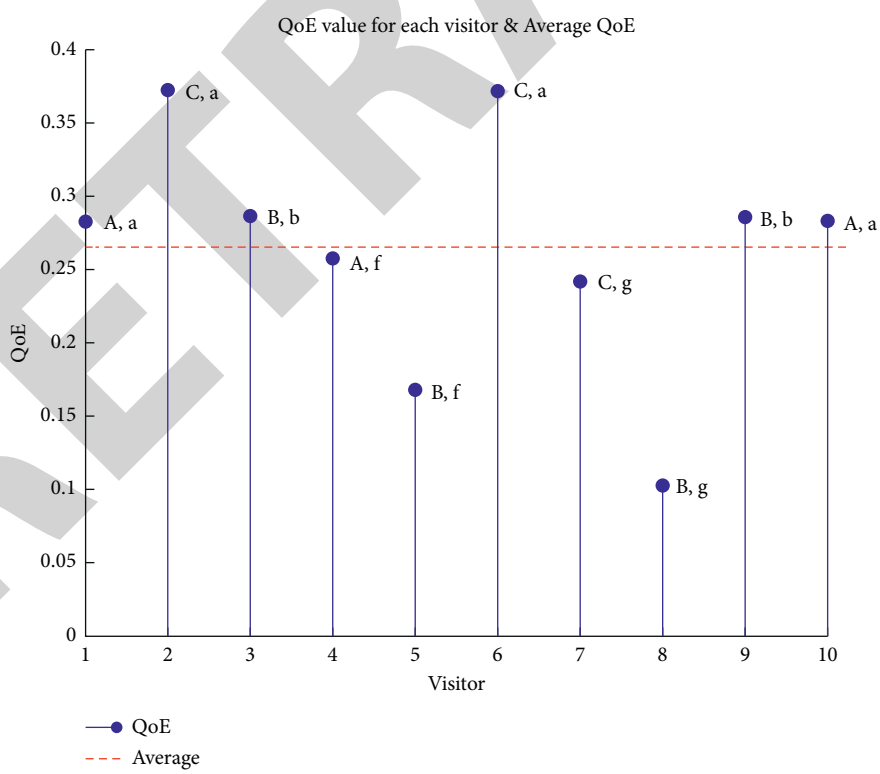


Figure 3: QoE value for each visitor and average QoE (forced by the proposed algorithm).

Figure 5 offerings the total number of visitors who have selected each recommendation as to the timing τ of the machine learning part of the algorithm progresses.

As it turns out, the game in the early stages is more active as the machine learning loop pushes visitors to review their recommendation options frequently. Then, the changes are

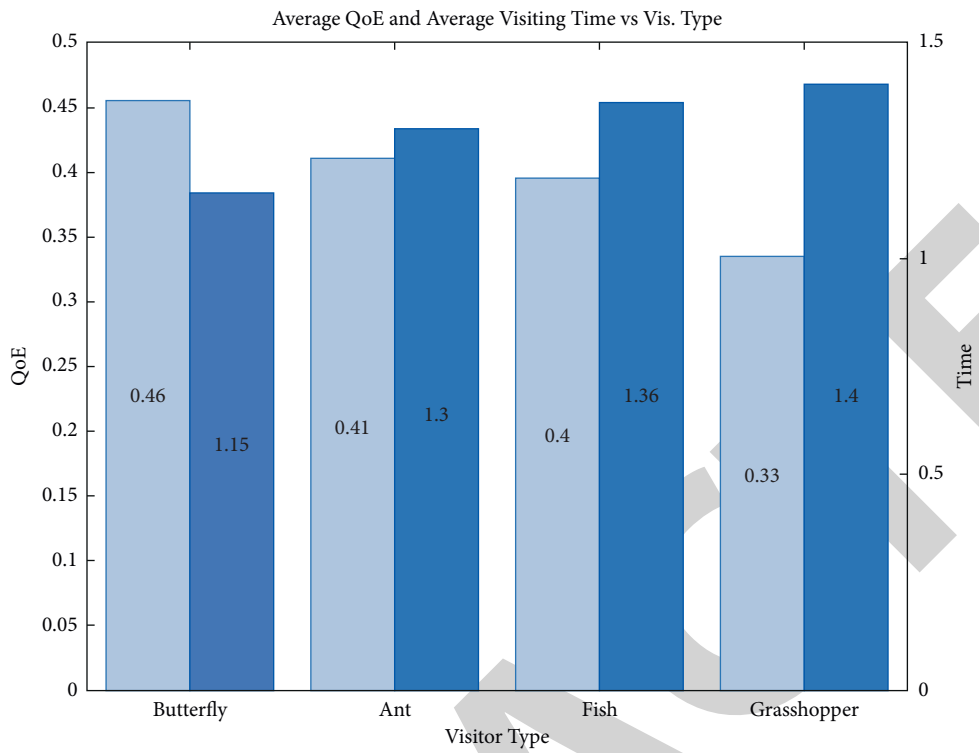


Figure 4: Average QoE and average visiting time vs. visiting time.

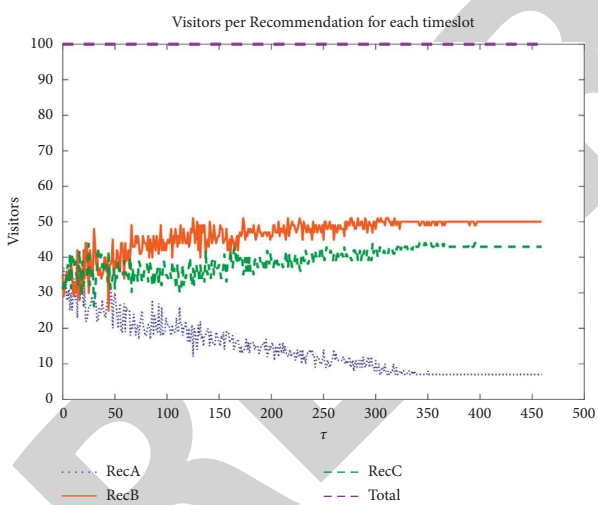


Figure 5: Visitors per recommendation for each timeslot τ.

reduced, and the visitors come to their final recommendation with certainty. We also notice that the number of repetitions is relatively small, implying a short execution time. In addition, we observe that, in the time range between 50 and 70, the algorithm's response to the dominance of a recommendation can be seen where, for that snapshot, it pushes the revision of the recommendations to the next round of recurrence machine learning. It is worth noting that we considered that the visitor system does not receive additional visitors in the scenarios we examined. Therefore, the stabilization we observe is final. However, if the system received new visitors during the algorithm's execution, this stabilization would cease with the entry of unique visitors and would return much faster. This is because, as we have shown, the equilibrium point exists and is secured.

We, therefore, conclude that the presence of the machine learning loop is significant for the operation of the algorithm. Without visitors being seen as automata learning, there would be no feedback loop that allows the algorithm to know the system as a whole. The circle of machine learning is the one that enables the elimination of such nonuniform solutions. In particular, in the event of an internal situation, the value of the penetration factor p_x in the linear law of probability renewal will give recommendation C a minimal value. Consequently, in the next iteration of the problem of finding the optimal visit time, many visitors will have been pushed away from recommendation C. Thus, the algorithm has avoided this undesirable outcome.

6. Conclusions

This paper presented an innovative cyber risk recommendation system for digital education management platforms. The system implements and proposes a distributed two-step algorithm based on game theory and machine learning, which is trained by the continuous change in the choice of recommendations by users to maximize the provision of the desired level of digital security and the corresponding risk these platforms capture. The proposed algorithm is an innovative effort in distance learning education systems. It is a fully automated risk assessment system with which educational institutions can ensure the levels of digital security and the quality of the user experience in a thorough and

adaptable way. The simplicity of implementation, as well as the low computational complexity, is what makes it extremely useful and functional. However, the low computational complexity hides in the background of the need for careful configuration, that is, there is a delicate balance between the fast implementation time and the time it needs to be configured. The variety of usage and configuration options allows the respective educational organizations to organize the right balance for the individual digital educational platform. This algorithm can be applied based on their individual needs and priorities.

Given the plethora of customization, many new paths can be drawn after studying the applications of the proposed system. The cost parameters, the benefit of the recommendations, and the congestion parameter should be directly linked to the platform and the individual quality and risk characteristics that characterize it to optimize the implementation of the algorithm and strengthen the potential of each organization to offer each recommendation. Finally, given the framework of machine learning and automata learning, convergence step b needs additional study in terms of its importance in terms of the behavior of machines (visitors) in conjunction with the implementation time, which we studied in this paper. More specifically, it is worth investigating whether a small refresh parameter attributes a feature to how visitors respond to the system of recommendations. In other words, a big step may mean a quick and imperfect way of updating the visitors' opinion, but, in some cases, depending on the way the cost of the recommendations is organized, it may be more desirable. In this way, additional knowledge can be obtained as to which price to choose and whether it will be in the high or low range.

This research's future directions are primarily concerned with the investigation and extension of the model with inherent capabilities of optimizations processing for the automated system to fully utilize the powers of more comprehensive dependencies of modeling learning systems with increased accuracy and efficiency. The potential examination of the impact of such a grouping methodology on risk assessment and management development, compared to standard ways of separation, is particularly intriguing, as is the possibility of doing such research utilizing nonparametric machine learning methods in the future.

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This study was supported by the Research Project on Curriculum Reform of Teacher Education in Henan Province, for Practical Research on "Integrated Training Mode of "Investigating-Training-Evaluating-Pursuing" Based Demand of Preschool Teacher" (No. 2021-JSJYZD-067).

References

[1] T. Ahmad, "Corona virus (COVID-19) pandemic and work from home: challenges of cybercrimes and cybersecurity," *SSRN Electronic Journal*, 2020.

[2] S. Hakak, W. Z. Khan, M. Imran, K.-K. R. Choo, and M. Shoaib, "Have you been a victim of COVID-19-related cyber incidents? survey, taxonomy, and mitigation strategies," *IEEE Access*, vol. 8, Article ID 124144, 2020.

[3] Comparitech, "Malware statistics in 2022: frequency, impact, cost & more," 2022, https://www.comparitech.com/antivirus/malware-statistics-facts/.

[4] Z. Lassoued, M. Alhendawi, and R. Bashitialshaaer, "An exploratory study of the obstacles for achieving quality in distance learning during the COVID-19 pandemic," *Education Sciences*, vol. 10, no. 9, p. 232, 2020.

[5] B. Chang, "Student privacy issues in online learning environments," *Distance Education*, vol. 42, no. 1, pp. 55–69, 2021.

[6] Z. Amin, "A practical road map for assessing cyber risk," *Journal of Risk Research*, vol. 22, no. 1, pp. 32–43, 2019.

[7] L. Novák, P. Doucek, P. Doucek, and L. Nedomová, "Efficient cyber risk management–auditors experience," in *Organizacija in Negotovosti V Digitalni Dobi/Organization and Uncertainty in the Digital Age*University of Economics, Prague, Czech Republic, 2018.

[8] Y. A. Basallo, V. E. Senti, and N. M. Sanchez, "Artificial intelligence techniques for information security risk assessment," *IEEE Latin America Transactions*, vol. 16, no. 3, pp. 897–901, 2018.

[9] A. Thanou, E. E. Tsiropoulou, and S. Papavassiliou, "Quality of experience under a prospect theoretic perspective: a cultural heritage space use case," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 1, pp. 135–148, 2019.

[10] M. Carroll, S. Rohin, K. H. Mark et al., "On the utility of learning about humans for human-AI coordination," *Advances in Neural Information Processing Systems*, vol. 32, 2019, https://papers.nips.cc/paper/2019/hash/f5b1b89d98b7286673128a5fb112cb9a-Abstract.html.

[11] S. Guopan, "The effect of probability on risk perception and risk preference in decision making," in *Proceedings of the 2010 International Conference on Education and Management Technology*, pp. 690–693, Cairo, Egypt, November 2010.

[12] G. D'Acquisto, J. Domingo-Ferrer, P. Kikiras, V. Torra, Y. A. D. Montjoye, and A. Bourka, "Privacy by design in big data: an overview of privacy enhancing technologies in the era of big data analytics," 2015, https://arxiv.org/abs/1512.06000.

[13] B. Paris, R. Reynolds, and C. McGowan, "Sins of omission: critical informatics perspectives on privacy in e-learning systems in higher education," *Journal of the Association for Information Science and Technology*, vol. 73, no. 5, pp. 708–725, 2022.

[14] I. A. A.-Q. Al-Hadi, N. M. Sharef, N. Sulaiman, and N. Mustapha, "Review of the temporal recommendation system with matrix factorization," *International Journal of Innovative computing*, vol. 13, 2017.

[15] R. Ankele, S. Marksteiner, K. Nahrgang, and H. Vallant, "Requirements and recommendations for IoT/IIoT models to automate security assurance through threat modelling, security analysis and penetration testing," in *Proceedings of the 14th International Conference on Availability*, pp. 1–8, Reliability and Security, New York, NY, USA, August 2019.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 1493493, 9 pages
https://doi.org/10.1155/2022/1493493

Research Article

English Text Recognition Deep Learning Framework to Automatically Identify Fake News

Fei Wu[1] and Xiaoyu Luo[2]

[1]*Hunan Institute of Engineering, 411104 Xiangtan, Hunan, China*
[2]*Hunan University of Technology and Business, 410205 Changsha, Hunan, China*

Correspondence should be addressed to Xiaoyu Luo; luoxiaoyu@hutb.edu.cn

Received 27 March 2022; Revised 6 April 2022; Accepted 8 April 2022; Published 28 April 2022

Academic Editor: Konstantinos Demertzis

Fake news spreading rapidly worldwide is considered one of the most severe problems of modern technology that needs to be addressed immediately. The remarkable increase in the use of social media as a critical source of information combined with the shaking of trust in traditional media, the high speed of digital news dissemination, and the vast amount of information circulating on the Internet have exacerbated the problem of so-called fake news. The present work proves the importance of detecting fake news by taking advantage of the information derived from friendships between users. Specifically, using an innovative deep temporal convolutional network (DTCN) scheme assisted using the tensor factorization non-negative RESCAL method, we take advantage of class-aware rate tables during and not after the factorization process, producing more accurate representations to detect fake news with exceptionally high reliability. In this way, the need to develop automated methods for detecting false information is demonstrated with the primary aim of protecting readers from misinformation.

1. Introduction

Man learns and forms consciousness within the social context. He adopts opinions and constructs a large part of his perceptions based on the data given to him as undisputed knowledge by people he trusts, such as teachers, parents, friends, and information sources. This social transmission of knowledge is at the heart of human civilization. However, often, the information and "knowledge" are wrong, intentionally or not. They are usually written and published to mislead [1], harm an organization, a legal or physical person, or even for economic or political benefits, often using impressive titles, or entirely made to increase readability [2]. Fake news is a yellow press or propaganda done with deliberate misinformation or pranks spread through traditional or social media [3, 4].

Considerable research is being conducted on tactics for combating and suppressing fake news, including disinformation, which is the deliberate distribution of false narratives for political reasons or to undermine social cohesiveness in targeted populations [5]. On the other hand, numerous solutions must be adjusted to individual types of fake news, based, for example, on whether the fake news is actively manufactured or produced inadvertently or unintentionally [6].

The difficulty of detecting fake news on social media is mainly because fake news is deliberately written to deceive readers [3, 7]. The content of the information can vary in terms of subject matter and writing style. For example, an actual event may be presented in a misleading context to support something false [8]. As a result, a machine learning model that detects fake news solely through content may not deliver the best results. Instead, we can take advantage of additional information generated by the network, such as the profile of users who interact with the news on social media, to get more accurate results [9].

The problem of detecting fake news [10] on social media has been studied and categorized into three main approaches: content, network, and content - network (hybrid approach) [11]. Content-only strategies focus on analyzing

the content of the news (e.g., vocabulary, syntax) and detecting patterns through natural language processing methods. However, to produce satisfactory results, a predefined scope is required, which is difficult to achieve in the case of fake news because of its diversity [12]. Network-based approaches extract information from different networks created by users who interact with the news, such as diffusion or relationship networks. In general, network-based methods perform better in dynamic environments and are more suitable for detecting fake news. Finally, hybrid approaches aim to combine the advantages of content and network-based models, integrating both language features and network information into one model [13].

The overall strategy is to discover awful news through human fact-checking and automated artificial intelligence (machine learning, natural language processing, and network analysis). Research communities have used two fundamental counter-strategies: lowering the importance of fake news and sending out warning messages [10, 14].

In the first strategy, objectionable content is pushed down by the search algorithm, such as the second or later pages of a search engine, making it less likely that visitors will see it (most users scan the first page of search results). However, two issues arise. One is that reality is not always black and white, and fact-checkers frequently differ on classifying the content included in computer training sets, risking false positives, and unjustifiable suppression [15]. Furthermore, because fake news evolves quickly, misleading identifiers may become obsolete in the future.

The second strategy entails attaching warnings to erroneous content by expert fact-checkers. Many studies show that corrections and cautions diminish misperceptions and sharing. Despite some early evidence that fact-checking could backfire, further research has revealed that these backfire effects are infrequent. However, the primary issue is that professional fact-checking is not scalable, i.e., it might take a significant amount of time and effort to research each claim. As a result, many (if not most) bogus claims go unchecked. Furthermore, the process is sluggish, and a warning may miss the peak period of viral propagation. Moreover, signs are usually connected to fake news rather than biased coverage of events.

A third strategy prioritizes trustworthy sources such as mainstream media and science communication publications. However, this technique has yielded inconsistent results, as many sites contain hype partisan commentary and confirmation bias, and specific community sections deny all scientific research.

The above, in combination with the fact that users play a key role in how fake news is spread, also users share information with other similar users (friends) and that the characteristics of the network are essential in categorizing or classifying fake news [3], suggesting that the exploring networks of user relationships could greatly facilitate the categorization of fake news [9, 16]. As a result, in this paper, we aim to take advantage of information derived from user friendships and to demonstrate their importance in the problem of detecting fake news. In most cases, standard tensor derivatization or decomposition methods are performed in an unsupervised environment [17]. The class information available for some of the data does not affect them. Instead, our goal is to construct class-aware factor tables using tensor factorization methods. We believe that using class information during and not after the factorization process can give more accurate representations for detecting false news with considerable reliability.

Specifically, using a pioneering DTCN scheme [18] assisted using the tensor Factorization non-negative RESCAL method [19], we can utilize factor-aware tables rather than after the factorization process to produce more accurate representations for localization of fake news with exceptionally high credibility [14].

2. Related Literature

Detecting misleading information ([20] p 19) is a constantly evolving problem, and recently, many survey papers have covered the topic of online fake news under numerous different approaches [7, 9].

Zhang and Ghorbani [3] presented a comprehensive overview of the findings relating to fake news. They assessed the harm caused by online fake news by looking at the various components that make up false news, including the source, audience, target, substance, and the broader social environment in which it circulates. The most up-to-date detection technologies are all about detecting the characteristics that signal it when fighting misinformation. Fake news detection was reviewed by comparing existing detection methods and examining existing datasets for supervised training models. Fake news was also classified using preexisting databases, which the researchers studied. Finally, they outlined several intriguing directions for future research into online fake news.

To address the issue of missing ground-truth data, as well as its high dimensionality and possible redundancy, the authors of [21] present a unique unsupervised feature learning technique for extracting discriminative features from the original data. It learns the compressed representation of unlabeled input using recurrent neural network-based asymmetric autoencoders and can elaborate on spectral and spectral-spatial characteristics. Their extractors can be added to the unsupervised segmentation pipeline and followed by any clustering technique. The trials indicated that their techniques produce high-quality segmentation without prior class labels and are one order of magnitude faster than 3-D convolutional AEs. Their methods outperform or perform similarly to previous methodologies, allowing for significant data reduction.

Pérez-Rosas et al. [11] focused on the automatic identification of fake content in online news. While testing their models, the researchers found that their best models could detect false content with the same level of accuracy as humans can. As a result of their work, they offered two datasets to detect fake stories: one based on seven different types of news domains, six different types of news domains, and one that was gathered from the web that included celebrities. They also explained the collection, annotation, and

validation method and offered many exploratory analyses to discover counterfeit and authentic news content linguistic variations.

Faustini and Covões [4] suggested a method for detecting false news that relied only on text attributes that could be created independent of the source platform. They tested five datasets, including texts and social media postings, in some language groups and found their findings attractive to baselines. They analyzed the accuracy achieved using a particular set of characteristics to those acquired using other common natural language processing approaches. They then ranked the algorithms in order of highest performance. In terms of the feature set, the bag-of-words technique produced the best results but not being statistically more significant than the other attributes. Finally, they created datasets using various channels, such as websites and social media.

Yang et al. [22] developed a text and image information based Convolutional neural network model that could incorporate text and image information with the associated explicit and latent properties. More importantly, it was expandable. They performed trials on a dataset obtained before election procedures and real-world false news records to show that TI-CNN was successful in detecting fake news. The findings demonstrated that using explicit and latent information acquired from multilayer neurons; their method could correctly see bogus info with very high efficiency.

The propagation of fake news has become a global issue, undermining public trust. Fake news and actual news propagate differently on social media. Thus, propagation-based detection systems, which use graph neural networks to form graphs with users as nodes and news sharing chains as edges and simultaneously learn propagation patterns and user preferences, have received much attention. The authors of [23] offer a method for detecting false news using the graph transformer network, which can learn efficient node representations while recognizing functional connections between nodes in the original graph. The proposed method's efficacy is proven by comparison studies utilizing a real-world dataset made of Twitter data.

Antoun et al. [7] proposed techniques for detecting fake news, identifying domains and bots in tweets. They used two distinct approaches to catching fake news. The first used breakthroughs in natural language understanding end-to-end deep learning models to recognize aesthetic distinctions between authentic and counterfeit news pieces. The second was built after the competition and surpassed the winner. The best approach for news domain recognition was a hybrid method that combined named entity characteristics with semantic embeddings generated from end-to-end models [4]. The technique for detecting bots on Twitter consisted of elements taken from news tweets mixed with information from the news tweets. The trials revealed the relevance of several aspects, and the findings suggested that the proposed models performed adequately. They want to enhance false news detection by using elements from fact-checking websites with Google queries.

3. Methodology

Sequence modeling problems have been addressed mainly through feedback neural networks. This work proposes to model the situation using TCN, which achieves top performance and reliability. Based on a discrete convergence operator that produces a map of output characteristics by dragging a kernel over the input, the suggested architecture is designed to create a map of output characteristics f. Using the multiplication between the kernel and the input stride (i.e., the piece of the input with the same size as the kernel), the output characteristic map is constructed. The depth of the output volume is determined by the number of M cores (filters) utilized at each convergent level (i.e., the number of output feature maps). Stride and padding are hyperparameters used to adjust the output feature maps' remaining spatial dimensions [24]. The first can be set to any direction of motion and reflects the distance between two consecutive input strides. Specifically, padding refers to the ability to silently expand the inputs by adding to their limitations (typically zeros) zeros to regulate the size of the output. Without padding, the output dimension would decrease after each convergent level [3, 9, 22].

Considering a one-dimensional sequence $x \in \mathrm{R}nT$ and a one-dimensional nucleus $w \in \mathrm{R}k$, the i-th element of the convergence between x and w is [18, 22, 25]:

$$f(i) = (x * w)(i) = \sum_{j=0}^{k-1} x(i-j)w(j). \tag{1}$$

with $f \in \mathrm{R}nT - k + 1$ if padding is not used; otherwise, the padding has the dimension of the input, i.e., $f \in \mathrm{R}nT$. The previous equation applies to the case of one-dimensional inputs but can easily be extended to inputs of larger dimensions.

In the model we propose, in addition to the contemporary architecture, we also use a fully probabilistic and self-regression model, which can process arbitrary length sequences and extract a sequence of equal length. The network uses causal (expanded) events to achieve this, and the remaining connections are used to handle substantial historical values. In the proposed TCN, the estimated value at time t depends only on previous samples and not on future ones. To achieve this behavior, the causal convergence replaces the standard convergence operator/symbol.

In addition, zero-length padding (filter size ¬ 1) is added to ensure that each level is the same length as the input level. Expanded causal events enhance the network's capabilities further, allowing the network receptive field (i.e., the number of input neurons to which the filter is applied) to increase and its ability to learn long-term dependencies on time series. By this logic, considering a one-dimensional input $x \in \mathrm{R}nT$ and a nucleus $w \in \mathrm{R}k$, an

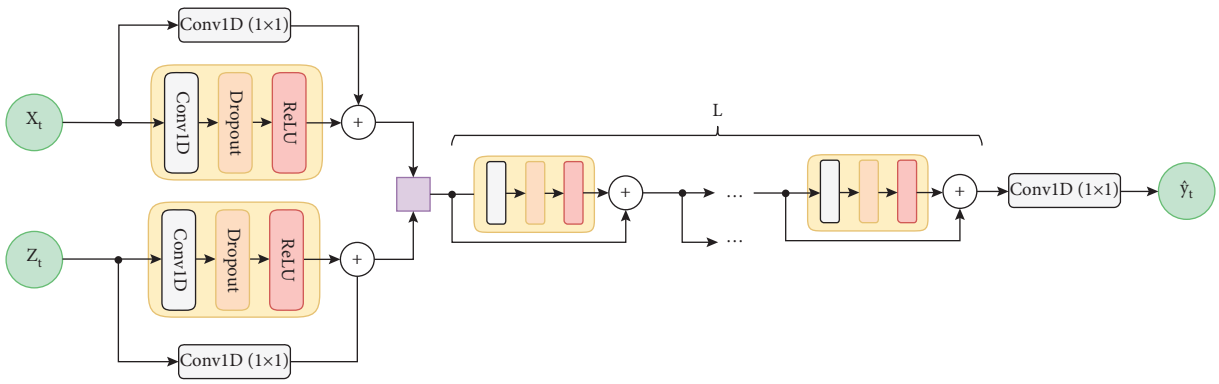


Figure 1: Proposed TCN architecture.

expanded convolution output using an expansion factor d becomes [4, 22, 26]:

$$f(i) = (x *_d w)(i) = \sum_{j=0}^{k-1} x(i - dj) w(j). \quad (2)$$

This is an essential advantage over simple causal convulsions, as in the latter case, the receptive field r increases linearly with the lattice depth $r = k(L-1)$. In contrast, with dilated convulsions, the dependence is exponential $r = 2L - 1k$, ensuring that the network uses a more extensive history size.

Despite the application of expanded convergence, the comprehensive system still needs many levels to learn the dynamics of the inputs. Besides, as it turns out experimentally, the performance often degrades with increasing network depth [27]. The degradation problem was addressed using a deep residual learning framework. Each layer in a typical neural network feeds into the next layer. Each layer in a network with leftover blocks provides the next layer and straight into the levels 2-3 hops away.

There are several interpretations of why residual blocks are lovely and how and why they are one of the essential principles that can enable a neural network to function at a high level on a wide range of tasks. We know, for example, that neural networks are universal function approximators with increasing accuracy as the number of layers grows. However, there is a limit to the number of layers that can be added to improve accuracy. If neural networks were universal function approximators, they should have learned any simplex or complex function. However, because of issues such as vanishing gradients and the curse of dimensionality, it turns out that even with suitably deep networks, it may be unable to learn simple functions such as the identity function. This is unwanted. Furthermore, as the number of layers increases, the accuracy will saturate at some point and subsequently decline. Moreover, in most cases, this is not due to overfitting. As a result, shallower networks can learn faster than deeper networks, which is counterintuitive. However, this is observed in practice and is commonly referred to as the degrading problem. We know that shorter networks outperform deeper equivalents with a few more layers in the degradation problem [19, 21].

In particular, the proposed methodology proposes that for a network of L levels with training error ϵ, introducing k additional levels on it should either leave the error unchanged or improve it. In the worst case, the k new added nonlinear levels should learn the identity mapping $y = H(x) = x$ where x is the output of the L level network and y is the output of the $L + k$-level network. According to the proposed solution, these stacked levels are proposed to fit in a residual mapping $F(x) = H(x) - x$ instead of the desired $H(x)$. Thus, the initial mapping is reshaped to $F(x) + x$, implemented via a simple front-end neural network with shortcut connections. In this way, identity mapping is learned simply by driving stacked level weights to zero values.

The architecture used is shown in Figure 1 [18, 22, 28].

At the first level of the network, the information is processed separately from the external data (when available). Later, the individual results will be combined and processed by the deep residual network L levels [29]. Each level consists of a residual block with one-dimensional expanded causal convolution, ReLU activation function, and dropout to avoid overfitting [30].

The process in the residual block is as shown in Figure 2 [26, 31, 32].

The output plane consists of a 1×1 convolution that allows the network to export a one-dimensional vector $y \in \mathrm{R}nT$ with the same dimension as the input vector x.

To approach multistep prediction, which essentially means that multidimensional data representation is required, we adopt the RESCAL negative factorization method. RESCAL factorization is a new tensor factorization model that scales very well into large amounts of data and can produce state-of-the-art results in many machine learning problems. More specifically, given an X tensor, each slice of X_k is factorized as follows [33, 33, 34]:

$$X_k \approx A R_k A^T, \quad k = 1, \ldots, m, \quad (3)$$

where A is an array $n \times r$ containing the latent representations of the n entities of the problem and R_k is an array $r \times r$ with the latent interactions of the r factors in the k dimension of the tensor. Tables A and R_k are computed by solving the following minimization problem [30, 35, 36]:

$$\begin{gathered} \frac{\min}{A, R_k} f(A, R_k) + g(A, R_k), \\ f(A, R_k) = \frac{1}{2} \sum_k \left\| X_k - A R_k A^T \right\|_f^2, \end{gathered} \quad (4)$$

is the problem of least squares of factorization and

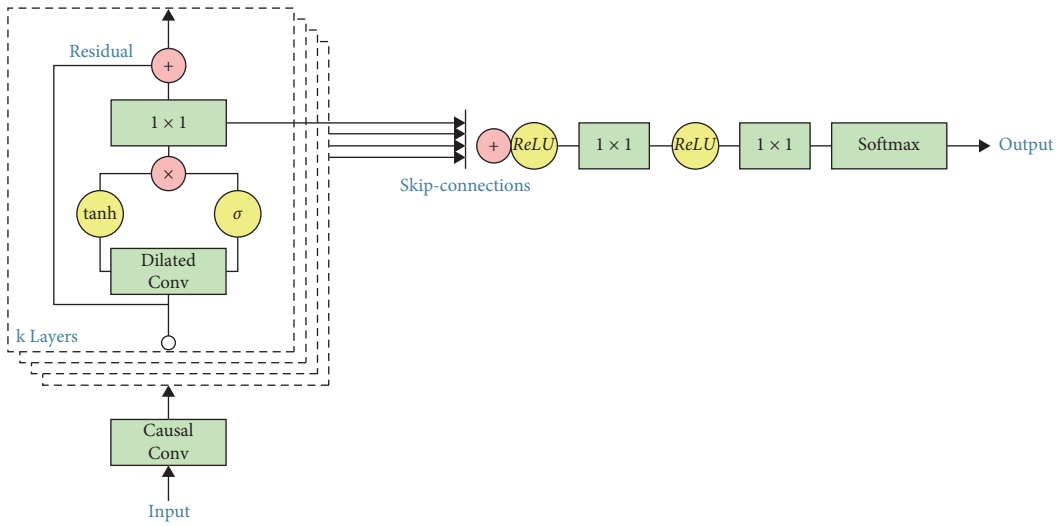


Figure 2: Residual block.

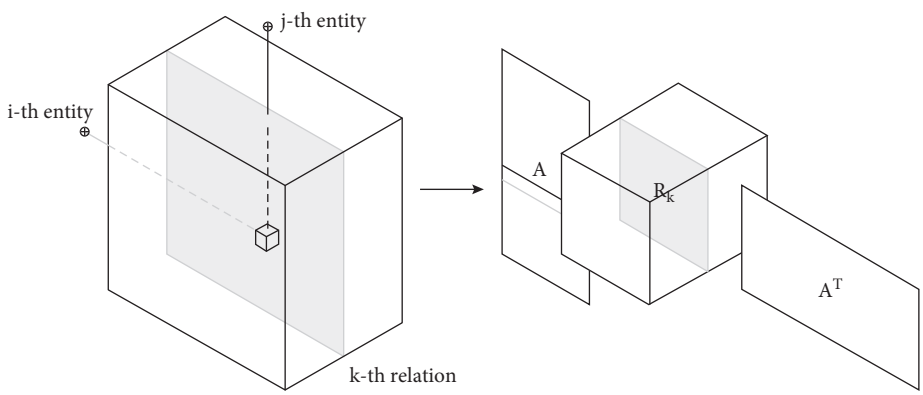


Figure 3: RESCAL factorization.

$$g(A, R_k) = \frac{1}{2}\lambda\left(\|A\|_f^2 + \sum_k R_{kf}^2\right), \tag{5}$$

is the normalization term added to avoid overfitting the model.

Relational data such as that generated by social networks can generally be considered non-negative. Therefore, the data in our problem can be factorized using a non-negative version of RESCAL. First, we represent the data using a third-order tensor. Each slice of T_k is factorized as follows:

$$T_k \approx AR_kA^T, \quad k = 1, \ldots, m, \tag{6}$$

where A is an array $n \times r$ containing the latent representations of n entities and R_k is an array $r \times r$ with the latent interactions of the r factors in the k dimension of the tensor. Parameters A and R_k are calculated by solving the following minimization problem [23, 26, 37]:

$$\begin{aligned} &\min_{A, R_k} f(A, R_k) + g(A, R_k), \\ &f(A, R_k) = \sum_k \left\|T_k - AR_kA^T\right\|_f^2, \end{aligned} \tag{7}$$

is the problem of least squares of factorization and

$$g(A, R_k) = \lambda_A\|A\|_f^2 + \lambda_R \sum_k R_{kf}^2, \tag{8}$$

is the normalization term. The non-negative updates for Tables A and R_k, respectively, are as follows[38, 39]:

$$\begin{aligned} A &\leftarrow A * \frac{\sum_k T_kAR_k^T + T_k^TAR_k}{A\left(\left[\sum_k R_kA^TAR_k^T + R_k^TA^TAR_k\right] + \lambda_A I\right.}, \\ R_k &\leftarrow R_k * \frac{A^TT_kA}{A^TAR_kA^TA + \lambda_R R_k}. \end{aligned} \tag{9}$$

Figure 3 shows the RESCAL Factorization [40, 41].

It is important to note that typically, non-negative tensor factorizations add additional constraints that can lead to complex factor tables that require more time to update, leading to scaling issues.

4. Experiments

The model we developed uses the non-negative RESCAL factorization within a TCN. Suppose, we have $2n$ number of news (posts) that have been posted on social media, $2p$ of

them with class tags, and the remaining $2(n-p)$ without, where $p<n$. There is an equal number of false and actual news [42]. First, we create two third-order tensors using binary representation to model user's friendships: $X_{(\text{fake})} \in \text{Ru} \times u \times p$ containing all news posts marked as false and $X_{(\text{real})} \in \text{Ru} \times u \times p$ with news that has been flagged as valid.

Neighborhood tables represent news posts. By stacking these tables, one after the other, we configure the tensors in a binary way and with whether the user interacted with the fake post - if he is following the user.

It is noted that when we declare that a user has interacted with a post, it means that he has posted the related post/news on his social media profile, in our case, on Twitter. Next, we apply the non-negative RESCAL factorization to the tensors X(fake) and X(real). As a result of the factorization, we end up with tables $A_{\text{fake}} \in \text{Ru} \times r$ and $A_{\text{real}} \in \text{Ru} \times r$, respectively, through the following non-negative updates that result [39, 43, 44]:

$$A_{\text{fake}} \leftarrow A_{\text{fake}} * \frac{\sum_k X_{(\text{fake})k} A_{\text{fake}} R_k + X^T_{(\text{fake})k} A_{\text{fake}} R_k}{A_{\text{fake}}\left(\left[\sum_k R_k (A_{\text{fake}})^T A_{\text{fake}} R_k^T + R_k (A_{\text{fake}})^T A_{\text{fake}} R_k\right] + \lambda_{A_{\text{fake}}} I\right.}, \tag{10}$$

where,

$$R_k \leftarrow R_k * \frac{\sum_k A^T_{\text{fake}} X_{(\text{fake})k} A_{\text{fake}}}{A^T_{\text{fake}} A_{\text{fake}} R_k (A_{\text{fake}})^T A_{\text{fake}} + \lambda_R R_k}, \tag{11}$$

$$A_{\text{real}} \leftarrow A_{\text{real}} * \frac{\sum_k X_{(\text{real})k} A_{\text{real}} R_k + X^T_{(\text{real})k} A_{\text{real}} R_k}{A_{\text{real}}\left(\left[\sum_k R_k (A_{\text{real}})^T A_{\text{real}} R_k^T + R_k (A_{\text{real}})^T A_{\text{real}} R_k\right] + \lambda^I_{A_{\text{real}} I}\right.}, \tag{12}$$

where,

$$R_k \leftarrow R_k * \frac{\sum_k A^T_{\text{real}} X_{(\text{real})k} A_{\text{real}}}{A^T_{\text{real}} A_{\text{real}} R_k (A_{\text{real}})^T A_{\text{real}} + \lambda_R R_k}. \tag{13}$$

The posts that were not used in the train set as slices of the tensors belong to the test set and do not have a tag yet. Suppose, we have a set of arrays $\text{M} = \{P_1, P_2, \ldots, P_2(n-p)\}$, with each array corresponding to a different unlabeled post, where $P_{\text{idx}} \in \text{Ru} \times u$ and $\text{idx} \in [1, 2\ (n-p)]$. The P_{idx} table is created for each post idx in the known binary way described earlier[45].

Next, we perform the non-negative RESCAL factorization on the tensors to generate the new factor tables through the following non-negative updates that occur:

$$A'_{\text{fake}} \leftarrow A'_{\text{fake}} * \frac{\sum_k X'_{(\text{fake})} A'_{\text{fake}} R_k + X'_{(\text{fake})k} A'_{\text{fake}} R_k}{A'_{\text{fake}}\left(\left[\sum_k R_k A^{TT}_{\text{fake}} A'_{\text{fake}} R_k^T + R_k A'^T_{\text{fake}} A'_{\text{fake}} R_k\right] + \lambda'_{A'_{\text{fake}}} I\right.}, \tag{14}$$

where,

$$R_k \leftarrow R_k * \frac{\sum_k A^{TT}_{\text{fake}} X'_{(\text{fake})} A'_{\text{fake}}}{A^{TT}_{\text{fake}} A'_{\text{fake}} R_k A^{TT}_{\text{fake}} A'_{\text{fake}} + \lambda_R R_k}, \tag{15}$$

$$A'_{\text{real}} \leftarrow A'_{\text{real}} * \frac{\sum_k X'_{(\text{real})} A'_{\text{real}} R_k + X'^T_{(\text{real})} A'_{\text{real}} R_k}{A'_{\text{real}}\left(\left[\sum_k R_k A^T_{\text{real}} A'_{\text{real}} R_k^T + R_k A'^T_{\text{real}} A'_{\text{real}} R_k\right] + \lambda_{A'_{\text{real}}} I\right.}, \tag{16}$$

where,:

$$R_k \leftarrow R_k * \frac{\sum_k A'^T_{\text{real}} X_{(\text{real})^k} A'_{\text{real}}}{A'^T_{\text{real}} A'_{\text{real}} R_k A'^T_{\text{real}} A'_{\text{real}} + \lambda_R R_k}. \tag{17}$$

Since the Euclidean distance is a known metric for calculating the distance that is regularly used in similar difference calculation problems, we calculate the prediction as follows: Let l be the still unknown tag of each post, with $l \in [0, 1]$, where 0 indicates an authentic post and 1 a false one. Finally, we remove the P_{idx} table from $X'_{(\text{fake})}$ and $X'_{(\text{real})}$, add the next $P_{\text{idx}+1}$ table from the set of tables M to both tensors, and repeat the procedure for the remaining $2(n-p)$ posts [46, 47].

To evaluate the proposed method, we conducted experiments with two public English-language datasets from two platforms, BuzzFeed and PolitiFact, which include both news content and network information, along with class tags that indicate if each news item is false or true. Content provides information related to the news article's content, such as the title, author, and text, while network information includes information such as user profiles, friendships, and activity. For our evaluation, we use only the knowledge of the network, and in particular, the friendship networks between the users who have posted the relevant news. To reduce the size and sporadicity of the data, we removed the users with node grade <3. Finally, we end up with two tensors of dimensions $182 \times 1449 \times 1449$ and $240 \times 1697 \times 1697$ for the BuzzFeed and PolitiFact datasets, respectively. The Colab environment with GPU was used for the experiments.

To measure the method's performance, we choose the precision evaluation metrics, recall, F1-score, and accuracy, often used in similar problems. The first 70% of the news is

Table 1: Performance of the proposed TCN.

Dataset	Accuracy	Precision	Recall	F1-score
BuzzFeed	93.670	93.770	93.775	93.690
PolitiFact	95.280	95.295	95.290	95.285

the train set, and the remaining 30% is the test set. The number of news items classified as accurate is equal to the number of items marked as false. We perform the experiments ten times, independently for each data set, and record the average results [48, 49].

The results of the process are presented in detail in Table 1.

The abovementioned results show that by using only network data and some class tags, adding class information in the middle and not after the tensor factorization process, and with a small number of factors leading to short computation times, we can achieve outstanding performance, even in problems that require the combination of much heterogeneous information and complex calculations. With this in mind, we can confirm our original hypothesis that explore networks between users that can be helpful in the process of detecting fake news.

In conclusion, the use of the proposed method creates a highly efficient TCN which can exploit a sizeable historical size effectively. With this architecture, lower memory requirements are achieved during training, and predictions for later time steps are not made sequentially. Still, they can be calculated in parallel, taking advantage of parameter sharing. In addition, the training of the proposed system is much more stable than that which includes RNNs because it allows avoiding the problem of explosion/disappearance of inclination.

The success of the method is mainly due to the following three reasons [9, 28, 41, 50, 51]:

(1) Local connectivity: One set of input neurons is connected to each hidden neuron (according to a specific space-time metric). Compared to a fully linked network, this feature significantly reduces the number of parameters that must be learned and facilitates calculations.

(2) Parameter Sharing: The weights used to determine the output neurons in a feature map are the same for each location so that the same kernel is employed. There is less of a learning curve because there are fewer parameters to master.

(3) Translation exchange rate: The network is resistant to a possible shift of its input.

5. Conclusions

In this work, using an innovative DTCN scheme assisted using the tensor Factorization non-negative RESCAL method, we manage to take advantage of class-aware factor tables rather than after the factorization process to produce more accurate representations in detecting fake news with exceptionally high reliability. Instead of applying factorization and classification separately, we proposed a method that combines them into a standard learning process. This approach uses user friendship networks that have interacted with the news and a set of class tags available for some of the news. We proposed a standard RESCAL negative factorization method to combine this information, which incorporates class tags into the factorization itself. In this way, we successfully arrive at a class-aware tensor-aware semi-supervised derivatization. To evaluate the method, we conducted experiments with two public datasets. The results demonstrate integrating class information into the factorization phase as a single process. They also validated our original hypothesis: how individuals connect with news on social media directly affects the news' legitimacy.

As a future extension, we would like to investigate how the proposed methodology is improved when more information is added to it, both from the network and from the news content. In addition, we plan to evaluate the performance of our approach in more databases and investigate the effect of data size on the scalability of the algorithm. It would be interesting to explore new ways of representing the available information with tensors to integrate it into the proposed method at the methodological level. [52–55].

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was a project supported by Hunan Provincial Social Science Fund "Research on Translation of Miao Culture Classics in Western Hunan Area under the Perspective of Cultural Anthropology" (Grant no. 18ZDB005), also by Scientific Research Fund of Hunan Provincial Education Department "A study on the English Translation of Hmong Epics from the Perspective of Ethnographic Thick Translation" (Grant no. 19B130); by First -Class Undergraduate Major in Hunan Province ----Business English (Grant No. (2020)179).

References

[1] P. Surendran, B. Navyasree, H. Kambham, and M. Anand Kumar, "Covid-19 fake news detector using hybrid convolutional and Bi-lstm model," in *Proceedings of the 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, July, 2021.

[2] F. Torgheh, M. R. Keyvanpour, and B. Masoumi, "A new method based on deep learning and time stabilization of the propagation path for fake news detection," in *Proceedings of the 2021 12th International Conference on Information and Knowledge Technology (IKT)*, pp. 57–61, Babol, Iran, December 2021.

[3] X. Zhang and A. A. Ghorbani, "An overview of online fake news: characterization, detection, and discussion,"

Information Processing & Management, vol. 572, Article ID 102025, 2020.

[4] P. H. A. Faustini and T. F. Covões, "Fake news detection in multiple platforms and languages," *Expert Systems with Applications*, vol. 158, p. 113503, 2020.

[5] S. Kumar, S. Kumar, P. Yadav, and M. Bagri, "A Survey on Analysis of Fake News Detection Techniques," in *Proceedings of the International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pp. 894–899, Coimbatore, India, March 2021.

[6] Y. Chang and X. Wang, "Detecting fake news via deep learning techniques," in *Proceedings of the ICMLCA 2021; 2nd International Conference on Machine Learning and Computer Application*, pp. 1–4, Shenyang, China, December 2021.

[7] W. Antoun, F. Baly, R. Achour, A. Hussein, and H. Hajj, "State of the art models for fake news detection tasks," in *Proceedings of the 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, pp. 519–524, Doha, Qatar, Feb 2020.

[8] S. Patil, S. Vairagade, and D. Theng, "Machine learning techniques for the classification of fake news," in *Proceedings of the 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA)*, pp. 1–5, Nagpur, India, November 2021.

[9] D. de Beer and M. Matthee, "Approaches to identify fake news: a systematic literature review," in *Integrated Science in Digital Age 2020*, T. Antipova, Ed., vol. 136, pp. 13–22, 2021.

[10] S. Mishra, P. Shukla, and R. Agarwal, "Analyzing machine learning enabled fake news detection techniques for diversified datasets," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1–18, Article ID 1575365, 2022.

[11] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, and R. Mihalcea, "Automatic Detection of Fake News," 2017, https://arxiv.org/abs/1708.07104.

[12] D. Li, H. Guo, Z. Wang, and Z. Zheng, "Unsupervised fake news detection based on autoencoder," *IEEE Access*, vol. 9, pp. 29356–29365, 2021.

[13] P. Qi, J. Cao, T. Yang, J. Guo, and J. Li, "Exploiting multi-domain visual information for fake news detection," in *Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM)*, pp. 518–527, Beijing, China, November 2019.

[14] P. Devika, A. Veena, E. Srilakshmi, A. R. Reddy, and E. Praveen, "Detection of fake reviews using NLP &Sentiment Analysis," in *Proceedings of the 2021 6th International Conference on Communication and Electronics Systems (ICCES)*, pp. 1534–1537, Coimbatre, India, July 2021.

[15] K. Nath, P. Soni, A. Ahuja, and R. Katarya, "Study of fake news detection using machine learning and deep learning classification methods," in *Proceedings of the 2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, pp. 434–438, Bangalore, India, Augest 2021.

[16] K. Sharma, F. Qian, H. Jiang, N. Ruchansky, M. Zhang, and Y. Liu, "Combating fake news," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 3, pp. 1–42, 2019.

[17] A. Kumar J, T. Esther Trueman, and E. Cambria, "Fake news detection using XLNet fine-tuning model," in *2021 International Conference on Computational Intelligence and Computing Applications (ICCICA)*, pp. 1–4, Nagpur, India, November 2021.

[18] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," 2016, http://arxiv.org/abs/1611.05267.

[19] D. Krompaß, M. Nickel, X. Jiang, and V. Tresp, *Non-Negative Tensor Factorization with*, p. 10.

[20] M. K. Elhadad, K. F. Li, and F. Gebali, "Detecting misleading information on COVID-19," *IEEE Access*, vol. 8, pp. 165201–165215, 2020.

[21] L. Tulczyjew, M. Kawulok, and J. Nalepa, "Unsupervised feature learning using recurrent neural nets for segmenting hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 12, pp. 2142–2146, 2021.

[22] Y. Yang, L. Zheng, J. Zhang, Q. Cui, Z. Li, and P. S. Yu, "TI-CNN: Convolutional Neural Networks for Fake News Detection," 2018, https://arxiv.org/abs/1806.00749.

[23] H. Matsumoto, S. Yoshida, and M. Muneyasu, "Propagation-based fake news detection using graph neural networks with transformer," in *Proceedings of the 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE)*, pp. 19-20, Kyoto, Japan, Octomber 2021.

[24] Y. Ma, X. Wang, Z. Dong, and H. Chen, "Cascaded LSTMs based deep reinforcement learning for goal-driven dialogue," *Natural Language Processing and Chinese Computing*, vol. 10619, pp. 29–41, 2018.

[25] A. Dhillon and G. K. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence*, vol. 9, no. 2, pp. 85–112, 2020.

[26] Y. Feng and X. Xiao, "An efficient residual-convolutional neural model for handwritten text recognition," in *Proceedings of the 2019 15th International Conference on Semantics, Knowledge and Grids (SKG)*, pp. 115–118, Guangzhou, China, September 2019.

[27] G. Khan, Z. Tariq, J. Hussain, M. A. Farooq, and M. U. G. khan, "Segmentation of crowd into multiple constituents using modified mask R-CNN based on mutual positioning of human," in *Proceedings of the International Conference on Communication Technologies (ComTech)*, pp. 19–25, Rawalpindi, Pakistan, March 2019.

[28] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.

[29] J. Du, L. Gui, R. Xu, and Y. He, "A convolutional attention model for text classification," *Natural Language Processing and Chinese Computing*, vol. 10619, pp. 183–195, 2018.

[30] P. Bhattacharya, S. B. Patel, R. Gupta, S. Tanwar, and J. J. P. C. Rodrigues, "SaTYa: trusted Bi-LSTM-Based fake news classification scheme for smart community," *IEEE Transactions on Computational Social Systems*, no. –10, pp. 1–10, 2021.

[31] Y. Wang, Y. Yang, W. Ding, and S. Li, "A residual-attention offline handwritten Chinese text recognition based on fully convolutional neural networks," *IEEE Access*, vol. 9, pp. 132301–132310, 2021.

[32] A. Bala, I. Ismail, R. Ibrahim, and S. M. Sait, "Applications of metaheuristics in reservoir computing techniques: a review," *IEEE Access*, vol. 6, pp. 58012–58029, 2018.

[33] K. Zhang, M. Sun, T. X. Han, X. Yuan, L. Guo, and T. Liu, "Residual networks of residual networks: multilevel residual networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 6, pp. 1303–1314, 2018.

[34] S. Y. Arafat and M. J. Iqbal, "Urdu-text detection and recognition in natural scene images using deep learning," *IEEE Access*, vol. 8, pp. 96787–96803, 2020.

[35] G. Peters, M. Lampart, and R. Weber, "Evolutionary rough k-medoid clustering," in *Transactions on Rough Sets VIII*, J. F. Peters and A. Skowron, Eds., vol. 5084pp. 289–306, 2008.

[36] Q. Rao, B. Yu, K. He, and B. Feng, "Regularization and iterative initialization of softmax for fast training of convolutional neural networks," in *Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Budapest, Hungary, July 2019.

[37] J. Gawlikowskiet al., "A Survey of Uncertainty in Deep Neural Networks," 2021, http://arxiv.org/abs/2107.03342.

[38] M. Cai, Y. Shi, J. Kang, J. Liu, and T. Su, "Convolutional maxout neural networks for low-resource speech recognition," in *Proceedings of the 9th International Symposium on Chinese Spoken Language Processing*, pp. 133–137, Singapore, September 2014.

[39] M. A. Hussain and T.-H. Tsai, "An efficient and fast softmax hardware architecture (EFSHA) for deep neural networks," in *Proceedings of the 2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pp. 1–4, Washington DC, DC, USA, June 2021.

[40] C. C. Aggarwal, "Neighborhood-based collaborative filtering," in *Recommender Systems*, C. C. Aggarwal, Ed., Springer, Cham, pp. 29–70, 2016.

[41] M. Srifi, A. Oussous, A. Ait Lahcen, and S. Mouline, "Recommender systems based on collaborative filtering using review texts-A survey," *Information*, vol. 11, no. 6, pp. 317–6, 2020.

[42] K. Chatterjee and D. Mukherjee, "A new integrated likelihood for estimating population size in dependent dual-record system," *Canadian Journal of Statistics*, vol. 46, no. 4, pp. 577–592, 2018.

[43] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., vol. 4321, pp. 291–324, 2007.

[44] K. T. Ilayarajaa, V. Vijayakumar, M. Sugadev, and T. Ravi, "Text recognition in moving vehicles using deep learning neural networks," in *Proceedings of the 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pp. 279–283, Coimbatore, India, March 2021.

[45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[46] Y. Yanagi, R. Orihara, Y. Sei, Y. Tahara, and A. Ohsuga, "Fake news detection with generated comments for news articles," in *Proceedings of the 2020 IEEE 24th International Conference on Intelligent Engineering Systems (INES)*, pp. 85–90, Reykjavík, Iceland, July 2020.

[47] A. C. Rivera, M. Tapia-Leon, and S. Lujan-Mora, "Recommendation systems in education: a systematic mapping study," *Proceedings of the International Conference on Information Technology & Systems (ICITS 2018)*, pp. 937–947, Cham, January 2018.

[48] S. S. Haykin and S. S. Haykin, *Neural Networks and Learning Machines*, Prentice-Hall, New York, 3rd ed edition, 2009.

[49] L. E. B. Salasar, J. G. Leite, and F. Louzada, "Likelihood-based inference for population size in a capture-recapture experiment with varying probabilities from occasion to occasion," *Brazilian Journal of Probability and Statistics*, vol. 30, no. 1, pp. 47–69, 2016.

[50] S. M. M. Seyednezhad, K. N. Cozart, J. A. Bowllan, and A. O. Smith, "A Review on Recommendation Systems: Context-Aware to Social-Based," 2018, http://arxiv.org/abs/1811.11866.

[51] A. Gasparin, S. Lukovic, and C. Alippi, "Deep Learning for Time Series Forecasting: The Electric Load Case," 2019, http://arxiv.org/abs/1907.09207.

[52] N. Kimura, I. Yoshinaga, K. Sekijima, I. Azechi, and D. Baba, "Convolutional neural network coupled with a transfer-learning approach for time-series flood predictions," *Water*, vol. 12, no. 1, p. 96, 2020.

[53] J. Dai, Y. Li, K. He, J. Sun, and R. Fcn, "Object Detection via Region-Based Fully Convolutional Networks," 2016, http://arxiv.org/abs/1605.06409.

[54] N. Snell, W. Fleck, T. Traylor, and J. Straub, "Manually classified real and fake news articles," in *Proceedings of the 2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 1405–1407, Las Vegas, NV, USA, December 2019.

[55] M. Cai, Y. Shi, and J. Liu, "Deep maxout neural networks for speech recognition," in *Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 291–296, Olomouc, Czech Republica, December 2013.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 2254411, 12 pages
https://doi.org/10.1155/2022/2254411

Research Article

A Blockchain-Based Verifiable User Data Access Control Policy for Secured Cloud Data Storage

Xinlong LI

School of Computer Science, Hunan Institute of Technology, Hengyang 421002, China

Correspondence should be addressed to Xinlong LI; li_xinlong26@yeah.net

Received 16 March 2022; Revised 7 April 2022; Accepted 11 April 2022; Published 27 April 2022

Academic Editor: Konstantinos Demertzis

Adding the adequate level of security of information systems dealing with sensitive data, privacy, or defense systems involves some form of access control. The audits performed are dealing with the determination of the allowed activities of the legal users, when attempting to access resources of the system. Usually, full access is provided after the user has been successfully authenticated through an authentication mechanism (e.g., password), while the corresponding authorization control is based on the confidentiality level of the respective resources and the authorization level assigned to each user. A very important diversification occurring in modern digital technologies is related to the identification based on blockchain technology, which is presented as a public, distributed data series, unable to modify its history and grouped in time-numbered blocks. In this work, a blockchain-based verifiable user data access control policy for secured cloud data storage is suggested for a version associated with big data in health care. It is an innovative system of applying classified access policies to secure resources in the cloud, which operates based on blockchain technology. System evaluation is carried out by studying a case in its resilience to Eclipse attack under different malicious user capabilities for routing table poisoning.

1. Introduction

Cloud data access control requires cooperation between processing sectors and at the same time protected and managed as a computational collaborative environment consisting of computational units under the management of distributed access control [1]. An access control state is said to be secure if no permission for access can lead to an unauthorized person. Blockchain [2] is a transparent, verifiable, permanent transactions management system operating and distributed in peer networks, offering and maintaining a robust mechanism of consensus, which, unlike the usual procedures, does not base its credibility and solvency on some reliable third entity.

In public blockchains, anyone interested can participate in the network, as access to their data is open by reading the chain and verifying the blocks, thus creating transparency in the information. This achieves the secure decentralization of the system since the members do not need to trust each other. On the other hand, there are many cases of applications where transactions or assets do not have to be disclosed or accessible to all, but by selected participants. Such transactions may be between competitors, medical history, transportation of goods, etc. [3]. That is the main reason why private blockchains were created. They are useful in cases where the integrity of the trace is not the most important prerequisite, and there is a need to standardize the exchange of information in a secure way between partners.

Combining the cloud and the blockchain can result in a verifiable [1, 4], permanent, and unmodified file in terms of data sharing between a private chain, eliminating the primary issue of supervision by allowing anybody permitted to join the network to observe and evaluate the activities transparently. In the event that something goes wrong with the processes (e.g., information leaks), blockchain makes it fairly straightforward to find the weak node [5]. Furthermore, the existence of a central service for storing and processing authentication information is not required for blockchain.

This capability is further strengthened by the blockchain's smart contracts [6]. They seek to provide security

above and beyond contract law as well as to lower the additional transaction costs involved with the award and implementation of intermediate contracts. Furthermore, all users of this network can see blockchain-based contracts. This feature improves transaction transparency and dependability in complex contexts by intelligently automating the approval of a framework for carrying out a preagreed process when conditions appear that both sides have delivered the preagreed services. It's worth noting that smart contracts on the blockchain can go beyond simple activities and include more detailed instructions in their code. Applying certain rules that regulate a wide range of options [6, 7]:

(1) Fact-based functionality: when triggered by external data that identify a specific and preagreed event (without them being primarily related to human intervention), smart contracts can modify other data.

(2) Functionality based on external data: these data can be provided by reliable data sources that can provide dynamic, feedback information in smart contracts.

(3) Functionality based on enforcement and proof: contracts may, based on the information provided, "enforce the functional application of a particular requirement and may demonstrate that certain conditions are met or not met.

(4) Functionality based on changes. These capabilities involve monitoring changes in system status over time and adapting to them.

In this paper, a blockchain-based verifiable user data access control policy for secured big data storage in the cloud is proposed based on the design of a data exchange network between systems that use cloud computing utilizing blockchain technology. The systems will be able to transmit securely, control, and detect data while sharing medical data with other medical institutions and research institutes without any risk to their privacy. The method includes utilizing blockchain components to distinguish the suspicious behavior successfully and repudiate access with the implementation of the model. The aim is to produce a cutting-edge system for applying classified access policies to secure cloud resources powered and enhanced by blockchain technology.

2. Literature Review

The use of the blockchain technology is relative concept in the research community. However, researchers have focused on the utilization of the most aspects of this innovative technology, and one of the most promising areas of research is the combination of the access control mechanisms with the blockchain. Chinnasamy et al. [8] in 2017 presented for the first time a distributed access control framework by combining blockchain with an access control model. They introduced smart contracts as a way to implement contextual access control restrictions and make authorization decisions. They also used blockchain to enforce access policies in dispersed situations where there is no central authority and to ensure that policies are enforced correctly and uniformly.

Also, Macías and Guitart [9] proposed using blockchain technology as an access control tool for representing and transferring resource access rights from one user to another. They advocated storing the representation of these rights in the form of transactions on the blockchain. They also employed attribute-based access control (ABAC) policies, which combine a collection of rules expressing conditions over a set of attributes associated with the subject, resource, or environment. In addition, Uchibeke et al. [10] in 2018 implemented identity-based access control (IBAC) and role-based access control (RBAC) on the Hyperledger Fabric blockchain, a private and permissioned scheme led by IBM, to achieve access control methods for big data (RBAC).

They built the request, grant, revoke, verify access, and view asset actions for each access control model. Finally, they contrasted the outcomes of both implementations and discussed the stability difficulties caused by the Hyperledger Fabric blockchain's newness. Finally, Rouhani and Deters [3] gave an outline of the current access control techniques' difficulties and how the blockchain can assist overcome them. They also looked at the obstacles that come with adopting a blockchain-based access control system as well as presenting an overview of related research projects and categorizing them based on different domains and access control methods.

On the other hand, Ghaffari et al. [11] conducted a comprehensive study to provide a comprehensive picture of the current state of the art in integrating blockchain and smart contracts in access control and authentication techniques. They began by outlining the history of distributed ledger technology, proposing a taxonomy for categorizing current methods based on type, application environment, and blockchain exploitation. They also looked at existing blockchain-based authentication and access control mechanisms in a variety of settings, including health care. Algarni et al. [12] suggested a solution based on a multiagent system and a blockchain to handle the delivery of lightweight and decentralized secure access control of an IoT system. The fundamental goal of this strategy was to create blockchain managers (BCMs) to secure IoT access control and allow secure communication between local IoT devices. Dar et al. [13] in 2021 attempted to give an analysis of the available empirical evidence by attempting to synthesize the literature in order to comprehend the state of the art in blockchain-based access control methods for underlying platforms. They found a sufficient number of relevant primary research and focused on many topics such as single point of failure, security, and privacy.

They also conducted a meta-analysis and thematic synthesis on the utilization of various blockchain platforms, application domains, and blockchain features. Gao et al. [14] proposed a blockchain-based security sharing mechanism for personal data as a solution to this challenge. They combined four independent components: the blockchain, ciphertext policy, attribute-based encryption (CP-ABE), and the interplanetary file system (IPFS). To maximize the scheme's decentralization, this is a user-centric scheme in

which the data owner encrypts the sharing data and saves it on IPFS.

Most of the above literature is utilizing the blockchain technology but rarely evaluate their work against certain cyberattacks. In the present work, we not only propose a novel scheme for a specific sector like health care but we also compare it against specific threats.

3. The Proposed System

To achieve the high demands on big data storage, the cloud computing mechanism offers a solution because it provides controlled and flexible data processing and exchange mechanisms as well as their respective storage spaces [15]. The increased interest has expanded in the field of health, including medical and research institutions and their cooperation. But despite the advantages that cloud computing offers, it lacks the functionality associated with data exchange due to the risks involved in exposing its content. For data proprietors, there's a risk that the data collected will end in the hands of malevolent users. In this context, the fear of violating the regulations and the exploitation of data creates an atmosphere of mistrust that does not ensure the implementation of data exchange. Blockchain technology can offer the right solution to deal with this problem through its attractive properties such as its decentralized and unchanging nature [9, 16].

3.1. Basic Functions of the System. The model proposed and described is based on the blockchain mechanism and specifically on the properties of smart contracts, but also cloud computing, and is used to exchange medical records between service providers, providing data control and at the same time proper management of their large volume. The actions of the beneficiaries are constantly monitored with the contribution of various mechanisms, and the violations are treated.

3.1.1. Blockchain Network. The pieces of information are stored in the blockchain. The requests that the system receives from external users for access to the desired data are created into blocks and are later transmitted to the chain during the delivery of the package to the user. The last action completes the creation of the block and allows its transmission to the blockchain network. Each block is identified by its unique value which is also its identity. The significance of executing side blocks in the network is to preserve an effective log to investigate violations of terms [17].

They are attached to parent blocks and include indexed references, identical to those listed in the smart contracts database. Creating multiple network connections brings together a complete collection of reports. A block is created from a processed form, which represents a request received from an external user and contains information related to the receipt of the request, the processing, and delivery of the data.

A peer-to-peer network is outlined on the concept of peers who work at the same time as clients and servers to the other hubs of the arrangement. The foremost common application of peer-to-peer organization is the distributed hash table (DHT) [16], which employs a hash function to certify ownership to the organization of nodes [18]. This allows peers to find resources employing a hash table: the records are stored in DHT in pairs [key, value], and each node can recover the value related to a given key.

A DHT is a sophisticated decentralized framework that gives an effective research mechanism in which any participant node can recover the value related to a given key. Each node needs to be coordinated with only a small part of the total system nodes—usually O (log n) where n is the system nodes—so that it needs to be a small amount of work for each change in the participating nodes (e.g., withdrawal). There are some classic issues that DHTs must deal with, such as load balancing, data integrity, and performance.

Nodes and keys receive m-bit IDs, for which the basic hash function is the SHA-256 algorithm, and consistent hashing is essential for chord robustness and high performance. According to the chord search protocol, nodes and keys are arranged in a circle of identifiers containing $2m$ positions, with values from 0 to $2m-1$ (the m should be large to avoid collisions). Each node has a successor and a predecessor.

The successor to a node is the next one in the clock cycle. Respectively, its predecessor is the immediately preceding one (at the same direction of rotation). If there is a node for each possible ID, then node 0 is the successor to node 1 and its predecessor is node $2m-1$. Of course, usually, there are "gaps" in the sequences of nodes.

For example, the successor of node 159 can be node 200 (there are no nodes with IDs between 159 and 200), which means that node 200 has a predecessor the node 159 [19, 20].

When a new node is entered, three properties must be retained [16, 21]:

(1) The successor of each node must point correctly to the next one

(2) Each key k must be stored by the successor (k)

The finger table of each node must be correct.

Every network transaction has one or more inputs and outputs, all of which are recorded on the blockchain. These outputs create chunks, which are recognized by the whole network and made available to the owner for future transactions. In addition, each input/output has a time-stamped function associated with it.

The hash outputs of transactions are used to uniquely identify them, whereas the output index of specific transactions is used to identify them. Figure 1 depicts an example of this technique.

Network's proof of work takes advantage of the seemingly random nature of cryptographic hashes. A party must construct a hash of the block header that does not exceed a particular value in order to establish that it did a given amount of computational labor to create a block. The hashing technique used is double SHA-256, and the specified structure is a hash that is less than or equal to a target value T. The purpose is to find a hash that is numerically less than

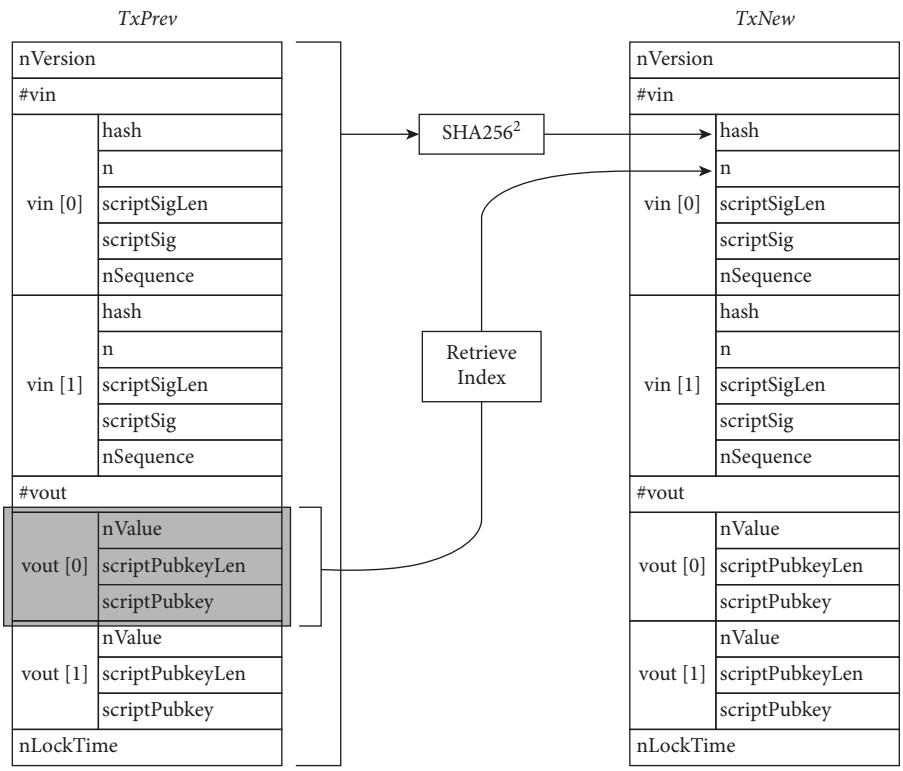


Figure 1: Transaction output reference computation.

the target, which we name the value threshold target. We alter a variable called nonce every time we want to change the hash result, usually by incrementing it by one. The likelihood of finding a nonce n for a given message (msg) such that $H = \text{SHA2562}(\text{msg}||n)$ is less than or equal to the target T is [16, 20]

$$P[H \leq T] = \frac{T}{2^{256}}. \tag{1}$$

The following quantity of computations is the average number of tries completed by a party attempting to find a proof of work:

$$\begin{aligned} T[H \leq T] &= \frac{1}{P[H \leq T]} \\ &= \frac{2^{256}}{T}. \end{aligned} \tag{2}$$

Finally, by simply assessing the nonce that comes with the message, it is simple and quick to determine whether it is genuine proof of work:

$$\text{SHA256}^2(\text{msg} \parallel n) \leq T. \tag{3}$$

3.1.2. Cryptographic Keys. Encryption keys [5, 22, 23] are labeled to perform specific tasks related to their security on the system. For the exchange and transmission of data between "unreliable" nodes, encryption keys are required, ensuring a level of security in the system. Specifically, the user's private key that sends the request for access to the system creates its private key and uses it to put its own "digital signature" on it. Respectively, the public-key of the user sends it in combination with the request, the public-key that he has created to be used for the verification of his identity through the control of the digital signature. The smart contract key is also a pair of keys generated by the authenticator which are attached to the smart contract delivered to the user so that he can decrypt the data he received but at the same time follow the rules of the smart contract so that there is control over the use of data by the system [24, 25].

The following is the mathematical formula for deriving public-key cryptography, with C denoting the encrypted message:

$$\begin{aligned} C &= \text{encrypt}(M, K\text{pub}), \\ M &= \text{decrypt}(C, K\text{pri}). \end{aligned} \tag{4}$$

The suggested method's public-key cryptography is based on elliptic curve multiplication. The following function, which produces an elliptic curve, is used to define the curve:

$$y^2 = \left(x^3 + 7\right) \bmod p. \tag{5}$$

In summary, a user who wants to access file sets from the system and the data holder creates a pair of private and public-keys. Upon receiving, the data holder confirms the validity of the request and the identity of the user, verifying the public-key signature of the user. The results obtained from the retrieval of the requested files are processed by the

system, and then, before the file is delivered to the user, it is encrypted with a "contract key," which is attached to the smart contract that is sent along with the data. By decrypting the file, the data holder gains full control over the actions performed by the user, as the smart contract is automatically activated [1, 24, 26].

3.1.3. Triggers. The main role of the application of triggers is to allow smart contracts to indirectly connect the system with the external environment of the system since the latter cannot interact directly with structures outside the network [27]. They do not hold any information and only act as intermediaries for the smooth communication of the level of requests with the level of processing them. Triggers also update process statements to and from the level of requests based on smart contract features.

3.2. System Design. The design of the system is based on open architecture systems, where there is the independence that ensures the smooth cooperation and operation between the individual operating applications and subsystems of the information system and the network cooperation between applications and/or systems located in different computer systems. Its modular architecture also allows for future extensions and replacements, integrations, upgrades, or changes to discrete software or hardware components. Finally, the n-tier architecture allows the flexibility of cost and load distribution between central systems and workstations for the efficient operation of the network and the ease of its scalability.

3.2.1. Users. They are all users whose intention is to access medical data, either for clinical or research purposes (e.g., health-care organizations, hospitals, research institutes, universities, and research scientists). Users send requests to the system for access to the data, which are subjected to a processing process.

3.2.2. Request Receipt Field. The model consists of structures that receive, process, and respond to requests placed in the system and related to access to existing data. This level interacts directly with the data processing and transmission level and has built-in mechanisms to interpret and translate actions between the internal and external environment. In addition, users communicate directly with this mechanism to send requests. Its structural elements are [2] as follows:

(1) Request conversion structure: it is responsible for converting requests into a format that can be recognized by the data processing and transmission field. The conversion results in a value that replaces the request and can be read by the system to retrieve the requested information. His final role is to respond and send a "response" to the applicant based on the request he has made.

(2) Structure of "translation" of smart contracts [8, 28]: this system has the responsibility to translate the actions of smart contracts to and from their environment as it cannot operate outside a blockchain network autonomously.

3.2.3. Data Processing and Transmission. The model includes components that assist the user's request for data access. Additionally, calculations are performed on the requested data, and functions are added which detect any action. Algorithmic processes are applied to the data and undertake to report on the actions performed. The results of each action that has been completed are transmitted to an unchanged network that guarantees fair control.

The system is also responsible for authenticating any request and action regarding access to digital medical records. Existing level entities are authenticator, nodes of processing and consent, smart contracts production structure, smart contract database, and blockchain network [2, 6, 29].

3.2.4. Cloud Computing Database. The database contains functions that are used to perform specific tasks. Only authorized personnel from the consent nodes have access to this system, as they host private information that requires safe methodologies adequately to ensure high-level protection. To access the data in this database, the required information is transmitted through calculations so that it can be shared [1, 30].

3.3. System Functionality. The operation of the individual applications, subsystems, and solutions consisting of the distinct parts of the information system ensure that the greatest possible uniformity is achieved in the interfaces between the different subsystems and in the way they operate, and common and friendly presentation modes will be chosen in terms of user interfaces, with applications and system scalability to be ensured. Also, the use of flexible management systems allows the functional control of the large volume of data, the increased availability of the system, and the possibility of controlling access to the data [3, 21].

3.3.1. Reception of the Request. The user sends a request to access specific data. The request is digitally signed by the user via his previously created private key. The request initially meets the level of receipt of requests. The triggers in the system convert the request into a structure that can be read by the system that processes and transmits the data and transmit it to that level. Initially, the authenticator verifies the legality of the request by checking the signature, using the corresponding public-key of the applicant which has been distributed by the user when sending his request [17]. The process proceeds further if there is a valid signature, otherwise interrupted, and considered as an invalid request.

3.3.2. Request Processing. Once the request is approved, the processing and consent nodes undertake to convert it into a

suitable form which will include, in addition to the desired data, a unique value representing the identity of the applicant (user ID) but also a time stamp of the time of receipt of the request. The two values are attached to the form, after first being hashed, through a mathematical hash function. The reason for which the specific data are requested is also indicated in the form, and finally, it is transferred to the existing database.

This database takes the form, retrieves the requested data, and sends it back to the processing and source level where the first modification by the consent nodes will occur. The time stamp of the request created on the form will be noted in the retrieved information. Consent nodes then send a request to the center of smart contracts to establish rules about the requested data [6, 7]. The corresponding smart contract will be generated and integrated into the form along with the data.

3.3.3. Distribution of the Requested Data. The new form that is the result of the previous processing is sent to the authenticator to undergo through the final stage. The authenticator generates an encryption key and points it to the smart contract that has been created. With the key, the user will be able to decrypt the requested data. This is important to ensure the secure transmission and detection of information. At the same time, the consent nodes construct a chain of block-based piece of data requested by the user and transmit it to the blockchain according to the chronological order in which it is created. As expected, the block will have a unique identifying value, following the cryptographic methods it has been subjected to, following the blockchain network way of operation.

The packet that has been created by processing the requested data retrieved from the existing database includes the data, the value of their "identity" (data ID), and the smart contract with the terms of use of the data. Eventually, the entire packet is encrypted by the authenticator so that it can only be identified by the holder of the appropriate private key, and by entering the user ID, it is sent back to the request system from where it was initiated [1, 30]. The smart contract is the reason for the effective monitoring of the package.

3.3.4. Delivery of the Data to the User. The user receives the edited packet and decrypts it with his private key. His security must somehow be validated. At this point, the contracts that have been configured by the processing system will play a key role. With the key attached to the smart contract, the user decrypts the data, and it is automatically activated. Any action on the decrypted data received by the user is reported and sent to the level of receipt of requests, from where it is translated and transferred to the level of processing and specifically to the consent nodes. They, in turn, store the reference in the blockchain chain to a side block that is inextricably linked to the block added during the previous procedure.

The reason for keeping the file containing the actions performed in the data is to prevent their malicious use. The installation of such reports reflects the ability of the smart contract to activate specific conditions when performing any transaction that is directly related to the requested data [31, 32]. Through this property, the control of the documents available to the user is achieved.

3.3.5. The Function of Smart Contracts. Smart contracts operate as systems that execute predefined instructions when performing actions that follow an organized framework. They are used to report actions related to the data requested by the user system and allow the data owners to secure and control them, as they will be monitored in a controlled environment, eliminating the relationship of trust required between the owner and the user. As mentioned above, reports about the actions of the data resulting from the user's system are updated and transmitted to the blockchain network. A set of actions can be applied to the data received by the user, which will activate the rule-based smart contracts. Data sensitivity can be categorized into high and low.

This is determined by the consent nodes when they obtain the data from the existing database. Based on the degree of importance of the package, some actions are excluded from the list of malicious acts, while others are violations [6, 28].

The identity of the data specified in the smart contracts gives an advantage in creating an effective medium so that the consent nodes can map, process, and verify the corresponding unique block. Comments are generated to describe the user-performed actions in the data. In most cases, they are comments of infringement or exclusion.

By extracting a key through specific commands, they are encrypted and stored in the smart contract database. The rights declared by the data holder are defined on the smart contracts. Unreliable data are handled appropriately by the owner [22, 26].

3.3.6. Data Exchange between the Database and the Consent Nodes. The exchange of data is crucial for secure operation of the information-sharing between entities where there is no trust. The data output from the database must maintain the integrity, and for this reason, their exchange methods need to be designed and structured with great care. For the approved request, the database makes a copy of the data and forwards it to the consent nodes which are responsible for configuring the entire package. The package includes, in addition to the data, an identifier of these (data ID) but also an identifier of the consent node that undertook the processing. The node in charge of the modification verifies the data received by comparing its type with the requested request. They are classified on a scale characterized by high or low sensitivity. For a highly sensitive data set, there is a need for greater security and anonymity.

The actions performed using the sent information are recorded in a format that will eventually convert them into blocks and will be added to the network. The result is obtained from the data management node by node. Once this is considered accurate, they are returned to the first node. The consent node sends a request stating the level of data

sensitivity to the smart contract generator to generate the corresponding contract with the rules.

Eventually, it is attached with the requested data and the completed file is encrypted by the authenticator with the user's public-key, and a time stamp is issued at the end of the process. All processing times are recorded by the consent node to allow efficiency-based optimization. In addition, the form with the performed actions includes the contribution of the second node [1]. The file is formatted in blocks and is now ready to be added to the blockchain system.

3.3.7. Main Block Structure in the Blockchain Chain. Each block, as mentioned in the description of blockchain technology, is uniquely identified and described by a fragmented value that has been calculated. The block includes its size as well as the block header. The blockchain uses SHA-256 to generate the hash value of a message M. The result of the SHA-256 is a 256-bit message summary [16, 21].

The header has gone through the fragmentation process through the SHA-256 algorithm and plays an important role in the blockchain, making it unchanged. It contains the fragmented value of the previous block that was added to the chain, so any change to a block should change the entire chain starting with the original block, the genesis block.

This fact ensures the integrity of the network since there is a maximum guarantee that it is not possible to achieve this goal. The mechanism also guarantees the origin of the data, so in case of malicious activity, the mismatch of the blocks will warn the system to enable accurate data verification. The block header consists of the rules to be followed for data validation in the block and the properties that will have [25].

In addition to the fragmented value of the previous block contained in the header, part of it is the Merkle tree root, which contributes to the security of the chain by ensuring that none of the blocks can be modified without transforming the header. The Merkle tree root results from the hashing of all records received by the block. The output is the result of the SHA-256 algorithm as used throughout the header. An important part of the heading is the time stamp of the creation of the block and a nonce value, which is a random number set by the consent nodes to generate the fragmented header value in conjunction with the target difficulty value [33].

The block contains an activity counter, the function of which is to log the amount of malicious attempts concerning the data recorded in it by time stamps, and data section. The time stamps are classified based on the time of receipt of the request, the time required for its processing, and the time needed to send the file to the user. The data section consists of the identity of the data owner, their sensitivity, the purpose of the request, the identity, and the signature of the processing and consent node. The arrangement that defines the whole block structure is the locking time [34].

3.3.8. Side-Block Structure. A side block is a form that comes from attaching a section to a master block, producing a new block with its own identity. The block side consists of its size but also the header with the sections found in the main block, in particular, the version of the block that uniquely identifies the references used to create it, the fragmented value of the previous block, the Merkle tree [18, 35] root of all records, its time stamp creation, the target difficulty value, and the nonce value. The listed components have the similar properties as the parent blocks but are attached to the side blocks. Like the parent block, the side block also has a counter for malicious activity, which is recorded in a report. It not only consists of the time stamp of the action, the action itself, the identity of the data holder, and the identity of the user but also the identity and signature of the consent node. The block is "locked" in time and attached to the parent block of the blockchain. Traces of data and reference can now be traced.

3.3.9. Overlay Layer. This is an extra layer that was included in the blockchain stack layer to map the communication arrangement between the participants. Overlay nodes are an abstract logical path and can be thought of as associated with virtual links, each of which underlies the physical network topology. [22, 23].

3.3.10. API Layer. It is the application programming interface that allows external applications or users to interface with the blockchain. It allows the extraction of information from one system to another in a clear way.

The proposed blockchain-based verifiable user data access control policy mechanism is depicted in Figure 2.

4. Attacks Scenario

The scenario under consideration will focus on questions of finding closest neighbors and will demonstrate that the proposed system supports the secure processing of such questions under different intruder capabilities [36]. The k-nearest neighbors (k-NN) method is a critical parsing function of common data processing operations (e.g., classification or grouping) [24, 37]. Figure 3 shows the model for secure computing of encrypted databases [24].

In this model, the owner (user_1) of a database needs to execute some DB queries. To take advantage of a service provider's computing resources, it exports the database to an encrypted scheme (encrypted DBMS). Therefore, all blocks are encrypted by user_1 in order to proceed further to the encrypted DBMS. On the encrypted DBMS E(DB), all submitted queries by any user are also encrypted, resulting in an encrypted response R (e.g., R is an encrypted set of blocks of the answer to a k-NN query) [25, 38, 39]. All users must agree on a specific encryption system that ensures the integrity of the whole system. The proposed encryption model consists of the following elements: a secret key K, an ET() database encryption function, a set of Aux auxiliary operators, and a decryption function D() as a result [36].

In particular, the proposed encryption scheme requires that the encrypted queries and DB points should be encrypted differently ([ET ≠ EQ]). The graduated product of p and q (represented by the column vectors) can be represented as $p^{\mathrm{T}}Iq$, where p^{T} is the inverse of p and I is an

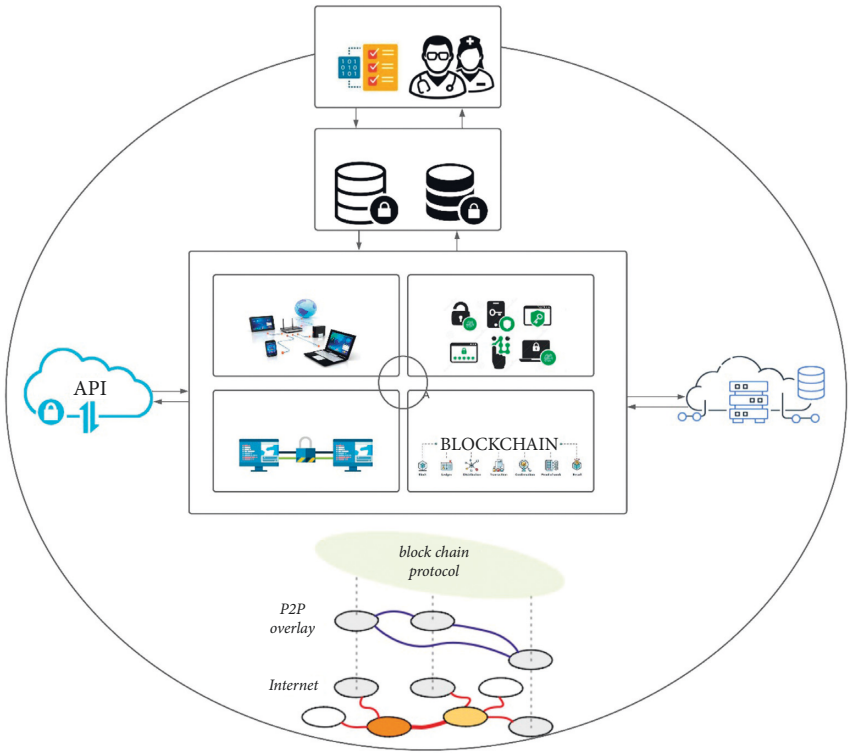


Figure 2: Proposed blockchain-based verifiable user data access control policy.

identity register $d \times d$. I can be replaced by MM^{-1} for any reversible register M, i.e., $p^{T}q = (p^{T}\alpha)\ (M^{-1}q)$. If we set $p' = E_{T}(p, K) = M_{T}p$ and, respectively, $q' = EQ(q, K) = M^{-1}q)$, it is not possible for one to determine the value of p or q, respectively, from p' or q' without knowing M. Also, $p'^{T}q' = p^{T}MM^{-1}q = p^{T}q$, i.e., the graduated product of type 2 is retained. If $p'1$ and $p'2$ are the encrypted points of $p1$ and $p2$ in DB, respectively, then $p'1^{T}\,p2' = p1^{T}\,MM^{T}\,p2$, which is not equal to $p1^{T}p2$ in general. Therefore, type 1 and 3 grade products are not retained. Thus, we can perform ASPE using M and M^{-1} as transformations function of DB points and queries, separately. Also, $p'^{T}q' = p^{T}MM^{-1}q = p^{T}q$, i.e., the grade 2 product is retained [24].

4.1. Attack Models. In the model, we assume that the encrypted DBMS, which may be in a third party (e.g., cloud service provider), is not secure. Therefore, we assume that an intruder (user_3-attacker) sees the encrypted DBMS environment [22, 25, 40]. Specifically, the attacker has access to the encrypted DBMS (encrypted queries, results, etc.) and in all components of the encryption system except from key (ET(), EQ(), D()), Aux, etc.).

We assume that the attacker's goal is to retrieve a portion of the DBA $\subseteq$ DB database and that he can perform cryptanalysis algorithms relative to the size of the encrypted database. In computational complexity theory, P, (nO(1)), is a fundamental order of complexity that contains all decision problems that can be solved by a deterministic turing machine in polynomial time. Our goal is to prevent the attacker from gaining part of the database. In addition to E(DB), the attacker may have additional information about the original data. To evaluate the encryption system, we will classify the attackers at different levels based on the knowledge they possess. Specifically, Level 1, the attacker only observes the encrypted database E(DB). Level 2, the attacker knows a set of simple blocks P in DB but does not know the corresponding encrypted values of these blocks in E(DB). Level 3, the attacker observes a set of P blocks in DB and knows the corresponding encrypted values of these blocks.

Among the three attack levels defined, we observe that level 2 attacks describe practical scenarios. This is because in some applications, it is not difficult to observe a small number of simple database blocks (e.g., by artificially inserting "spy" blocks into the database). In addition, it is considered that the attacker cannot observe the simple questions in all cases. In particular, the attacker is not allowed to pretend to be user_2 and query the database. Note that level 3 attacks are rare in practice, as it is not easy for someone who does not have the encryption key to associate known simple blocks with their encrypted values.

4.2. Queries of k-NN Neighbors in the Model. We will focus on questions from nearest k-NN neighbors and explain how the proposed encryption scheme (which includes the above five components) responds to the secure support of k-NN applications in the model. A k-NN query looks for k points in a database that are closest to a given query point q. Note that each database set can be modified as a multidimensional point if we consider some of its features as dimensions and their values as coordinates. One approach to securely supporting k-NN is the distance preserving transformation (DTP) for point encryption so that the distance between any

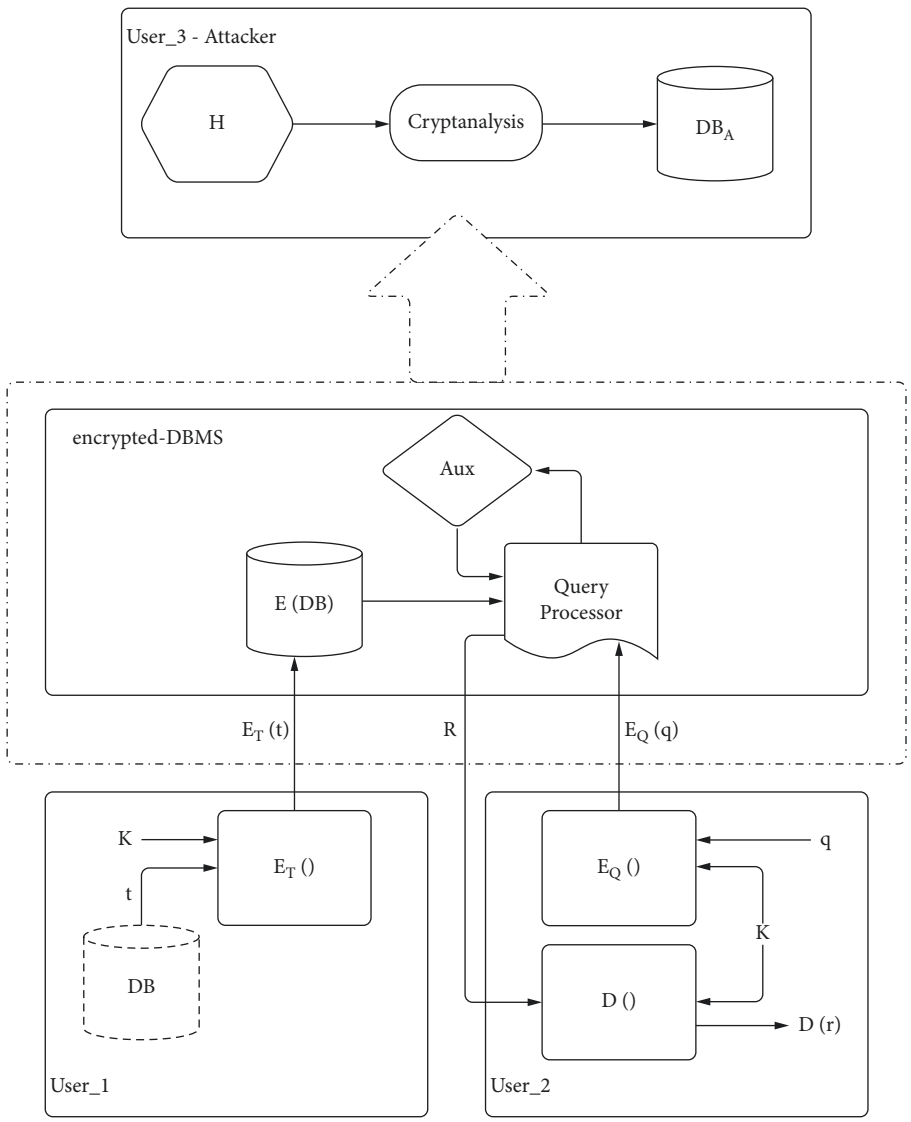


Figure 3: The proposed attack scenario.

two encrypted points in E(DB) is the same as that between the corresponding DB starting points. Given this property, k-NN can be computed in the encrypted database. Unfortunately, this transformation is proving to be unsafe in practice. If an attacker has access to the encrypted DPT database E(DB) and knows a few points in the plaintext DB database, he can fully recover the DB.

Similar k-NN query computing problems on an unreliable platform are studied services where users query an unreliable server that maintains the data. These applications focus on protecting users' privacy (query content) since the database is considered to belong to the server. While some studies also concern the privacy of files in the database, k-anonymity is adopted as a standard for database protection. We observe that k-anonymity has a different security goal compared to the proposed model. K-anonymity aims to prevent an attacker from locating a person from the database, but the contents of the database may be exposed. In addition, most of these models require the existence of a trusted intermediate (anonymous location) that handles the transformation of data and queries. This piece, in addition to being a single point of attack, undermines performance as every question and result must go through it.

4.3. Distant Recovery Encryption. In the k-NN calculation, the distances between the database points at a query point are calculated to find the nearest neighbors to the search point, so an encryption scheme that allows the system to calculate $d(p1, p2)$ at E(DB) for base points $p1$ and $p2$ at DB is not safe. However, the proposed encryption system is secure against level 2 attacks, as it does not allow distance calculation.

Distant Recovery Encryption (DRE). We have an $E(p, K)$ which is the encrypted value of a point p in DB. E is recoverable from a distance if and only if there is a computational procedure f for which for every p_1, p_2 K, $f(E(p_1, K)$, it

holds that $E(p_2, K)) = d(p_1, p_2)$. If E is DPT, we have $d(E(p_1, K), E(p_2, K)) = d(p_1, p_2)$. For a point p in DB represented as a column vector, the encrypted value $E(p, K)$ of p of a DPT E can be expressed as $Np + t$, where N is a rectangular register $d \times d$ and t is a two-dimensional column vector. The distance between the points is maintained, that is, $D(p_1, p_2) = d(E(p_1, K), E(p_2, K))$. Therefore, DPT supports efficient k-NN calculations. However, DRE and therefore DPT are secure and resilient in the proposed scheme.

Specifically, and assuming that a DRE E is used to encrypt the DB to get the E(DB), a level 3 attacker with $H = \langle E(\mathrm{DB}), P, I\rangle$ can retrieve DB if P contains at least $d+1$ points x_i $(1 \le i \le d+1)$ so that the set of vectors $\{x_j - x_1 \mid 2 \le j \le d+1\}$ is linearly independent. Therefore, although no DRE can survive this level 3 attack, this pattern survives as DHT uses a hash function to assign file ownership to network nodes which generate a 256-bit key k. Specifically, the proposed system uses an encryption function that does not reveal distance information as data of two points $p1$, $p2$ in DB, and it must be decided which of the two points is closest to a question point q, as well as

$$\sqrt{\|p1\|^2 - 2p1 * q + \|q\|^2} \ge \sqrt{\|p2\|^2 - 2p2 * q + \|q\|^2}, \tag{6}$$

where $||p||$ represents the Euclidean norm of p and $*$ represents the gradient system. $||p||2$ can be represented by $p * p$. Thus, inequality is subdivided into several calculations of gradients. This indicates a graded encryption of Espe product conservation, i.e., $\forall p1, p2 \in \mathrm{B}$, $p1 * p2 = \mathrm{Espe}(p1, K) * \mathrm{Espe}(p2, K)$, to calculate k-NN.

Even if the attacker manages to "upgrade" the knowledge of level 2 to level 3 using the "signature linking" attack, the proposed scheme is a guarantee and in particular, if at level 2, $H = \langle E(\mathrm{DB}), P\rangle$, the intruder constructs the signature of P from the distances per pair between every two points in P. Suppose the points in P are classified and $P = \{x_1, x_2, \ldots, x_{|P|}\}$. The signature of P, sig (P), is a vector of size $|P|C_2$ whose form is $(d(x_1, x_2), d(x_1, x_3), \ldots, d(x_1, x_{|P|}), d(x_{|P|}-1, x_{|P|}))$. The attacker tries to find a sorted set of encrypted points Q in E (DB) so that $|Q| = |P|$ and Q give the same signature as P. Let $Q = \{x_1', x_2', \ldots x_{|P|}'\}$. Sig (Q) is $(f(x_1', x_2'), f(x_1', x_3'), \ldots, f(x_1', x_{|P|}')), f(x_{|P|}' - 1, x_{|P|}'))$. If there is only one set Q with that signature, the attacker can conclude that x_i' is the encrypted $I(x_i) = x_i'$ for all $x_i \in P$. With this I, $H = \langle E(\mathrm{DB}), P, I\rangle$, and the attacker can carry out a level 3 attack. The success of the signature linking attack is based on two issues: if Q is simple to discover and in case is conceivable that another set Q' gives the same signature collision.

For the first question, we notice that the search space in the proposed shape is huge and cannot be effectively reduced by the "pruning" technique. For the second, we are able to appear that the likelihood of a signature conflict is extremely impossible. Moreover, indeed in case different Qs with the same signature as P are recognized, the attacker cannot increment the estimate of P to diminish the likelihood of a collision and rehash the attack as within the proposed design, the item maintenance encryption is not remotely retrievable which is given as follows:

$$f(p1', p2') = \sqrt{p1' * p1' - 2(p1' * p2')} + p2' * p2' \neq d(p1', p2'). \tag{7}$$

Therefore, the encryption function ET is not remotely retrievable as if the encryption E is remotely retrievable (i.e., E is DRE), then there is a computational procedure f such that for all points p_1 and p_2 and any encryption key K_1, it holds that $a1 = E(p_1, K_1)$ and $a_2 = E(p_2, K_1)$, and we have $f(a_1, a_2) = d(p_1, p_2)$. That is, considering the encrypted values a_1 and a_2, the distance $d(p_1, p_2)$ can be calculated from f, regardless of the encryption key.

5. Conclusions

The blockchain-based verifiable user data access control policy for secured big data storage in the cloud that was analyzed is based on the design of a data exchange network between systems that use cloud computing utilizing blockchain technology. The design includes the utilization of blockchain components to successfully distinguish the suspicious behavior and repudiate access with the implementation of the model, and the systems will be able to securely transmit, control, and detect data, while sharing medical data with other medical institutions and research institutes, without any risk to their privacy.

The variety of solutions offered and the costs involved are indicative of how difficult it is to secure a similar system in a hostile environment. It is reasonable to conclude that its securing requires specialized ways of assigning IDs to the nodes, dispersing the nodes, instant data copying, and an access mechanism that offers high possibilities of safeguarding security and privacy. In any case, despite the possibility of achieving a practically acceptable level of security in critical applications, it is obvious that a lot of research effort is still required as the requirements are high and constantly increasing.

Data Availability

Data are available on reasonable request to the author.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This study was supported by the Foundation of Hunan Educational Committee (Grant nos. 19C0533 and 20A144).

References

[1] K. Gai, J. Guo, L. Zhu, and S. Yu, "Blockchain meets cloud computing: a survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2009–2030, 2020.

[2] W. Li, M. He, and S. Haiquan, "An overview of blockchain technology: applications, challenges and future trends," in *Proceedings of the 2021 IEEE 11th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 31–39, Beijing, China, June 2021.

[3] S. Rouhani and R. Deters, "Blockchain based access control systems: state of the art and challenges," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 423–428, Thessaloniki Greece, October 2019.

[4] S. Cao, G. Zhang, P. Liu, X. Zhang, and F. Neri, "Cloud-assisted secure eHealth systems for tamper-proofing EHR via blockchain," *Information Sciences*, vol. 485, pp. 427–440, 2019.

[5] S. Lienkov, G. Zhyrov, I. Pampukha, and I. Chetverikov, "Block encryption algorithm for digital information using open keys for selfgeneration of closed random private keys," in *Proceedings of the 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT)*, pp. 200–203, Kyiv, Ukraine, December.2019.

[6] B. K. Mohanta, S. S. Panda, and D. Jena, "An overview of smart contract and use cases in blockchain technology," in *Proceedings of the 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1–4, Bengaluru, India, July 2018.

[7] V. Y. Kemmoe, W. Stone, J. Kim, D. Kim, and J. Son, "Recent advances in smart contracts: a technical overview and state of the art," *IEEE Access*, vol. 8, pp. 117782–117801, 2020.

[8] P. Chinnasamy, B. Vinodhini, V. Praveena, C. Vinothini, and B. Ben Sujitha, "Blockchain based access control and data sharing systems for smart devices," *Journal of Physics: Conference Series*, vol. 1767, no. 1, Article ID 012056, 2021.

[9] M. Macías and J. Guitart, "Trust-aware operation of providers in cloud markets," *Distributed Applications and Interoperable Systems*, pp. 31–37, Berlin, Heidelberg, 2014.

[10] U. Ugobame Uchibeke, K. A. Schneider, S. Hosseinzadeh Kassani, and R. Deters, "Blockchain access control ecosystem for big data security," in *Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 1373–1378, Halifax, NS, Canada, July 2018.

[11] F. Ghaffari, E. Bertin, J. Hatin, and N. Crespi, "Authentication and access control based on distributed ledger technology: a survey," in *Proceedings of the 2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, pp. 79–86, Paris, France, September 2020.

[12] S. Algarni, F. Eassa, K. Almarhabi et al., "Blockchain-based secured access control in an IoT system," *Applied Sciences*, vol. 11, no. 4, p. 1772, 2021.

[13] A. B. Dar, A. I. Baba, A. H. Lone, R. Naaz, and F. Wu, "Blockchain driven access control mechanisms, models and frameworks: a systematic literature," vol. 1379, 2020, https://eprint.iacr.org/2020/1379.

[14] H. Gao, Z. Ma, S. Luo, Y. Xu, and Z. Wu, "BSSPD: A Blockchain-Based Security Sharing Scheme for Personal Data with Fine-Grained Access Control," vol. 2021, Article ID 6658920, 2021.

[15] J. Wu, S. Guo, J. Li, and D. Zeng, "Big data meet green challenges: big data toward green applications," *IEEE Systems Journal*, vol. 10, no. 3, pp. 888–900, 2016.

[16] Y. Doi, S. Wakayama, and S. Ozaki, "A design for distributed backup and migration of distributed hash tables," in *Proceedings of the 2008 International Symposium on Applications and the Internet*, pp. 213–216, Turku, Finland, July 2008.

[17] I. Homoliak, S. Venugopalan, Q. Hum, and P. Szalachowski, "A security reference architecture for blockchains," in *Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain)*, pp. 390–397, Atlanta, GA, USA, July 2019.

[18] J.-F. Paris and T. Schwarz, "Merkle hash grids instead of Merkle trees," in *Proceedings of the 2020 28th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, pp. 1–8, Nice, France, November 2020.

[19] M. Nithya and N. U. Maheshwari, "Load rebalancing for Hadoop Distributed File System using distributed hash table," in *Proceedings of the 2017 International Conference on Intelligent Sustainable Systems (ICISS)*, pp. 939–943, Palladam, India, December 2017.

[20] V. S. Varanasi and S. Chilukuri, "Adaptive differentiated edge caching with machine learning for V2X communication," in *Proceedings of the 2019 11th International Conference on Communication Systems Networks (COMSNETS)*, pp. 481–484, Bengaluru, India, January 2019.

[21] R. Al-Aaridhi, A. Yueksektepe, and K. Graffi, "Access control for secure distributed data structures in Distributed Hash Tables," in *Proceedings of the 2017 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, pp. 1–3, Osaka, Japan, June 2017.

[22] O. Ahmedova, U. Mardiyev, and O. Tursunov, "Generation and distribution secret encryption keys with parameter," in *Proceedings of the 2020 International Conference on Information Science and Communications Technologies (ICISCT)*, pp. 1–4, Tashkent, Uzbekistan, November 2020.

[23] M. Islam, M. Shah, Z. Khan, T. Mahmood, and M. J. Khan, "A new symmetric key encryption algorithm using images as secret keys," in *Proceedings of the 2015 13th International Conference on Frontiers of Information Technology (FIT)*, pp. 1–5, Islamabad, Pakistan, December 2015.

[24] W. K. Wong, D. W.-l. Cheung, B. Kao, and N. Mamoulis, "Secure kNN computation on encrypted databases," in *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pp. 139–152, Rhode Island USA, June 2009.

[25] S. Y. Bonde and U. S. Bhadade, "Analysis of encryption algorithms (RSA, SRNN and 2 key pair) for information security," *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, in *Proceedings of the 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, pp. 1–5, Pune, India, August 2017.

[26] J. Ai, H. Huang, Y. Han, and Z. Wu, "Research on key management server key Re-encryption technology," in *Proceedings of the 2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, pp. 1899–1903, Chengdu, China, December 2018.

[27] B. H. Swathi, M. S. Meghana, and P. Lokamathe, "An analysis on blockchain consensus protocols for fault tolerance," in *Proceedings of the 2021 2nd International Conference for Emerging Technology (INCET)*, pp. 1–4, Belagavi, India, May 2021.

[28] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, "Blockchain-enabled smart contracts: architecture, applications, and future trends," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 11, pp. 2266–2277, 2019.

[29] G. B. Mermer, E. Zeydan, and S. S. Arslan, "An overview of blockchain technologies: principles, opportunities and challenges," in *Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, Izmir, Turkey, May 2018.

[30] H. Singh, S. Tyagi, and P. Kumar, "High availability and accessibility of services in cloud environment," in *Proceedings of the 2018 4th International Conference on Computing Sciences (ICCS)*, pp. 67–71, Jalandhar, India, August 2018.

[31] V. S. Batra, J. Bhattacharya, H. Chauhan, A. Gupta, M. Mohania, and U. Sharma, "Policy driven data administration," in *Proceedings of the Proceedings Third International Workshop on Policies for Distributed Systems and Networks*, pp. 220–223, Monterey, CA, USA, June 2002.

[32] X. Yao, X. Zhou, and J. Ma, "Differential privacy of big data: an overview," in *Proceedings of the 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)*, pp. 7–12, NY, USA, April 2016.

[33] V. S. Mahalle and A. K. Shahade, "Enhancing the data security in Cloud by implementing hybrid (Rsa & Aes) encryption algorithm," in *Proceedings of the 2014 International Conference on Power, Automation and Communication (INPAC)*, pp. 146–149, Amravati, India, October 2014.

[34] C. Cai, H. Duan, and C. Wang, "Tutorial: building secure and trustworthy blockchain applications," in *Proceedings of the 2018 IEEE Cybersecurity Development (SecDev)*, pp. 120-121, MA, USA, September 2018.

[35] W. Zhai, K. Qi, J. Duan, and C. Cheng, "Merkle quad-tree based remote sensing image analysis," in *Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 6193–6196, TX, USA, July 2017.

[36] Z. Fei, Z. Luo, Z. Liu et al., "Retracted article: analysis of the role of breast dynamic nuclear magnetic resonance imaging in the treatment of breast tumors," *Multimedia Tools and Applications*, vol. 80, no. 19, 30003 pages, 2021.

[37] L. Li, Y. Zhang, and Y. Zhao, "k-Nearest Neighbors for automated classification of celestial objects," *Science in China - Series G: Physics Mechanics and Astronomy*, vol. 51, no. 7, pp. 916–922, 2008.

[38] M. Nguyen, M. O. Gani, and V. Raychoudhury, "Yours truly? Survey on accessibility of our personal data in the connected world," in *Proceedings of the 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 292–297, Kyoto, Japan, March 2019.

[39] H. Zheng, T. Tran, and O. Arden, "Total Eclipse of the enclave: detecting Eclipse attacks from inside TEEs," in *Proceedings of the 2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 1–5, Sydney, Australia, May 2021.

[40] H. b. Jaafar, N. b. Mukahar, and D. A. Binti Ramli, "A methodology of nearest neighbor: design and comparison of biometric image database," in *Proceedings of the 2016 IEEE Student Conference on Research and Development (SCOReD)*, pp. 1–6, Kuala Lumpur, Malaysia, December 2016.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6044071, 7 pages
https://doi.org/10.1155/2022/6044071

Research Article

Machine Learning Methods to Detect Voltage Glitch Attacks on IoT/IIoT Infrastructures

Wei Jiang

Zhengzhou College of Finance and Economics, Zhengzhou 450000, China

Correspondence should be addressed to Wei Jiang; jiangwei198308@163.com

Received 10 February 2022; Revised 18 February 2022; Accepted 21 February 2022; Published 26 April 2022

Academic Editor: Konstantinos Demertzis

A majority of modern IoT/IIoT digital systems rely on cryptographic implementations to provide satisfactory levels of security. Hardware attacks such as side-channel analysis attacks or fault injection attacks can significantly degrade and even eliminate the desired level of security of the infrastructure in question. One of the most dangerous attacks of this type is voltage glitch attacks (VGAs), which can change the intended behavior of a system. By effectively manipulating the voltage at a specific time, an error can be injected that can change the intentional conduct and bypass system security features or even extract confidential information such as encryption keys by analyzing incorrect outputs of the firmware. This study proposes an innovative VGAs detection system based on advanced machine learning. Specifically, an innovative semisupervised learning methodology is used that utilizes a hybrid combination of algorithms. Specifically, a heuristic clustering method is used based on a linear fragmentation of group classes. In contrast, the ELM methodology is used as an algorithm for retrieving hidden variables through convex optimization.

1. Introduction

The Internet of Things (IIoT) is a network of networked sensors, instruments, and other devices that, when combined with industrial applications such as production and energy management, provide a complex system of services that allows for higher-level automation [1, 2]. Data collection, exchange, and analysis are substantially facilitated by this connectedness, which greatly aids performance improvement throughout the value chain. Physical systems such as sensors, actuators, control systems, security mechanisms, and other IIoT systems are frequently combined as a multi-layered digital technology architecture, where physical networking media (wired and wireless) protocols that collect and transfer information to the upper and lower layers of the communications layer are mentioned at the hardware level, while at the network level, physical networking media (wired and wireless) and protocols that obtain and send data to the upper and lower layers of the communications layer are mentioned [3].

Cyber-physical platforms [2, 4] are super-grid interactive computer and communication technologies that use feedback loops to monitor, coordinate, and control physical elements. Physical processes impact IoT computations and vice versa [5]. This solution combines the dynamics of physical processes with those of software and networking, resulting in abstract technical analysis and design models for a unified whole that is more akin to the intersection than the merger of the physical and digital worlds. Cyber-physical systems are a new generation of sophisticated capabilities that use information technology, communications, precise control, coordination, and autonomy to achieve physical association with the digital environment [3]. Understanding the standard components, the dynamics of information systems, hardware, software, networks, and the physical processes that model a scenario, as well as the relationships between them, is required for their design [6].

Industry 4.0 [3] defines cyber-physical workflow as an optimal combination of equipment and items, encompassing production facilities, storage mechanisms, enterprise resource planning, manufacturing execution system, outbound logistics, and service provisioning [7, 8]. They are integrated systems encompassing the production cycle and

storing and evaluating the generated data for industrial process modeling and analysis. Intelligent machines communicate via machine-to-machine (M2M) communication, performing controls on both sides and making decentralized judgments. The communication network and other intermediary elements are the interfaces that interact with the normal interfaces of the physical with the digital world [9–11].

As it is widely understood, the design of low power circuits is a critical operational factor of Industry 4.0, where devices such as interconnected sensors, actuators, and digital-analog signal converters are actively integrated into the IoT as autonomous mechanisms of the production process [3, 10, 12]. In low power combinational circuits, the dynamic power supply can receive a signal transition either as a functional or glitch. Before it reaches a steady state, a signal can go through many static changes called glitches. As glitches dissipate 20–70% of the total power consumption, they play a vital role in their operation, so it is necessary to control them thoroughly for the smooth operation of low power circuits [13].

The presence of hardware attacks such as side-channel analysis attacks and fault injection attacks can significantly degrade and even eliminate the desired level of security of low power circuits included in Industry 4.0 [5]. Such physical attacks are numerous and can be classified into two main categories as follows [13–15]:

(1) Invasive/noninvasive: invasive attacks necessitate interfering with the chip shell to gain direct access to the chip's interior components. Connecting a cable to a data bus to access data transfers is a good illustration of this. Noninvasive attacks, on the other hand, rely solely on externally available data (sometimes inadvertently emitted) such as operational time and power consumption.

(2) Active/passive: active attacks aim to stop equipment from functioning properly. Error-induced assaults, for example, will attempt to introduce computational errors. Passive assaults, on the other hand, will just watch the behavior of the devices throughout processing without interfering with it.

In general, the above is also referred to as implementation attacks. They include any effort that is dependent on information derived from an electronic system's implementation rather than flaws in the implemented algorithm itself (e.g., cryptanalysis and software implementation mistakes) [16, 17]. Timing information, power consumption, electromagnetic leakage, and even sound can all be used as supplementary data sources. Side-channel attacks, fault attacks, optical fault injection, electromagnetic fault injection, clock/voltage glitch, and other examples of this sort of attack vary depending on the medium utilized. [13, 15].

The most dangerous and difficult-to-detect type of attack is the VGAs [14, 15]. It is achieved at a physical level and interferes with the operation of the material by applying physical disturbances or changing environmental conditions, for example, using heavy-ion radiation and magnetic or electronic interference. These disturbances can cause the supply voltage to fluctuate (supply disturbances), introduce laser memory errors, or modify the input/output value of the circuit. Error input based on this type of attack may also include the addition of specially designed hardware to the system under evaluation, which allows the introduction of specific kinds of errors and the monitoring of costs to examine the effects of errors on system operation [13, 16, 18]. Depending on their mistake and location, VGAs fall into two categories as follows [13, 15, 19]:

(1) Contact fault input: direct physical contact with the target system, causing voltage or current disturbances in the target chip.

(2) Noncontact hardware error input: there is no direct physical contact with the target system. Instead, an external source produces a natural phenomenon like heavy-ion radiation or electromagnetic interference that causes the target chip to malfunction.

Dealing with the highly complex and undetectable attacks of hardware-related VGAs is an open problem in the research community, both in hardware development and digital security, as reflected in the international literature.

2. Literature Review

The massive increase in data flow across IoT sensors and, more importantly, in IIoT communication protocols has raised security concerns, emphasizing the significance of reliable approaches for promptly and accurately identifying threats. Security professionals and researchers rely on automated methods aided by deep learning to improve the efficacy of unwanted behavior detection, which is gaining popularity in the corporate world.

Sengupta [6] conducted a comprehensive review of IoT security concerns and countermeasures, with a focus on IIoT, and classified attacks based on the vulnerability object. This classification would make it easier for scholars to figure out which attacks are relevant to their particular field of study. Following that, each attack is mapped to one or more layers of the generic IoT/IIoT architecture, followed by a discussion of the available defenses. Researchers would also have a better understanding of the major security research concerns and their solutions in the field of IoT/IIoT by using a complete taxonomy. Finally, they present a case study on two critical industrial IoT applications.

Barenghi et al. [15] concentrated on fault injection attacks that did not have specific hardware or capabilities. They presented a detailed overview of these cryptographic device attacks and the solutions that have been devised to combat them. They compiled a list of attacks for the most important and widely used ciphers, stating which ones have been successfully implemented. They divided fault injection attacks into two categories as follows: low cost and high cost. They went over the protections, including intrusion detection and fault diagnosis, before examining the connection between fault injection and power analysis threats.

Vosoughi and Köse [16] advocated using the on-chip voltage regulator's existing resources as a countermeasure against VGA to improve their durability. They compared the number of phases in the multi-phase voltage regulator (MPVR) to the number of phases in the VGA. On a substitution box (S box) of an AES, they tested the efficiency of the proposed countermeasure. When compared to the unprotected S-box of an AES device, the faults induced by the VGA on the cryptographic circuit were reduced by 5.45% with a single-phase on-chip VR and by 91.82% with an MPVR with 32 phases, demonstrating the efficacy of their technique.

Bozzato et al. [13] introduced the voltage fault injection (V-FI) approach, which uses off-the-shelf and low-cost equipment to generate completely arbitrary voltage glitch waveforms. They looked into the possibility of automatically and unsupervised detecting a valid set of attack parameters, including the glitch waveform. The results revealed an increase in firmware extraction speed and a significant reduction in the number of injected bugs needed to accomplish the attack. They also demonstrated previously unknown firmware extraction attacks on six microcontrollers from three major brands, which targeted the bootloader interface and extracted the firmware from the internal protected flash memory. The most difficult attacks shown exploit numerous vulnerabilities and inject over one million flaws, relying primarily on the newly proposed technique's performance and repetition. They demonstrated that an attacker could employ voltage fault injection to defeat the safeguards supplied by the microcontrollers under test, even with low resources.

Software attacks targeting hardware vulnerabilities was a term used by Polychronou et al. [20] to describe a specific class of malicious attack vectors targeting IoT/IIoT devices (SATHV). These techniques are aimed at both the hardware flaws in system microarchitecture and the side-channel leakages they cause in the system, and they do not require physical access to the device. They also recommended security measures that might be used to prevent sensitive data from being extracted, malicious implant code from being implanted, and privileged code from being accessed. They attempted to educate designers on the negative consequences of attacks and detection measures outlined in the literature. They offered two tables based on the criteria that listed and classified the side effects and detection mechanisms. They believe that IoT/IIoT systems require more robust security solutions because, in addition to the ease of attacks, defenders do not realize which attack routes will be employed in advance, thus they must design and optimize numerous detection techniques at the same time.

For the first time in the literature, our work proposes a heuristic semisupervised learning method, which uses a simplified methodology for linear segmentation of groups classes. Using an extremely simple and fast ELM [21] recovers the hidden variables that lead to the problem's solution. It is important to note that most of the solutions proposed are well-defined techniques that include microprocessor-type solutions, special hardware, countermeasure technologies, etc., which are very difficult to impossible to be a widely accepted solution.

3. Materials and Methods

To detect VGAs, we first model the problem of clustering N data into P classes and the set of P classes. Every data x_i with $i \in N = \{1, 2, ..., N\}$ belongs to the space $R^{1\times D}$. We define table $X \in R^{N\times D}$ with lines x_i. Each sample x_i belongs to a class of P. We define the variable z_i with $i \in N$, which belongs to the space $\{0, 1\}^{1\times P}$ with $z_i\ 1P = 1$, that is, a binary variable of dimension P that takes the value one only at a position p if and only if the data belong to the class p. Similar to x_i, we define the variable $Z \in R^{N\times P}$ with lines z_i and the set of index tables $Z_{N,P} = \{Z \in \{0, 1\}^{N\times P} | Z \cdot 1_P = 1_N\}$. This variable is a latent variable as we do not have access to the ground truth of the data. The purpose is to retrieve the values of the hidden variable and at the same time to train the ELM classifier h: $R^D \longrightarrow Z_{1,P}$, which will accept a characteristic vector data of dimension D as input and will return the index vector of the class to which the data belong. We can choose the classifier as follows [10, 11, 13, 21]:

$$\begin{aligned} f(x) &= \left(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_P(\mathbf{x})\right), \\ h(x) &= \left(h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_P(\mathbf{x})\right), \\ h_j(x) &= \begin{cases} 1, & j = \arg\max_i f_i(\mathbf{x}), \\ 0, & \text{elsewhere}, \end{cases} \end{aligned} \tag{1}$$

where f: $R^D \longrightarrow R^P$.

Extending the equation to the problem of unknown classes, the objective function is also minimized for z_i as follows [22–24]:

$$\min_{Z,f} \frac{1}{N} \sum_{i\in\mathcal{N}} \ell\left(\mathbf{z}_i, f\left(\mathbf{x}_i\right)\right) + \lambda\Omega(f). \tag{2}$$

Let us consider that the data are displayed in a space where the classes are linearly separable (the partition surfaces for each pair of classes are superficial). The function f can take the following form:

$$f(x) = xw + b, \quad w \in \mathbb{R}^{D\times P}, b \in \mathbb{R}^{1\times P}. \tag{3}$$

Finally, if we define the function as the square error and the normalization term as the L_2 norm of w, then the problem takes the following form:

$$\min_{Z,w,b} \frac{1}{2N} Z - Xw - 1_N b_F^2 + \frac{\lambda}{2} \mathrm{Tr}\left(w^T w\right). \tag{4}$$

Holding the Z as constant, we can find the minimum value of the function on w and b in closed form. To find the coefficients in this way, the ELM methodology is used as an algorithm for retrieving hidden variables through the solution of a convex program [21, 25].

ELMs are feedforward single hidden-layer feedforward neural networks (SLFNs). Given N random discrete observations $\{(x_i, t_i)\}$ for $i = 1$ as N, where $x_i \in R$ n with $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]\ T$ and $t_i \in R$ m with $t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]\ T$, an ELM with hidden nodes (neurons) K and activation function $g(x)$ is mathematically modeled with the following formula [21, 22]:

$$f\left(x_j; w, b, \beta\right) = \sum_{i=1}^{K} \beta_i * g\left(w_i * x_j + b_i\right) = o_j, j = 1, 2, \ldots, N, \tag{5}$$

where the variable $w_i = [w_{i1}, w_{i2}, \ldots, w_{in}]$ T is the vector of weights that connects the node i of the hidden plane with the nodes of the input plane, $\beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]$ T is the vector of weights that connects the node i of the hidden level with the nodes of the output layer, and b_i is the threshold of the hidden node i. A typical SLFN with hidden nodes K and activation function g(x) can approach N random observations with zero mean error value [21]:

$$\sum_{j=1}^{K} \left\| o_j - t_j \right\| = 0. \tag{6}$$

Therefore, there are β_i, b_i, and w_i such that

$$\begin{aligned} f\left(x_j; w, b, \beta\right) &= \sum_{i=1}^{K} \beta_i * g\left(w_i * x_j + b_i\right) \\ &= t_j,\ j = 1, 2, \ldots, N. \end{aligned} \tag{7}$$

For a given SLFN, there are N such equations (as many nodes of the hidden layer) that can be written as follows [26]:

$$H\beta = T, \tag{8}$$

where the array **H** is the output of the hidden layer.

$$H_{NxK} = \begin{bmatrix} g\left(w_1 * x_1 + b_1\right) & \cdots & g\left(w_K * x_1 + b_K\right) \\ \vdots & \ddots & \vdots \\ g\left(w_1 * x_N + b_1\right) & \cdots & g\left(w_K * x_N + b_K\right) \end{bmatrix}. \tag{9}$$

Table β symbolizes the table of output weights:

$$\beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdots \\ \beta_K \end{bmatrix}. \tag{10}$$

And, **T** is the table of the desired output values:

$$T = \begin{bmatrix} t_1 \\ t_2 \\ \cdots \\ t_m \end{bmatrix}. \tag{11}$$

The training process aims to find values for the variables w_i, b_i, and β_i for $i = 1,2, \ldots, K$ for which it applies [21, 21]:

$$\left\| H * \widehat{\beta} - T \right\| = \min_{w_i, b_i, \beta_L} \left\| H * \beta - T \right\|, \tag{12}$$

which corresponds to minimizing the cost function.

$$E = \sum_{j=1}^{N} \left(\sum_{i=1}^{K} \beta_i * g\left(w_i * x_j + b_i\right) - t_j \right)^2. \tag{13}$$

According to the backpropagation algorithm, a gradient descent algorithm is used to find the value:

$$\min_{w_i, b_i, \beta_t} \left\| H * \beta - T \right\|. \tag{14}$$

In the minimization process, the vector **W**, which is the sum of the weights (w_i, b_i) and the biases (β_i), is adjusted iteratively according to the following relation [26]:

$$W_k = W_{k-1} - n \frac{\partial E(W)}{\partial W}, \tag{15}$$

where n is the learning rate of the neural network. We used an easy-to-use, simple, and fast ELM as an algorithm for retrieving hidden variables in problem-solving. This heuristic methodology performs a linear fragmentation of class groups semiautomatically [21].

4. Experiments

To implement the scenario of the use of the proposed algorithm, the exact ways and the main factors that contribute to the energy consumption in the combined microcircuit circuits were studied. While the inputs of a combination circuit are excited by flip-flops, the internal gates of the circuit may need several shifts until they reach a steady state. These extra transitions are called glitches [27, 28]. Although not anticipated by designers, they are not necessarily design errors in terms of logical behavior. Still, they are a big problem in terms of digital security due to the fact that extra transitions consume energy. This form of energy is also known as glitch power and is quite tricky to calculate accurately [27, 29]. All experiments were conducted in the Google Colab no-GPU environment.

The percentage of the total energy that can come from glitches, which can be legitimately based on the circuit design and illegal due to VGAs, is quite large and difficult to calculate accurately. Since a percentage of the total power consumption diffuses into a circuit due to glitches, the tools for estimating the total power must be accurate in the presence of this phenomenon. This can be done electrically but only for medium-sized circuits. On the other hand, reasonable accuracy has not yet been achieved in detail. A distinctive feature of static circuits is that the total power consumption is mainly caused by signal switching. Therefore, logic gateway-level simulation algorithms calculate the average power dissipated by monitoring the activity (e.g., number of transitions) of a gateway output using the following relation [13, 16, 30]:

$$P_{\text{avg}} = f \frac{V_{DD}^2}{2} \sum_{i}^{n} C_{L_i} a_i, \tag{16}$$

where f is the clock frequency and n is the number of gates. At the same time, C_{Li} and a_i are the output capacity and the number of gate transitions of gate i during the period under consideration, respectively. It is important to note that the above relation does not consider the power consumed by the internal capacitors and by the short-circuit currents. The total power consumption of a circuit consists mainly of dynamic power consumption and static power consumption, which include other components respectively, as shown in the following equation [17, 31, 32]:

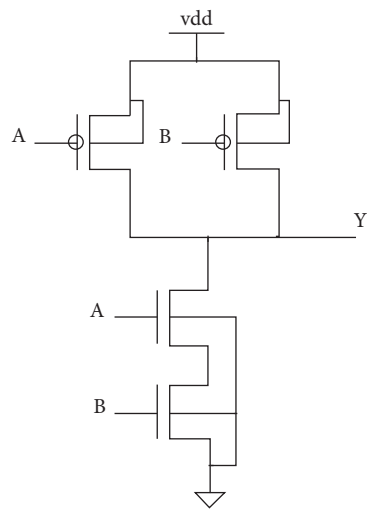


Figure 1: NAND gate (2 inputs).

Table 1: Performance of the proposed method.

Time slots	F-1 score (average)	Precision (average)	Recall (average)	Accuracy (average)
T1	81.00	80.90	80.90	80.90
T2	83.40	84.00	84.00	83.80
T3	88.80	88.90	88.90	88.80
T4	82.60	82.90	82.80	82.70
T5	89.90	88.90	89.00	89.00
T6	86.80	86.90	87.00	86.90
T7	90.20	90.30	90.30	90.30
T8	88.00	88.20	88.20	88.00
T9	89.10	89.10	89.00	89.20
T10	90.70	90.70	90.70	90.70
T11	83.90	83.90	83.80	83.90
T12	89.70	89.60	89.60	89.80
Average	87.00	87.00	87.00	87.00

$$\overline{P} = f_{\text{clk}} \int_0^{T_{\text{clk}}} Vdd^* I_{\text{supply}} \mathrm{d}t. \tag{17}$$

The input signals of a gateway are varied in such a way as to produce a value at the output of the gateway. However, depending on the time at which the signal changes take place, there is a possibility that an additional output value will be generated, resulting in a static glitch [27, 29, 30].

In the present work, a simulation was created that deals with the analysis and study of glitches made in the logical NAND 2 input gate designed at 1.2 μm and with a supply voltage of 1.1 V. There are two ways that a glitch can appear on this portal. The first is to create the glitch in this gate, which is done by the appearance of two transitions at its entrances with very close arrival times and the logical behavior of the gate to lead to it. The second is by propagation through the gate, wherein in this case, a glitch reaches the entrance of a gate and causes a similar situation at the exit node. Creating a glitch on a node spread to the following logical levels until logical or electrical masking can neutralize it [13, 16, 31].

The 2-input NAND gateway and its schematic simulation to collect glitches used to evaluate the proposed system are shown in Figure 1.

To create a glitch, we need to perform the transition $\text{CD} = 01 \longrightarrow 10$ and the transition $\text{CD} = 10 \longrightarrow 01$. We need two transitions of the input signals of the NAND 2 gate from $0 \longrightarrow 1$ and $1 \longrightarrow 0$. Creating a glitch at a node in the circuit begins to spread to the following logical levels until logical or electrical masking can neutralize it. More specifically, the glitches study area has two boundaries [29, 31]:

(1) The start time of the transition of one signal is equal to the end time of the transition of the other signal.

(2) The start time of the transition of the other signal should not exceed the end time of the transition of the first signal, that is, it should always be $t_1 < t_2$.

Great attention was paid to this study, so that the analysis is done each time before the procedure begins to avoid the breakdown of areas where glitches cannot occur. A total of 8,890

glitches were generated randomly distributed over a 12-hour time horizon.

Table 1 lists the success rates achieved by the proposed semisupervised algorithm. The values were calculated as the average of the metrics for each time slot, in which the glitches were randomly distributed.

The results are considered satisfactory given the complexity of the problem and the nonuniform classes that indicate the glitches detection problem. In general, the finding is that the proposed system can reliably evaluate and categorize the current anomalies associated with VGAs.

5. Discussion and Conclusions

Hardware attacks such as VGAs are among the most important modern attacks on IoT/IIoT devices [20]. Features such as the predicted behavior of the device can be changed, or even secret information such as encryption keys can be changed intercepted [33]. Given the growing complexity, ever-changing distributed industrial environment combined with the weakness of traditional systems, which in most cases fails to adapt to modern challenges, it is necessary to use alternative and more effective methods to protect industrial infrastructures [4, 7].

This study proposes an innovative VGAs detection system based on advanced machine learning. Specifically, an innovative semisupervised learning methodology is used, which utilizes a hybrid combination of algorithms [34]. It is an innovative heuristic nonaccelerated learning method for fragmenting VGAs problem-class groups. At the same time, an ELM is used as an algorithm to retrieve hidden variables for optimal problem-solving. The proposed methodology has serious advantages over other types of learning. Their main advantage, and the reason that makes it an ideal method for predicting short-term trend shifts, is to avoid using the time-consuming, repetitive backpropagation algorithm [35]. The proposed system uses unsupervised learning to determine the unknown distribution of data. At the same time, ELM is limited to a multiplication of tables, which reduces by almost 75% the time required to complete the classification. Also, avoiding the use of retrospective techniques such as backpropagation contributes to the nonappearance of local minima during the model's training, which affects the model's accuracy.

The evaluation of the system was carried out in an innovative data set created based on a highly complex and original scenario related to the operation of IoT/IIoT [36]. The results obtained are very encouraging and reflect the usefulness and effectiveness of machine learning systems in solving complex problems.

Future extensions of this research work should first focus on optimizing the model's hyperparameters to improve the performance and generalization it can achieve significantly. It is also imperative to make a thorough comparison between classical and modern machine learning architectures to understand the predictive power of the proposed method. Finally, self-determination methods should be explored to make the system autonomous.

Data Availability

Data are available on reasonable request to the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] S. Datta, R. A. G. Antonio, A. R. S. Ison, and J. M. Rabaey, "A programmable hyper-dimensional processor architecture for human-centric IoT," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 3, pp. 439–452, 2019.

[2] N. Jazdi, "Cyber Physical Systems in the Context of Industry 4.0," in *Proceedings of the 2014 IEEE International Conference on Automation, Quality and Testing, Robotics*, pp. 1–4, Cluj-Napoca, Romania, May 2014.

[3] A. Ustundag and E. Cevikcan, *Industry 4.0: Managing the Digital Transformation*, Springer International Publishing, Manhattan, NY, USA, 2018.

[4] A. G. Kravets, A. A. Bolshakov, and M. V. Shcherbakov, *Cyber-Physical Systems: Industry 4.0 Challenges*, Springer International Publishing, vol. 260, Manhattan, NY, USA, , 2020.

[5] P. Radanliev, D. D. Roure, K. Page et al., "Cyber Risk at the Edge: Current and Future Trends on Cyber Risk Analytics and Artificial Intelligence in the Industrial Internet of Things and Industry 4.0 Supply Chains," Dec. 2020, https://www.preprints.org/manuscript/201903.0123/v2.

[6] J. Sengupta, "A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT," *Journal of Network and Computer Applications*, vol. 20, 2020.

[7] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, "Cyber threats to industrial IoT: a survey on attacks and countermeasures," *IoT*, vol. 2, no. 1, pp. 163–186, 2021.

[8] E. Vasilomanolakis, J. Daubert, M. Luthra, V. Gazis, A. Wiesmaier, and P. Kikiras, "On the security and privacy of Internet of Things architectures and systems," in *Proceedings of the 2015 International Workshop on Secure Internet of Things (SIoT)*, pp. 49–57, Vienna, Austria, September 2015.

[9] N. Velásquez Villagrán, P. Pesado, and E. Estevez, "Cloud Robotics for Industry 4.0 - A Literature Review," in *Cloud Computing, Big Data & Emerging Topics*, pp. 3–15, Springer International Publishing, Manhattan, NY, USA, 2020.

[10] L. Hou, Y. Zhang, Y. Yu, Y. Shi, and K. Liang, "Overview of data mining and visual analytics towards big data in smart grid," in *Proceedings of the 2016 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI)*, pp. 453–456, Beijing, China, October 2016.

[11] A. Cuzzocrea, "Big data lakes: models, frameworks, and techniques," in *Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1–4, Jeju Island, Korea (South), January 2021.

[12] A. I. Khan and A. Al-Badi, "Ubiquitous application testing on cloud," in *Proceedings of the 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, pp. 1–4, Shah Alam, Malaysia, July 2018.

[13] C. Bozzato, R. Focardi, and F. Palmarini, "Shaping the glitch: optimizing voltage fault injection attacks," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2019, pp. 199–224, 2019.

[14] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan, "The sorcerer's apprentice guide to Fault Attacks," *Proceedings of the IEEE*, vol. 94, no. 2, pp. 370–382, 2006.

[15] A. Barenghi, L. Breveglieri, I. Koren, and D. Naccache, "fault injection attacks on cryptographic devices: theory, practice, and countermeasures," *Proceedings of the IEEE*, vol. 100, no. 11, pp. 3056–3076, 2012.

[16] A. Vosoughi and S. Köse, "Leveraging on-chip voltage regulators against fault injection attacks," in *Proceedings of the 2019 on Great Lakes Symposium on VLSI*, pp. 15–20, New York, NY, USA, May 2019.

[17] B. Zhou and Z. Liu, "Method of multi-resolution and effective singular value decomposition in under-determined blind source separation and its application to the fault diagnosis of roller bearing," in *Proceedings of the 2015 11th International Conference on Computational Intelligence and Security (CIS)*, pp. 462–465, Shenzhen, China, Dec. 2015.

[18] J. M. Grossman, S. Aubin, E. Gomez et al., "New apparatus for magneto-optical trapping of francium," in *Proceedings of the Technical Digest. Summaries of Papers Presented at the Quantum Electronics and Laser Science Conference. Postconference Technical Digest (IEEE Cat. No.01CH37172)*, p. 220, Baltimore, MD, USA, May 2001.

[19] K. T. Chitty-Venkata and A. Somani, "Impact of structural faults on neural network performance," in *Proceedings of the 2019 IEEE 30th International Conference on Application-specific Systems, Architectures and Processors (ASAP) (ASAP)*, vol. 2160–052X, p. 35, New York, NY, USA, July 2019.

[20] N.-F. Polychronou, P.-H. Thevenon, M. Puys, and V. Beroulle, "A comprehensive survey of attacks without physical access targeting hardware vulnerabilities in IoT/IIoT devices, and their detection mechanisms," *ACM Transactions on Design Automation of Electronic Systems*, vol. 27, no. 1, pp. 1–35, 2022.

[21] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, Dec. 2006.

[22] T. F. de Lima, H.-T. Peng, A. N. Tait et al., "Machine learning with neuromorphic photonics," *Journal of Lightwave Technology*, vol. 37, no. 5, pp. 1515–1534, 2019.

[23] D. Belforte, "Overview of the laser machining industry," in *Procedings of the Technical Digest. Summaries of Papers Presented at the Conference on Lasers and Electro-Optics. Postconference Edition. CLEO '99. Conference on Lasers and Electro-Optics (IEEE Cat. No.99CH37013)*, p. 82, Baltimore, MD, USA, May 1999.

[24] N. Elmrabit, F. Zhou, F. Li, and H. Zhou, "Evaluation of machine learning algorithms for anomaly detection," in *Proceedings of the 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1–8, Dublin, Ireland, June 2020.

[25] C. Barrera-Singana, A. Valenzuela, and M. P. Comech, "Dynamic control modelling of a bipole converter station in a multi-terminal HVDC grid," in *Proceedings of the 2017 International Conference On Information Systems And Computer Science (INCISCOS)*, pp. 146–151, Quito, Ecuador, November 2017.

[26] B. Deng, X. Zhang, W. Gong, and D. Shang, "An overview of extreme learning machine," in *Proceedings of the 2019 4th International Conference on Control, Robotics and Cybernetics (CRC*, pp. 189–195, Tokyo, Japan, September 2019.

[27] H. Martin, T. Korak, E. S. Millan, and M. Hutter, "Fault Attacks on STRNGs: impact of glitches, temperature, and underpowering on randomness," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 266–277, 2015.

[28] S. Bawell, "A hybrid-coded architecture for glitch-free gain control," in *Proceedings of the 2016 IEEE MTT-S International Microwave Symposium (IMS)*, pp. 1–4, San Francisco, CA, USA, Febura. 2016.

[29] M. Kasim, V. Gupta, and M. Jebin, "Methodology for Detecting Glitch on Clock, Reset and CDC Path," in *Proceedings of the 2020 5th International Conference on Communication and Electronics Systems (ICCES)*, pp. 300–304, Coimbatore, India, June 2020.

[30] C. Spensky, A. Machiry, N. Burow et al., "Glitching demystified: analyzing control-flow-based glitching attacks and defenses," in *Proceedings of the 2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 400–412, Taipei, Taiwan, June 2021.

[31] M. Slimani, P. Matherat, and Y. Mathieu, "A dual threshold voltage technique for glitch minimization," in *Proceedings of the 2012 19th IEEE International Conference on Electronics, Circuits, and Systems (ICECS 2012)*, pp. 444–447, Seville, Spain, September 2012.

[32] J. Obermaier, R. Specht, and G. Sigl, "Fuzzy-glitch: a practical ring oscillator based clock glitch attack," in *Proceedings of the 2017 International Conference on Applied Electronics (AE)*, pp. 1–6, Pilsen, Czech Republic, September 2017.

[33] B. Bordel, R. Alcarria, and T. Robles, "Lightweight encryption for short-range wireless biometric authentication systems in Industry 4.0," *Integrated Computer-Aided Engineering*, pp. 1–21, 2021.

[34] K. Al Jallad, M. Aljnidi, and M. S. Desouki, "Anomaly detection optimization using big data and deep learning to reduce false-positive," *Journal of Big Data*, vol. 7, no. 1, p. 68, 2020.

[35] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.

[36] C. Song and X. Wu, "Smart city + IoT standardization application practice model and realization of key technologies," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–11, 2022.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3150626, 9 pages
https://doi.org/10.1155/2022/3150626

Research Article

An Advanced Deep Attention Collaborative Mechanism for Secure Educational Email Services

Yanfang Chen and Yongzhao Yang

Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Yanfang Chen; chenyanfang0822@163.com

Received 3 March 2022; Revised 18 March 2022; Accepted 12 April 2022; Published 26 April 2022

Academic Editor: Konstantinos Demertzis

The COVID-19 crisis has once again highlighted the vulnerabilities of some critical areas in cyberspace, especially in the field of education, as distance learning and social distance have increased their dependence on digital technologies and connectivity. Many recent cyberattacks on e-learning systems, educational content services, and trainee management systems have created severe demands for specialized technological solutions to protect the security of modern training methods. Email is one of the most critical technologies of educational organizations that are attacked daily by spam, phishing campaigns, and all kinds of malicious programs. Considering the efforts made by the global research community to ensure educational processes, this study presents an advanced deep attention collaborative filter for secure academic email services. It is a specialized application of intelligent techniques that, for the first time, examines and models the problem of spam as a system of graphs where collaborative referral systems undertake the processing and analysis of direct and indirect social information to detect and categorize spam emails. In this study, nonnegative matrix factorization (NMF) is applied to the social graph adjacent table to place users in one (or more) overlapping communities. Also, using a deep attention mechanism, it becomes personalized for each user. At the same time, with the introduction of exponential random graph models (ERGMs) in the process of factorization, local dependencies are significantly mitigated to achieve the revelation of malicious communities. This methodology is being tested successfully in implementing mail protection systems for educational organizations. According to the findings, the proposed algorithm outperforms all other compared algorithms in every metric tested.

1. Introduction

During the coronavirus pandemic, the rapid and violent digital transformation is called upon to deal with a massive wave of sophisticated and persistent cyberattacks related to the new reality [1–4]. This new reality is becoming particularly evident in education, which has become one of the ideal targets for digital attacks, as distance learning has become a necessity for billions of students worldwide [5].

In particular, advanced phishing campaigns are constantly evolving, using e-learning and distance education, access to education services, and educational content management systems as a theme. Schools and universities have switched to large-scale e-learning platforms, often without knowing crucial privacy issues [6]. Students in distance learning programs are also attacked daily. Furthermore, the rate of ransomware attacks is growing exponentially, with specialized cybercriminals first extracting large amounts of sensitive personal data before encrypting the educational databases of an institution or training organization. They then threaten to publish these data unless a ransom is paid, putting additional pressure on the organizations called upon to meet the criminals' demands in question [7].

The recipients are targeted by phishing attacks, which imitate the login portals of universities to steal credentials, etc. In most cases, these scams are related to the email of educational organizations, where the fundamental security flaws can be easily bypassed. In most phishing campaigns targeting educational organizations, the attack begins with

an email that supposedly contains information about the institutions' instructions on complying with the COVID-19 protocols set by the relevant ministries, with instructions for course changes, grading, hypothetical links to digital classrooms, etc. Misleading spam emails encourage the recipient to click on an attached HTML file, which leads them to a fake login page similar to the teacher's login page, with disastrous results. These pages look very convincing, and the URLs use a similar naming pattern that includes the top-level domain of the relevant educational organizations [8, 9].

The most common and widely used machine learning technique for protection against related threats and the intelligent classification of spam emails is based on the use of measures that are based on distance. These methods are considered supervised learning methods, which presuppose that the whole set includes the input data and the desired categorization for each element [10, 11].

To categorize each new element in a class, it is necessary to calculate its distance from each part of the training set, finally considering only the assignments closest to it. These methods are based on analogy and not on producing a generalization model. Thus, there is no training stage, and no model is created until a new observation needs to be categorized. For this reason, the relevant categorizers are also called lazy classifiers [12]. Also, the specific methods for classifying a new observation must compare it with available observations of the training set, which requires storing all or at least part of the training data.

Although these methods can significantly achieve the complexity of detecting complex dependencies between the variables that make up the problem, they are relatively simple to implement and use and can achieve high classification performance [12, 13]. Many comparisons between observations require very effective indexing techniques. Their categorization of new observations takes longer and requires high availability of computing resources. Furthermore, the classification results are sensitive to the local characteristics of the data, the existence of insignificant input variables, and the number of categorization observations, increasing the risk of over-adaptation. We propose an innovative methodology to use a personalized attention mechanism to overcome the relevant obstacles.

The study is organized as follows: Section 2 provides an overview of the various relevant approaches that have been identified in the literature. Section 3 presents the proposed methodology. The scenarios and results are presented in Section 4. Finally, a summary of the findings and a list of potential future research directions are concluded in Section 5.

2. Literature Review

The recent literature concerning the field of detection mechanisms for email services is as follows.

Abdullahi and Kaya [14] suggested a deep learning system for detecting phishing in emails and messages. On both email and SMS collections, they utilized ML algorithms such as SVM classifier, multinomial naive Bayes, decision tree, random forest, logistic regression, and dense neural network. They used existing assessment criteria to evaluate the classifiers on the datasets they employed. The analysis was carried out using coding techniques and the TensorFlow technology, and the outcome revealed that dense neural networks outperform deep learning classifications in identifying phishing attempts in all the samples. The suggested strategy outperformed traditional machine learning algorithms on real datasets.

Fang et al. [15] employed an algorithm dubbed THEMIS to identify suspicious emails. They began by analyzing the email layout and then recommended their scheme, which was used to model emails at the email header, body, character, and word level simultaneously, using an improved recurrent convolutional neural network (RCNN) model with multilevel vectors and attention mechanism. This approach was used in both the header and the model's body, causing it to pay greater attention to the more critical information between them. They utilized an imbalanced dataset with actual ratios of phishing and genuine emails to perform tests and assess THEMIS achieving overall accuracy high levels, according to the testing data. In the meanwhile, the false-positive rate was low. The filter's accuracy and low FPR help detect phishing emails with high likelihood, while benevolent emails were filtered out as little as possible. They aim to improve their model for identifying phishing emails that do not have a header and simply have content.

Phomkeona and Okamura [16] proposed a method for categorizing and diagnosing zero-day malicious emails based on data gathered from the email header and content. They merged it with dynamic analytic data as a collection of 27 features, including machine translation detection, risk word detection, and other characteristics, using numerous application programming interfaces. To teach and evaluate the system, four distinct language email datasets were employed to replicate real-world diversity and zero-day harmful email attack scenarios. They achieved a reasonable detection rate for both zero-day malicious email types and regular spam. They stated that by adding new contaminated spam datasets to train the algorithm and utilizing a translation API to boost accuracy, their model could be improved.

Kaddoura et al. [17] applied deep learning methods to identify link-less emails, and they proposed a spam email detection system based on FFNN. Different settings were used to optimize hyperparameters. The Enron dataset was preprocessed, and two feature extraction algorithms were used. On the Enron dataset, their model was tested to classify emails as spam or regular. This approach is compared to the BERT dataset. They also looked at numerous variants of its design in terms of layer count, the number of neurons per layer, and the number of neurons per input layer and calculated the $F1$ score for each one. Precision, recall, and accuracy were calculated to illustrate the approach's success.

Asudani et al. [18] investigated the efficiency of a pretrained embedding model for email categorization using deep learning filters such as the long short-term memory and convolutional neural network models. They employed pretrained word embedding using global vectors (GloVe) and Bidirectional Encoder Representations Transformers (BERT) to discover links between words, which helped them

categorize messages into relevant categories using machine learning models. They concluded that word embedding models boost the accuracy of the email classifier. The experimentation used two benchmark datasets: SpamAssassin and Enron. GloVe embedding achieved quicker execution and improved performance on massive datasets, according to the research. Traditional machine learning methods categorize an email as benevolent or spam, and the CNN model with GloVe embedding yields somewhat greater accuracy than the model with BERT embedding.

Based on imperative phrases, Ali [19] proposed a framework for categorizing email content into three categories: order/command, request, and general. For email categorization, this study employed Word2Vec to convert words into vectors and two deep learning algorithms, namely convolutional neural networks and recurrent neural networks. They experimented using a sufficient email data collection obtained from a personal Gmail account and Enron. A random 10% of the dataset is used to test the machine, while 90% of the dataset is used for training the machine. Increases in the training dataset ratio result in enhanced algorithm accuracy, according to these trials. The experimental results reveal that RNN outperformed CNN in terms of accuracy. They also compared their approaches to the previously used method fuzzy ANN and found that their suggested methods CNN and RNN outperformed fuzzy ANN. They want to test the model on larger datasets since they think that combining many models and utilizing a hybrid method will increase accuracy.

In the research presented so far, the identification of spam and utilizing machine learning do not include the social information extracted from collaborative filtering algorithms, neither around users nor around objects. However, social information is perhaps the main springboard for the early suppression of spam and its scams. It is imperative to integrate it into forecasting systems. One of them is to find user communities based on the links (e.g., friendship and trust) they have in the social network. These communities can then be used to generate suggestions—predictions, which will reveal how to spread the unwanted content and, respectively, to identify potentially infected nodes that are bots in botnets. In general, the placement of users in "neighborhoods" on the social network is based on NMF techniques, where each user is not considered to belong exclusively to one neighborhood.

In the same way that a user does not rate items from a single category, so his social contacts do not fall into a single class (e.g., some may be coworkers, some friends, and some may belong in both groups mentioned above). On the contrary, they may belong to more, with a different percentage of participation in each. Therefore, the discovery and exploitation of these distinct groupings are expected to improve the production of recommendations—forecasts [20, 21].

NMF will be applied to the social graph adjacent table to place users in one (or more) overlapping communities in this research. Also, the NMF will be personalized for each user examined and will relate to that part of the social network that corresponds to the user's neighbors in question [20, 22, 23]. The most important and original feature of the proposed methodology is the introduction of ERGMs in the factorization process to mitigate the dominant local logic of the NMF, which focuses on the level of acne and considers each one independent of the others, to achieve the extracting latent factors that describe the placement of social network members in two or more communities and the explicit disclosure of cases of malicious use.

3. Proposed Methodology

Nonnegative factorization [24] of arrays is part of a broader family of dimensional techniques, which attempt to construct a partial representation of very high-dimensional data by projecting them into a lower-dimensional space. The difference in NMF from other methods is the limitation of the nonnegativity of the elements of the generated tables, which allows a better interpretation of the result. In the specific case we are considering, let $A \in R_+^{n\times n}$ be a table adjacent to a graph of n nodes. We want to factorize it into two nonnegative arrays: the basis matrix $W \in R_+^{n\times r}$ and the coefficient matrix $H \in R_+^{r\times n}$, so that [25–27]:

$$A \approx Ae \equiv WH, \tag{1}$$

where r is the number of communities and $r \ll n$.

The purpose of the NMF is to calculate the elements of W and H so that their product is as "close" as possible to A, with proximity being measured by some distance function. More strictly, nonnegative table factorization is the following combination optimization problem (which in this case is described as a minimization problem) [28, 29]:

$$\begin{aligned} &\min_{W,H}, D(A\|WH), \\ &\text{Subject to}, W \geq 0, \quad H \geq 0, \end{aligned} \tag{2}$$

where $D(\cdot||\cdot)$ is a distance function of arrays A and W H. In general, the problem of minimizing the function D is NP-hard, for which additionally no convex formulations are known that would lead to finding the total minimum of D concerning both tables W and H at the same time. Although the optimization of D is non-convex for both arrays simultaneously, it is nevertheless convex for each of the two arrays separately; i.e., keeping unchanged, e.g., W, the problem becomes convex for H (and vice versa) [30].

To create an immediate and accurate forecasting process, we use the Bayesian NMF and the benefits through the retrospective allocation. The Bayesian NMF is a subcategory of the probabilistic NMF, which approximates the parameters W and H using the classical relation of the Bayesian inference [31, 32]:

$$P(W, H, \Theta|A)\alpha P(A|W, H, \Theta), P(W, H|, \Theta), P(\Theta), \tag{3}$$

where the base and coefficient tables are the parameters of the model and Θ is the space of the hyperparameters (which regulate the statistical behavior of the distribution from which the tables W and H are derived). To use the above relation, a necessary condition is to speculate on the statistical origin of the neighborhood table data and the

product's table factors. The left part of the relation expresses the a posteriori probability that the model parameters receive a specified value based on the specific data. In contrast, the first term of the product of the right part of the same relation expresses the likelihood of the model, the next is the a priori probability of model parameters for the specific hyperparameters, and finally, the last term is the likelihood of occurrence of the hyperparameters themselves. A careful selection of the probability function and the ex ante probability can result in an algorithm that exhibits better and faster convergence [33, 34]. Thus, the model's parameters must be observed, and based on this observation, the probability distribution that best expresses their statistical properties must be selected. Then, after the probability formation has been clarified, the appropriate pre-possibility for the area of the hyperparameters is set. To choose the proper a priori probability and probability function, factorization is performed, so the arrangement of the edges between the nodes is expected to show star-type phenomena, i.e., some few nodes with a large number of tangent edges and many nodes with a small number of tangent edges [35–37].

Therefore, there are many "open" triangles, and the most suitable ERGM for the occasion is the 2-star model, which is calculated as [34, 38, 39]follows:

$$H = \theta m(G) + \tau s(G),$$
$$m(G) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{i,j}, s(G) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{i,j} \sum_{k=1(\neq j)}^{n-1} a_{ik}, \tag{4}$$

where a_{ij} is the element of the graph adjacent table A (with value 1 if there is an edge between nodes i and j, and 0 otherwise), $m(G)$ is the network statistic that models the number of edges of the graph (whose influence is controlled by the hyperparameter i), and $s(G)$ is the corresponding magnitude for the number of 2 stars (whose influence is controlled by the hyperparameter t, respectively). It should also be noted that in this case, the graph is nondirectional (i.e., for the i and j elements of the neighborhood table, the equation $a_{ij} = a_{ji}$ applies). Since there is no detailed solution for the model described, we must resort to approximate techniques for estimating the value of the hyperparameters θ and τ. As a first step, we rewrite the network statistics as a function, not of the edges, but the degree of k_i each node [33, 39, 40]:

$$m(G) = \frac{1}{2}\sum_{i=1}^{n} k_i, s(G) = \frac{1}{2}\sum_{i=1}^{n} k_i(k_i - 1) = \frac{1}{2}\sum_{i=1}^{n} k_i^2 - m(G). \tag{5}$$

Then, the Hamiltonian is derived as follows:

$$H = \theta m(G) + \tau s(G) = \frac{\tau}{2}\sum_{i=1}^{n} k_i^2 + \frac{\theta - \tau}{2}\sum_{i=1}^{n} k_i. \tag{6}$$

To facilitate the following calculations, the hyperparameters θ and τ are replaced by the auxiliary hyperparameters J and B, which are defined as follows:

$$J = \frac{(n-1)\tau}{2}, \quad B = \frac{\tau - \theta}{2}, \tag{7}$$

so the Hamiltonian takes its final form:

$$H = \frac{J}{n-1}\sum_{i=1}^{n} (k_i)^2 + B\sum_{i=1}^{n} k_i. \tag{8}$$

Comparing the equations, we observe that they express the same ERGM [41], i.e., the 2-star model, using different network statistics. The free energy of the model described by the Hamiltonian is calculated as follows [40, 42, 43]:

$$F = -n(n-1)J(\phi_0)^2 + \frac{1}{2}n(n-1)\ln\left(1 + e^{4J\phi_0 + 2B}\right) + \frac{n}{2}\ln[(n-1)J] - \frac{n}{2}\ln 4\pi, \tag{9}$$

where φ_0 is the solution of the mean field approach given as follows:

$$\phi_0 = \frac{1}{2}\left[\tanh(2J\phi_0 + B) + 1\right]. \tag{10}$$

The partial derivative of the free energy of the model for the hyperparameter B is equal to the expected value of the sum of the degrees of the nodes:

$$\langle \sum_{i=1}^{n} k_i \rangle = \frac{\partial F}{\partial B} \Rightarrow \sum_{i=1}^{n} \langle k_i \rangle = \frac{\partial F}{\partial B}. \tag{11}$$

By approximating the expected value of the degree of each node $\langle k_i \rangle$ with its most probable/expected value, i.e., the predicted value $\langle k_i \rangle$ of the degree of all nodes, we have [44] the following:

$$\sum_{i=1}^{n} \langle k_i \rangle = \frac{\partial F}{\partial B} \Rightarrow n\langle k \rangle = \frac{\partial F}{\partial B} \Rightarrow \langle k \rangle = \frac{1}{n}\frac{\partial F}{\partial B} \Rightarrow$$
$$\langle k \rangle = (n-1)\frac{e^{4J\phi_0 + 2B}}{1 + e^{4.J\phi_0 + 2B}} = (n-1)\frac{1}{2}\left[\tanh(2J\phi_0 + B)\right]$$
$$\langle k \rangle = (n-1)\phi_0 \Rightarrow \phi_0 = \frac{n-1}{\langle k \rangle}, \tag{12}$$

making φ_0 equal to:

$$\phi_0 = \frac{\langle k \rangle}{n-1}, \quad \phi_0 \in (0, 1). \tag{13}$$

Similarly, the partial derivative of free energy for the auxiliary hyperparameter $J/n-1$ is equal to the expected value of the sum of the squares of the degrees of the nodes [45]:

$$\langle \sum_{i=1}^{n} k_i^2 \rangle = \sum_{i=1}^{n} \langle k_i^2 \rangle = \frac{\partial F}{\partial (J/n-1)}. \tag{14}$$

So, the expected value of the square of the degree of a node $\langle k_i^2 \rangle$ is approximated by its most probable/expected

value, that is, the expected value of the square of the degree of all nodes [28, 34, 46]:

$$\langle \sum_{i=1}^{n} k_i^2 \rangle = \sum_{i=1}^{n} \langle k_i^2 \rangle = \frac{\partial F}{\partial (J/n-1)} \Rightarrow n\langle k^2 \rangle = (n-1)\frac{\partial F}{\partial J} \Rightarrow \langle K^2 \rangle = \frac{n-1}{n}\frac{\partial F}{\partial J} \Rightarrow$$
$$\langle k^2 \rangle = -(n-1)^2(\phi_0)^2 + 2(n-1)^2 \frac{e^{4,J\phi_0+2B}}{1+e^{4J\phi_0+2B}} + \frac{1}{2}(n-1)\frac{1}{J} \Rightarrow$$
$$\langle k^2 \rangle = -(n-1)^2(\phi_0)^2 + 2(n-1)^2 \frac{1}{2}\left[\tanh(2J\phi_0+B)+1\right] + \frac{1}{2}(n-1)\frac{1}{J} \Rightarrow$$
$$\langle k^2 \rangle = -(n-1)^2(\phi_0)^2 + 2(n-1)^2\phi_0 + \frac{1}{2}(n-1)\frac{1}{J}. \tag{15}$$

So:

$$\langle k^2 \rangle = \langle -k^2 \rangle + 2(n-1)\langle k \rangle + \frac{1}{2}(n-1)\frac{1}{J} \Rightarrow$$
$$\frac{1}{2}(n-1)\frac{1}{J} = \langle k^2 \rangle + \langle k \rangle^2 - 2(n-1)\langle k \rangle \Rightarrow \tag{16}$$
$$J = \frac{(n-1)}{2\left(\langle k^2 \rangle + \langle k^2 \rangle - 2(n-1)\langle k \rangle\right)}.$$

So:

$$2\phi_0 - 1 = \tanh(2J\phi_0 + B) \Rightarrow 2J\phi_0 + B = \tanh^{-1}(2\phi_0 - 1) \Rightarrow$$
$$B = \tanh^{-1}(2\phi_0 - 1) - 2J\phi_0. \tag{17}$$

Given the following identity, we have the following:

$$\tan^{-1}(x) = \frac{1}{2}\ln\frac{1+x}{1-x}, \quad |x| < 1. \tag{18}$$

So, it turns out:

$$B = \frac{1}{2}\ln\phi_0 - \frac{1}{\langle k^2 \rangle + \langle k \rangle^2 - 2(n-1)\langle k \rangle}. \tag{19}$$

Therefore, the above equations fully describe the 2-star model in terms of the mean field. Given the original Hamiltonian of the solvable 2-star model, then the following relation is true [36, 40]:

$$H = -\frac{J}{n-1}\sum_{i=1}^{n} k_2^i + B\sum_{i=1}^{n} k_i = \frac{\partial F}{\partial J}J + \frac{\partial F}{\partial B}B = \nabla F. \tag{20}$$

So, the Hamiltonian of the model is equal to the gradient of the free energy, and by substituting the values of B and J, we have [44]the following:

$$H = \nabla F = \frac{\partial F}{\partial J}J + \frac{\partial F}{\partial B}B$$
$$= -n(n-1)J(\phi_0)^2 + 2n(n-1)\phi_0 + \frac{n}{2} + n(n-1)B\phi_0. \tag{21}$$

Therefore:

$$H = -n(n-1)J(\phi_0)^2 + 2n(n-1)J\phi_0 + n(n-1)B\phi_0$$
$$= n(n-1)\phi_0\left[-J\phi_0 + 2J + B\right]$$
$$= n\langle k \rangle\left[-J\phi_0 + 2J + \tanh^{-1}(2\phi_0 - 1) - 2J\phi_0\right]$$
$$= \sum_{i=1}^{n}\langle k_i \rangle\left[2J - 3J\phi_0 + \tanh^{-1}(2\phi_0 - 1)\right] \tag{22}$$
$$= 2\left[2J - 3J\phi_0 + \tanh^{-1}(2\phi_0 - 1)\right]\frac{1}{2}\sum_{i=1}^{n} k_i$$
$$= \left[2\tanh^{-1}(2\phi_0 - 1) + 2(2 - 3\phi_0)J\right]m(G).$$

So, we end up:

$$H = \Theta m(G), \quad \Theta = \ln\phi_0 + 2(2 - 3\phi_0)J. \tag{23}$$

This is a significant conclusion for two reasons. Firstly, we were able to find an approximate solution in the 2-star model. Secondly, the specific solution we found can be easily integrated into the factorization process, thus displaying the entities that create the spam [14, 17].

Regarding the probability that models the local properties of the graph edges, in the probabilistic NMF we propose, the Poisson distribution optimizes the generalized Kullback–Leibler deviation. Thus, the following function greatly simplifies the required calculations [24, 44, 47]:

$$p\left(\widetilde{a_{ij}} | a_{ij}\right) \alpha \frac{\widetilde{a_{ij}} a_{ij}}{a_{ij}!} e^{-\widetilde{a_{ij}}}. \tag{24}$$

Having chosen the probability distribution and the ex-probability, we can now use the classical relation of the Bayesian inference to approach the ex post probability that the parameters of our model (the elements of the table $Ae = WH$) take specific values, as well as the hyper-parameters Θ of the model [48].

For the proposed model to succeed in imitating the actions of the human brain in a simplified way, the attention mechanism is used, which is also an attempt to implement

the same measure of selective human concentration in some relevant things while ignoring some, respectively. This procedure allows for the particular treatment of different versions of the same situation and identifying events that significantly change the proposed control process. An abstract implementation related to the environment vector c_i for output y_i is generated using the weighted sum of the annotations so that [44]:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j. \tag{25}$$

In the simplest case, the weights α_{ij} are calculated from a softmax function given by the following equation:

$$\alpha_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(e_{ik}\right)}, \tag{26}$$

where e_{ij} is the output rating of the feed mechanism described by function a, which attempts to record the alignment between the input to j and output to i.

So given that the element $\widetilde{a_{ij}}$ results from the interior product of the ith row vector of the table W on the jth column vector of H, we arrive at the following relation [40, 44, 49]:

$$P\left(w_i^T h_j | a_{ij}, \Theta\right) \alpha \frac{w_i^T h_j^{a_{ij}}}{a_{ij}!} e^{(\Theta-1)} w_i^T h_j. \tag{27}$$

The slope of the equation for the vectors $w_i^T h_j$ is calculated as follows:

$$\begin{aligned} \nabla D_{w_i^T}\left(a_{ij}, w_i^T, h_j\right) &= \sum_{j=1}^{k}\left[-\frac{a_{ij}}{w_i^T h_j} h_J^T + (1-\Theta) h_j^T\right] \\ &= -\frac{a_i^T}{\widetilde{a_1}^T} H^T + (1-\Theta) e^T H^T, \\ \nabla D_{hj}\left(a_{ij}, W_i^T, h_j\right) &= \sum_{j=1}^{k}\left[-W_i^T + \frac{a_{ij}}{W_i^T h_j} + (1-\Theta) W_i^T\right] \\ &= -W^T \frac{a_i^T}{\widetilde{a_1}^T} + (1-\Theta) W^T e^T, \end{aligned} \tag{28}$$

while, respectively, they are renewed as follows:

$$\begin{aligned} \left(W_i^T\right)^{t+1} &\leftarrow \left(W_i^T\right)^{(t)} \circ \frac{\nabla D^-_{W_1^T}}{\nabla D^-_{W_1^T}} \Rightarrow \left(W_i^T\right)^{(t+1)} \leftarrow \frac{a_1^T / \widetilde{a_1} H^T H^T}{(1-\Theta) e^T H^T} \\ h_j^{(t+1)} &\leftarrow h_j^{(t)} \circ \frac{\nabla D^-_{h_j}}{\nabla D^+_{h_j}} \Rightarrow h_j^{(t+1)} \leftarrow h_j^{(t)} \circ \frac{W a_1^T / \widetilde{a_1}}{(1-\Theta) W^T e^T}, \end{aligned} \tag{29}$$

resulting in the following information rules for the base tables and coefficients W and H:

$$\begin{aligned} W^{(t+1)} &\leftarrow W^{(t)} \circ \frac{A/WH\, H^T}{(1-\Theta) E^T H^T}, \\ H^{(t+1)} &\leftarrow H^{(t)} \circ \frac{W^T A/WH}{(1-\Theta) W^T E}. \end{aligned} \tag{30}$$

Thus, in combination with the deterministic calculation of Θ, we have the following:

$$1 - \Theta > 0 \Rightarrow \Theta < 1 \Rightarrow \ln\phi_0 + 2\left(2 - 3\phi_0\right) J < 1. \tag{31}$$

In conclusion, all social networks are far from being characterized as cliques, so the value of connectivity φ_0 is much lower than unity and closer to zero (i.e., $\varphi_0 \ll 1$ applies). Therefore, the term $\ln \varphi_0$ takes very small (negative) values. On the other hand, the hyperparameter J is positive (its use in the Hubbard–Stratonovich transform imposes its nonnegativity), so the second term of the sum is positive (it is an exact mathematical transformation that is used to convert a particle theory into its respective field theory by linearizing the density operator in the many-body interaction term of the Hamiltonian and introducing an auxiliary scalar field [50]). Thus, in social network graphs, the effect of the term $\ln \varphi_0$ is stronger in shaping the final value of the hyperparameter Θ, so the above relation applies to all networks where the original hypotheses investigated apply. The contribution of ex ante probability to factorization (more specifically of ERGMs) and its contribution to the aggregation of members of a user's network to generate recommendations are to be quantified in the following experimental process.

4. Scenarios and Results

To model an email referral system, this study uses user actions about activity and the likelihood that a node will engage in abnormal behavior related to the spread of spam. The application is used based on a case study applied in the environment of educational organizations. Instead of classifying two classes into junk and desirable, we treat the problem as multi-category classification in which each class is a user recommendation action in an email. The most common activities are reading, replying, saving, waiting, deleting from the mailing list, terminating, including junk mail filters, and deleting. As part of a collaborative social filtering system, the previously described algorithm was applied to two different datasets containing user actions in mail messages and information about their social relationships.

The first collection is small and consists of usage data and, more specifically, the relevant statistics of a user's traffic to an educational podcast service. On the other hand, the second collection is medium-sized and contains educational material evaluations on an online learning platform on a five-point scale. Despite their differences, the two datasets

are highly sparse and show the characteristics of free-scale networks regarding the number of ratings they contain and the distribution of social network nodes (most edges fall on a few nodes, while most nodes touch a few edges). A user's ratings are extracted from the data collection and divided into two distinct sets: training and testing. The test set is then repositioned in the data collection. In the next step, the algorithms generate recommendations, and a list of objects is returned as output, compared with the data in the test set. The whole process is repeated five times for a list size of 5 to 25 items. For the generated recommendations to make sense, in each iteration of the experimental protocol, only users are selected who have evaluated at least twice the number of objects from the respective list size. Several memorandum collaborative systems were implemented to assess the quality of the recommendations produced in different environments.

The levels of the metrics refer to the average of the respective values, while the similarity function used was the logarithm of the probability ratio of the data. Local-level trust metrics were also examined, namely the MoleTrust 1 algorithm (which considers only the users with whom the user in question is directly associated with acne) and the MoleTrust 2 (which also finds the neighbors of the user in question). TidalTrust was also implemented, which calculates user similarity based on the shortest paths between the user in question and all other users on the same connected component. The comparison also includes an algorithm that calculates each user's reputation on the social network and, more specifically, TrustWalker, which takes a random walk on the graph, selecting its next step uniformly at random. The random walk starts from the user in question, and when it reaches its stationary distribution, the nodes with the highest probability are returned as more similar. Finally, the user network clustering methodology was tested to estimate the relative performance of ERGMs in the position of ex ante probability. The configuration includes the Bayesian NMF algorithm using the Poisson distribution in place of the likelihood. The method of filtering the neighbors that were applied was that of the nearest-N; i.e., in producing the recommendations, only the N closest neighbors are taken into account. The value of parameter N was set to 5 after a series of verification experiments in which it was found that for values of N less than 5, the results of the metrics were unstable. In contrast, the results were lower for values of N greater than 5. It should be noted, however, that the relative classification of the algorithms remains constant, regardless of the value of N [20, 23, 31, 44, 51].

A first observation of the results is that the competing algorithms perform significantly lower performance than the proposed one. This behavior is attributed to the fact that the proposed methodology's interaction between users and objects is deemed valid and can be the basis for a possible recommendation. Another interesting point is that the density ratio between the two datasets is reflected almost linearly in the individual results achieved by the different systems. An equally important observation is that each user's network is a good source of information for making recommendations in sparse datasets, clearly superior to traditional collaborative methods. As the level of data dilution increases, the efficiency of conventional collaborative recommendation algorithms decreases, as there are fewer and fewer users whose object ratings match. In this case, social algorithms "unfold" all their dynamics. The results prove the apparent superiority of the social algorithms that perform a local search on all metrics. It is also worth noting that all three algorithms based on local social network search show similar results even though they explore each user's neighborhood at a different depth.

Nevertheless, they do not perform well in either dataset. The above observation concludes that basing the recommendations exclusively on the social network's most popular (or the most frequently visited) nodes does not guarantee similarity in preferences. Finally, the proposed methodology seems to achieve the best results compared with the local and the full search on the social network. Placing the nodes that are part of a user's network in overlapping clusters leads to complete analysis of users' proximity to the network, especially when compared to the basic assumptions made by other algorithms. The addition of ERGMs further improves this analysis in place of the ex ante probability. In this way, the central view given to each acne by the Bayesian NMF is addressed, i.e., by introducing structural features of the graph in the process.

5. Conclusions

The consequent increase in the popularity of online educational resources, combined with this lack of preparedness, has made the education sector an ideal target for digital phishing attacks. The detection of spam and the timely assessment of these threats allow the detection of events that can significantly mitigate the effects of organized cyber-attacks. An advanced deep attention collaborative filter was presented to ensure the educational processes and the protection of the educational system. It is a specialized application of intelligent techniques where the spam problem is examined as a social graph to identify harmful communities. Using a deep attention mechanism, the methodology becomes personalized for each user, while critical, innovative optimization processes help evaluate social graphs.

This methodology is being tested with great success in implementing mail protection systems for educational organizations based on the integration of mnemonic, social, and collaborative recommendation systems. The results obtained show a steady improvement of all performance metrics compared with all comparable implementations. This improvement is attributed to the unique filtering capabilities of the proposed methodology. Instead of summing up many users, it tries to discover patterns in their social behavior by (overlapping) grouping them into regions. This process bears similarities to how overlapping finding commonalities algorithms are operating.

A possible research direction would be to include factorizing more complex network features, such as triangles. In this case, however, the model calculations become pretty complicated. It is not easy to derive an approximate solution

similar to the one presented, so the implementation with advanced equipment such as GPU or TPU should be investigated. Accordingly, a more generalized approach to model estimation should be explored as to whether they would achieve better results and greater generalization in community evaluation. Finally, adopting post-hybrid methods of automatic optimization of communities is considered very important for the further development and use of the methodology.

Data Availability

The data are available on reasonable request.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] W. Holmes and S. Anastopoulou, "What do students at distance universities think about AI?" in *Proceedings of the Sixth (2019) ACM Conference on Learning @ Scale*, pp. 1–4, Chicago IL USA, June. 2019.

[2] N. A. Khan, S. N. Brohi, and N. Zaman, "Ten deadly cyber security threats amid COVID-19 pandemic," May 2020.

[3] T. Ahmad, "Corona virus (COVID-19) pandemic and work from home: challenges of cybercrimes and cybersecurity," *SSRN Electronic Journal*, 2020.

[4] J. Chigada and R. Madzinga, "Cyberattacks and threats during COVID-19: a systematic literature review," *SA Journal of Information Management*, vol. 23, no. 1, February. 2021.

[5] T. Weil and S. Murugesan, "IT risk and resilience-cr," *IT Professional*, vol. 22, no. 3, pp. 4–10, May 2020.

[6] M. Iezzi, "Practical privacy-preserving data science with homomorphic encryption: an overview," in *Proceedings of the 2020 IEEE International Conference on Big Data (Big Data)*, pp. 3979–3988, Atlanta, GA, USA, December 2020.

[7] R. Khweiled, M. Jazzar, and D. Eleyan, "Cybercrimes during COVID -19 Pandemic," *International Journal of Information Engineering and Electronic Business*, vol. 13, no. 2, April. 2021.

[8] M. Hijji and G. Alam, "A multivocal literature review on growing social engineering based cyber-attacks/threats during the COVID-19 pandemic: challenges and prospective solutions," *IEEE Access*, vol. 9, pp. 7152–7169, 2021.

[9] S. Venkatesha, K. R. Reddy, and B. R. Chandavarkar, "Social engineering attacks during the COVID-19 pandemic," *SN Computer Science*, vol. 2, no. 2, p. 78, April. 2021.

[10] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, Mar. 2021.

[11] B. Deng, X. Zhang, W. Gong, and D. Shang, "An overview of extreme learning machine," in *Proceedings of the 2019 4th International Conference on Control, Robotics and Cybernetics*, pp. 189–195, CRC, Tokyo, Japan, September. 2019.

[12] S. Raschka, "An overview of general performance metrics of binary classifier systems," October. 2014, http://arxiv.org/abs/1410.5330.

[13] P. Bevington and D. K. Robinson, *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill Education, New York, 2003.

[14] A. A. Abdullahi and M. Kaya, "A deep learning based method to detect email and SMS spams," in *Proceedings of the 2021 International Conference on Decision Aid Sciences and Application (DASA)*, pp. 430–435, Sakheer, Bahrain, December. 2021.

[15] Y. Fang, C. Zhang, C. Huang, L. Liu, and Y. Yang, "Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism," *IEEE Access*, vol. 7, pp. 56329–56340, 2019.

[16] S. Phomkeona and K. Okamura, "Zero-day malicious email investigation and detection using features with deep-learning approach," *Journal of Information Processing*, vol. 28, no. 0, pp. 222–229, 2020.

[17] S. Kaddoura, O. Alfandi, and N. Dahmani, "A spam email detection mechanism for English language text emails using deep learning approach," in *Proceedings of the 2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Bayonne, France*, pp. 193–198, Bayonne, France, September. 2020.

[18] D. S. Asudani, N. K. Nagwani, and P. Singh, "Exploring the effectiveness of word embedding based deep learning model for improving email classification," *Data Technologies and Applications*, February. 2022.

[19] N. Ali, A. Fatima, H. Shahzadi, A. Ullah, and K. Polat, "Feature extraction aligned email classification based on imperative sentence selection through deep learning," *Journal of Artificial Intelligence and Systems*, vol. 3, no. 1, pp. 93–114, 2021.

[20] C. C. Aggarwal, "Neighborhood-based collaborative filtering," in *Recommender Systems: The Textbook*, pp. 29–70, Springer International Publishing, New York, 2016.

[21] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261–273, 2015.

[22] M. Deschênes, "Recommender systems to support learners' Agency in a Learning Context: a systematic review," *International Journal of Educational Technology in Higher Education*, vol. 17, no. 1, p. 50, October. 2020.

[23] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*, pp. 291–324, Springer, Berlin, Heidelberg, 2007.

[24] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, October. 1999.

[25] N. Thai-Nghe, L. Drumond, A. Krohn-Grimberghe, and L. Schmidt-Thieme, "Recommender system for predicting student performance," *Procedia Computer Science*, vol. 1, no. 2, pp. 2811–2819, 2010.

[26] M. Srifi, A. Oussous, A. Ait Lahcen, and S. Mouline, "Recommender systems based on collaborative filtering using review texts-A survey," *Information*, vol. 11, no. 6, pp. 317–6, June. 2020.

[27] I. A. A.-Q. Al-Hadi, N. M. Sharef, N. Sulaiman, and N. Mustapha, "Review of the temporal recommendation system with matrix factorization," *International Journal of Innovative Computing, Information and Control*, vol. 13, p. 16, 2017.

[28] S. Gupta, S. Al-Obaidi, and L. Ferrara, "Meta-analysis and machine learning models to optimize the efficiency of self-healing capacity of cementitious material," *Materials*, vol. 14, no. 16, p. 4437, January. 2021.

[29] S. Joshi, T. Jain, and N. Nair, "Emotion based music recommendation system using LSTM - CNN architecture," in

Proceedings of the 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 01–06, Kharagpur, India, July. 2021.

[30] G. Alexandridis, G. Siolas, and A. Stafylopatis, "Accuracy versus novelty and diversity in recommender systems: a nonuniform random walk approach," in *Recommendation and Search in Social Networks*, Ö. Ulusoy, A. U. Tansel, and E. Arkun, Eds., Springer International Publishing, New York, NY,, pp. 41–57, 2015.

[31] J. O. Berger, "Bayesian analysis," in *Statistical Decision Theory and Bayesian Analysis*, pp. 118–307, Springer, New York, NY, 1985.

[32] M. S. Mahmud, J. Z. Huang, S. Salloum, T. Z. Emara, and K. Sadatdiynov, "A survey of data partitioning and sampling methods to support big data analysis," *Big Data Mining and Analytics*, vol. 3, no. 2, pp. 85–101, 2020.

[33] J. Chanda, S. Sengupta, A. Kanjilal, and S. Bhattacharya, "CA-graph: a context aware graph to model enterprise service bus," in *Proceedings of the 2010 Annual IEEE India Conference (INDICON)*, pp. 1–4, Kolkata, India, December 2010.

[34] M. Gori, M. Maggini, and L. Sarti, "Graph matching using random walks," in *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 3, pp. 394–397, Cambridge, UK, August 2004.

[35] B. K. Deka, "Transformations of graph database model from multidimensional data model," in *Proceedings of the 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 2836–2840, New Delhi, India, March. 2016.

[36] B. He, Y. Cui, J. Chen, and P. Xie, "A spatial data mining method for mineral resources potential assessment," in *Proceedings of the 2011 IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services*, pp. 96–99, Fuzhou, China, June. 2011.

[37] S.-Y. Kim and Y.-H. Yoo, "A new haptic model using bond graphs," in *Proceedings of the 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology*, pp. 387–392, Seoul, Korea (South), November 2009.

[38] A. E. Barinov and A. A. Zakharov, "Clustering using a random walk on graph for head pose estimation," in *Proceedings of the 2015 International Conference on Mechanical Engineering, Automation and Control Systems (MEACS)*, pp. 1–5, Tomsk, Russia, December 2015.

[39] V. Nguyen, J. Leeka, O. Bodenreider, and A. Sheth, "A formal graph model for RDF and its implementation," June. 2016, http://arxiv.org/abs/1606.00480.

[40] S. Blyumin, A. Pogodaev, and E. Khabibullina, "Graph-structural modeling of some special organizational systems," in *Proceedings of the 2020 2nd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA)*, pp. 279–283, Lipetsk, Russia, November 2020.

[41] G. Alexandridis, G. Siolas, and A. Stafylopatis, "An efficient collaborative recommender system based on k-separability," in *Artificial Neural Networks – ICANN 2010*pp. 198–207, Berlin, Heidelberg, 2010.

[42] P. MohanaPriya and S. M. Shalinie, "Restricted Boltzmann machine based detection system for DDoS attack in software defined networks," in *Proceedings of the 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*, pp. 1–6, Chennai, India, March. 2017.

[43] S. Goel and V. K. Panchal, "Performance evaluation of a new modified firefly algorithm," in *Proceedings of the Infocom Technologies and Optimization Proceedings of 3rd International Conference on Reliability*, pp. 1–6, Noida, India, October 2014.

[44] K. Han, Y. Li, and B. Xia, "A cascade model-aware generative adversarial example detection method," *Tsinghua Science and Technology*, vol. 26, no. 6, pp. 800–812, 2021.

[45] G. Alexandridis, G. Siolas, and A. Stafylopatis, "A biased random walk recommender based on Rejection Sampling," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)*, pp. 648–652, Niagara Ontario Canada, August 2013.

[46] P. Gattineni and G. R. S. Dharan, "Intrusion Detection Mechanisms: SVM, random forest, and extreme learning machine (ELM)," in *Proceedings of the 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 273–276, Coimbatore, India, September. 2021.

[47] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," May 2014, http://arxiv.org/abs/1312.6114.

[48] G. Alexandridis, G. Siolas, and A. Stafylopatis, "Enhancing social collaborative filtering through the application of non-negative matrix factorization and exponential random graph models," *Data Mining and Knowledge Discovery*, vol. 31, no. 4, pp. 1031–1059, July. 2017.

[49] V. Alieksieiev and B. Andrii, "Information analysis and knowledge gain within graph data model," in *Proceedings of the 2019 IEEE 14th International Conference on Computer Sciences and Information Technologies(CSIT)*, vol. 3, pp. 268–271, Lviv, Ukraine, September. 2019.

[50] K. Seki, Y. Otsuka, and S. Yunoki, "Gutzwiller wave function on a quantum computer using a discrete Hubbard-Stratonovich transformation," January. 2022, http://arxiv.org/abs/2201.11381.

[51] S. Trinks and C. Felden, "Real time analytics - sp," in *Proceedings of the 2017 IEEE International Conference on Big Data (Big Data*, pp. 4843–4845, Boston, MA, USA, December 2017.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6294058, 12 pages
https://doi.org/10.1155/2022/6294058

Research Article

Digital Forensics for Malware Classification: An Approach for Binary Code to Pixel Vector Transition

Muhammad Rehan Naeem,[1] **Rashid Amin**,[1] **Sultan S. Alshamrani**,[2] **and Abdullah Alshehri**[3]

[1]*Department of Computer Science, University of Engineering and Technology Taxila, Taxila, Pakistan*
[2]*Department of Information Technology College of Computer and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia*
[3]*Department of Information Technology, Al Baha University, Al Baha, Saudi Arabia*

Correspondence should be addressed to Rashid Amin; rashid4nw@gmail.com

Received 10 March 2022; Revised 2 April 2022; Accepted 7 April 2022; Published 21 April 2022

Academic Editor: Konstantinos Demertzis

The most often reported danger to computer security is malware. Antivirus company AV-Test Institute reports that more than 5 million malware samples are created each day. A malware classification method is frequently required to prioritize these occurrences because security teams cannot address all of that malware at once. Malware's variety, volume, and sophistication are all growing at an alarming rate. Hackers and attackers routinely design systems that can automatically rearrange and encrypt their code to escape discovery. Traditional machine learning approaches, in which classifiers learn based on a hand-crafted feature vector, are ineffective for classifying malware. Recently, deep convolutional neural networks (CNNs) successfully identified and classified malware. To categorize malware, a smart system has been suggested in this research. A novel model of deep learning is introduced to categorize malware families and multiclassification. The malware file is converted to a grayscale picture, and the image is then classified using a convolutional neural network. To evaluate the performance of our technique, we used a Microsoft malware dataset of 10,000 samples with nine distinct classifications. The findings stood out among the deep learning models with 99.97% accuracy for nine malware types.

1. Introduction

Malware assaults increasingly pose a serious security threat to the internet and computer networks. According to Symantec research, 123 million devices record hundreds of harmful threat behaviors per second [1]. They are the most widely recognized computer security dangers. The figures are staggering, with some antivirus vendors reporting daily malware samples of more than 5 million. The number of internet-connected devices is predicted to reach 200 billion by 2020 [2], and they can be elaborate and add complexity, but you can be more specific. That malware is too much for security professionals to manage at once. To prioritize these events, a malware classification mechanism is frequently required.

Malware attacks on mobile devices and the internet of things (IoT) are becoming more common. Thanks to the complex system software environment and sensory devices, adversaries will find it easier to attack the system. Malware is harmful software that wreaks havoc on our digital systems' functionality, privacy, and dependability. There are several forms or families of malware, such as Trojans, Backdoors, and Worms, among others. Viruses and malware are currently among the most dangerous threats to our systems [3].

To conceal their identity, malware authors utilize a variety of approaches and strategies while writing code. As a result, determining the malware family or kind is the most difficult component. Traditional antivirus software struggles to keep up with the massive volume of malware that emerges every day. Computer scientists and antivirus companies

have begun to use machine learning models to overcome this problem. Academic researchers and developers have proposed machine learning classifiers such as neural networks and logistic regression to classify malicious software [4, 5]. Clustering and categorizing the files' respective families are the first steps toward efficiently arranging and assessing many of them. Furthermore, such clustering criteria may be useful in detecting harmful and specific family members of newly found files on our computers. We require malware files with their relatives to group them and define new malware based on those clusters to enable study in this sector. As a result, the malware industry has evolved vast and is well organized. In the internet era, malware attacks on financial institutions and common individuals are rising. To deal with this fast malware development, flexible malware categorization algorithms for variants of malware files belonging to the same family are required [6–8]. A method of classifying malware by its family, regardless of whether it is a true variation, looks to be a very productive and successful technique for dealing with malware's fast expansion. Here are a few scenarios in which a solution to this challenge may be beneficial. Antimalware generator is the first, while malware developer identification is the second.

Analysts benefit from malware classification since they can further probe the malware's operations. Malware with similar structures is clustered together into a single cluster [9]. Furthermore, by identifying the malware's family, we may understand how it operates. Because of high-tech development evolutions in the digital software environment, mobile environment, social networks, smart cities, cloud computing, internet of things (IoT), and other areas, malware analysis and classification are a fast-growing sector requiring attention. Researchers have achieved great results using recurrent neural networks (RNNs) for speech recognition and handwriting identification [5, 10]. Many researchers have used machine learning approaches to identify and categorize malware. Machine learning-based malware detection comprises two phases: the first involves extracting features from photographs, and the second involves malware categorization [11, 12]. Using theoretical methodologies, automated malware detection takes too long and produces inaccurate findings. The automatic program that implements the new systems learned from this manual study improves its performance. We were inspired by Fred R. Barnard's quote, "A picture is worth a thousand words." This study also looks into malware to determine whether the same holds. Visualizations have always been important in gaining a clear understanding of any framework or data. Visuals make more sense to us logically than any other representation [13, 14]. So the issue now is whether these sequential models can effectively detect malware families. Deep learning models were utilized in our research to categorize malware into different families.

Most malware classification algorithms employ feature vectors, which indicate harmful traits [15]. This research will classify both existing malware and new malware generated in the future. Antimalware businesses can swiftly develop antimalware for current malware and any future malware. Our main contribution to this research is that we use DL models to classify malware of various forms, and we examined previous work on malware image-based classification. Our methodology does not need feature engineering or domain expertise, such as binary disassembly, reverse engineering, or assembly language. Because we have supplied the picture dataset straight to the model as input, our technique may be used for real-time classification due to the low preparation time. We tested our model against the Microsoft dataset to categorize the new malware, which contained nine different malware families. We created a novel malware classification method that is computationally cost-effective, scalable, and efficient, based on single or hybrid deep learning models.

2. Related Work

More than a few researchers focused on malware visualization to better classify and identify malware studies to get maximum accuracy and in less time. This research section presents the related malware identification, visualization, and classification for which we used deep learning and ML models as a foundation.

Shaid et al. [16] proposed an approach to malware detection that relies on the unique behavior of malware executable files, which has been presented. The main thing is to identify any similarity in the conduct of malware samples. The researchers executed the malware executable file in the virtual environment in this technique. When they run the malware, something like an image or pattern is generated. They check the pattern sequence using a color map and check the similarity of the behavior in those images by using some statistical techniques. They got higher accuracy ranging from 95.91% to 98% by taking 1,102 malware image samples, and they got it from 12 different malware families. But this approach of exception malware in a virtual environment proves very time-intensive. Naeem et al. [17] introduced a malware classifier that worked on feature extraction first, and then to categorize the virus, they adopted a support-vector machine (SVM). This technique got 97.4% classification accuracy using a dataset containing 25 malware families with 9,339 sample files. But as a result, conventional methods need a feature analysis that takes a long time to compute.

For image categorization, we turn to deep learning to solve this problem. Our proposed solution is based on a new methodology recently developed. The author applies CNN for malware [5, 18]. Their base model applied different architectures, but the model is relatively narrow in that working style. Jhu-Sin Luo proposed the method that executes with GPU by using TensorFlow, which takes a significantly shorter time for processing. But, this scheme did not work on the virtual device and cannot identify the malware behavior. This technique is based on image recognition. But there are still some flaws in this method. If a developer differently rewrites the complete code, the results will be changed, and this technique will fail [11]. Dey et al. provide information on existing malware detection techniques based on image processing. They used an entropy filter to find the patterns in the images and got a better result

[12, 19] than Natraj et al. [20]. Han et al. propose many techniques of visual matrixes shown in RGB colors based on the opcode sequences in the executable malware files; subsequently, to determine how similar the two datasets were, they were compared using entropy graphs between images. But these techniques worked only for the Windows PE files and were unable to work on packed data samples [21].

A new feature fusion technique to reassemble the features extracted from pretrained AlexNet has been proposed here. They used 25 classes to identify the malware and used the images removed from it to identify the malware using various SVM, decision tree, and K-nearest neighbor variants [13]. Utilizing these machine learning variants achieved 99.3 percent accuracy [10, 22]. Natraj et al. [20] introduced a different scheme of malware classification by feature extraction. The author converted the binary executable malware file into 2D grayscale images. They generated images and used a visible pattern or features to detect malware. Results indicated that it was more accurate and substantially less time-consuming than previous methods. To strengthen the security of existing infrastructure, scientists look at malware samples to figure out how they operate and the tactics that malware authors utilize. Malware analysis is being used for the classification of malware [7]. Determining which class a piece of malware belongs to is known as classification. After determining that a file is malware, we must evaluate its family. Static and dynamic malware analyses are the two main forms of analysis. It is possible to analyze a sample without running it through a process known as static analysis. In contrast, dynamic analysis is the process of executing a sample to determine its behavior like how it performed in different environments [23]. But we used a different approach; we have analyzed malware files by converting executable malware files into grayscale images, so in this way, there is no harm to our system, which does not need to execute the file.

2.1. Static Malware Analysis. Static malware analysis involves thoroughly testing a binary without attempting to execute it. This approach can be used for a variety of executable representations. It is much easier to perform static analysis when the actual code is present [17]. The binary file can be disassembled, and the assembly code is examined if the code is not accessible. Printable strings in the program's header, deconstructing the program, identifying byte sequences, examining the file's structure, and so on are all examples of static analysis [24, 25].

2.2. Dynamic Malware Analysis. It is possible to identify malware via dynamic analysis by running an executable file and observing its behavior. The malware is operated in a virtualized environment or VM to keep the virus' effects contained. Using a virtual machine, we can take a snapshot of the system before the virus starts running, and we can quickly revert to the saved state once the analysis is complete [26].

Images depicting malware assaults, such as spear-phishing attacks, were used in another study to describe the timeline of the attack, with colors indicating which sorts of system connections were successful [27]. However, applying only one feature is insufficient for effective real-world malware detection or classification context since malware writers' obfuscation tactics may obscure a feature utilized in the machine learning model. Therefore, there is a need to develop algorithms to deal with a wide range of traits. Current techniques can be split into two groups depending on where the features are merged. Early or data-level fusion techniques merge many data sources into a single feature vector, subsequently fed into a machine learning algorithm [28, 29]. As n-gram systems need human counting of n-grams during training, a convolutional neural network-based technique reduces this need. N-gram-like signatures are alternatively learned using convolution. This approach avoids the conventional feature extraction pipeline, feature selection, reduction, and classification because both algorithms are immediately tweaked during training [30].

Despite the rising danger posed by Android malware, researchers still lack a comprehensive understanding of common behaviors and developing patterns across malware families operating on the platform. Without this method, researchers risk developing algorithms that identify just historical threats while ignoring the most current ones. The author [31] analyzed approximately 1.2 million malware samples from 1.28 K families over eight years, making it the most comprehensive research of Android malware behavior. The author's objective is to understand better how Android malware has grown, with a particular emphasis on malware repackaging. Many harmless programs are piggybacked with a malicious payload (rider) in this sort of attack, enabling low-cost malware manufacture. Slicing the software to determine which components are benign and malicious is one of the most time-consuming aspects of examining repackaged malware. The author employed differential analysis to isolate irrelevant software components from the campaign to overcome this issue, enabling him to focus only on the bad riders' behavior. The analytical approach is based on publicly available data repositories and recent improvements in the systematization of antivirus information obtained from several sources. According to the author, the Android malware ecosystem has significantly evolved since its inception in 2010, regarding the kind of destructive activities carried out by malware and the amount of obfuscation utilized to escape detection. Finally, the ramifications of the results are discussed for Android malware detection research, emphasizing areas where the research community should concentrate its efforts. The ridership of malware families, in particular, varies with time. This reveals a substantial experimental bias in research that use automated algorithms to identify families without accounting for variance.

While fast expansion indicates an ecosystem's health, it creates challenges for mobile software developers in terms of generating and maintaining high-quality products and customers are worried about the usability and security of emerging apps. In this context, it is vital to give valuable and

practical tools' help to mobile software developers that are informed and enabled by a full understanding of the ecosystem's evolutionary processes. The author [32] seeks to develop an architecture capable of systematically and continuously mining a mobile software ecosystem, emphasizing Android. Large-scale ecological longitudinal characterization research is conducted using this platform. To understand the ecosystem's evolutionary dynamics, the focus should be on the behavioral development patterns of ecosystem components such as mobile platforms, user applications developed on the platforms, and app users. Additionally, the characterization findings enable proactive app quality assurance and long-term app security. Additionally, the author examines risks and future steps and provides an update on this project with early data.

Given the frequent changes to the Android framework and the continued growth of Android malware, it is difficult to detect malware in a scalable and efficient manner over time. To address this issue, the author [33] presents DroidEvolver, an Android malware detection system that can automatically and continuously self-update while detecting malware without human intervention. While most existing malware detection systems can be updated by retraining on new applications with true labels, DroidEvolver can be updated without retraining or true labels, due to the insight that DroidEvolver updates itself via evolving feature sets and pseudolabels via online learning techniques. DroidEvolver's detection performance was examined using a dataset of 33,294 benign and 34,722 malicious apps created during six years. The F-measure of DroidEvolver is on average 2.19 times better than that of MAMADROID's state-of-the-art overtime malware detection system MAMADROID, and DroidEvolver's malware detection efficacy is 28.58 times more than that of MAMADROID. Additionally, DroidEvolver is immune to code obfuscation methods that are widely used.

Current malware detection methods for Android are dominated by machine learning-based categorization. On the other hand, current techniques are significantly constrained by their dependence on fresh malware samples that may not be instantly accessible and on ongoing retraining, which may be rather costly given the rapid growth of both the Android platform and its user apps. As a result, new and developing malware make their way through, as seen by the ongoing growth of malware in the wild. As a result, a more practical detector must be accurate on a subset of datasets and preserve its capabilities over time without needing retraining. The author [34] presents and investigates the sustainability issue for learning-based app classifiers in this study. This study establishes sustainability indicators for five cutting-edge Android malware detectors.

Additionally, the author created DroidSpan, a groundbreaking categorization technique for Android apps based on a unique behavioral profile that captures sensitive access distribution through lightweight profiling. The author compared the endurance of DroidSpan to five baseline detectors over eight years using longitudinal datasets. There were 13,627 benign programs and 12,755 malicious programs in the datasets. DroidSpan exceeded all baselines in terms of sustainability at a reasonable price by 6%–32 % for same-period detection and 21%–37 % for overtime detection, according to rigorous testing. The important takeaway, which also explains DroidSpan's success, is that learning-based malware detection requires the usage of persistent features that distinguish malware from benign apps over time, which may be uncovered via an app evolution study.

Machine learning techniques for Android malware detection must be periodically refreshed; otherwise, the trained classifier may be unable to distinguish newly found or developing malware kinds. This project aims to create a long-term Android malware detector that, once trained on a dataset, can detect new infections without requiring retraining. The author [35] examines how benign and malicious application behaviors evolve and determines the behavioral characteristics that consistently identify benign and malicious apps. The first results show that this approach has a promising future. On a seven-year benchmark, the proposed technique achieved exceptionally competitive detection accuracy for up to five years, outperforming the state-of-the-art, which lasted just two years.

Classification based on machine learning has long been a prominent way of malware protection. While there are many learning-based malware detection systems for Android, malicious applications continue to arise with increasing frequency in different Android app marketplaces. "How it is that new and developing malware can evade such a broad range of detection techniques?" the author [36] inquires in this regard. Intuitively, the performance deterioration of malware detectors may be the core reason for their failure to identify new infections after training on older samples. This research examines the degraded performance of four cutting-edge Android malware detectors to address the problem. The author verified that these present solutions significantly deteriorate and fast over time. The author introduces a unique categorization approach based on a long-term characterization study of Android apps focusing on dynamic behaviors. It compared our innovative technology to four current detectors and discovered considerable advantages for our new system. The key point is that studying app development over time may aid in detecting malware.

2.3. The Evolution in Malware Classification. Other systems had malware before 1986, but the PC had the first. Brain.A was the virus. Basit and Amjad, two Pakistani brothers, created it. They constructed a virus that replicated using floppy discs to demonstrate the PC's vulnerability. It infected the floppy drive's boot sector and every inserted floppy disc. One is the Omega virus. It was dubbed Omega because of the omega symbol engraved on the console. The Michelangelo virus rewrote the first 100 hard drive sectors in 1992. Walker, the next virus, arrived in 1992. It was an animated walker crossing the screen. The ambulance virus, like Walker, animated an ambulance vehicle traveling across the screen, but it also included sound effects [37, 38]. The Casino virus was one of the most intriguing viruses of the early 1990s. The Casino virus copies the file allocation table to memory and deletes the original. Then, it offers the user a slot game.

The history of malware may be divided into five categories, each corresponding to a historical period during which events in that category occurred. The first is malware development in its early stages. The first malware began to arise during this period. The early Windows period is the second category, while the evolution of network worms is the third. With the extensive use of the internet, malware has grown more common [39–41]. Rootkit and ransomware fall under the fourth group. Before 2010, this was the most hazardous evolution of malware. Rootkit and ransomware constitute the fourth category. Before 2010, this was the most destructive evolution of malware. We look at malware designed for virtual spying and disruption.

Some nations' spy agencies generated this virus. This is the newest stage of malware development [42, 43]. Malware production evolved from showmanship, vengeance, and profit to espionage and sabotage. Profit is still a driving force behind malware development and will be in the future. Malware authors have used espionage and sabotage for military goals. It is safe for attackers to employ and may do the same damage as military assaults with all its might [7, 13]. It remains to be seen how antivirus businesses respond to attackers with practically infinite resources for malware production and profit-driven malware developers. When it comes to military usage of malware, we may see more events like Stuxnet in the future. It remains to be seen how antivirus firms will cope with attackers with almost unlimited resources to create malware and those motivated only by profit [44, 45]. However, with occurrences like Stuxnet, we may see alternative uses for malware in the future and malware classification evolution is shown in Figure 1.

3. Methodology

This section explains how the deep learning algorithm is used to classify malware. For the classification issue with several classes, we provide a novel approach. The malware executable is first converted into grayscale graphics using our suggested technique. The photographs are fed into a fine-tuned deep learning model to identify and categorize the malware family. So, by placing the malware family, we get an idea about malware behaviors and types. In this way, it will be helpful for malware analysts to search for that specific behavior and generate antimalware. It is challenging to propose a wide-ranging malware classification system that can handle a massive quantity of malicious code and identify its family. In this section of the methodology, we discuss the steps that we performed to do this classification; the steps are as follows: data collection and preprocessing of the dataset, visualization of binary to a grayscale image, mode training, and model testing.

3.1. Data Acquisition. Microsoft is giving an unprecedented malware dataset to the data science community and promoting open-source progress on successful approaches for categorizing malware files into various families. Microsoft provided known malware files from nine distinct families that are included in this collection. Id, a 20-character hash value, and Class, a number, designate one of nine family names to which the malware belongs. Remnit, Lollipop, Kelihos ver3, Vundo, Simda, Tracur, Kelihos ver1, Obfuscator.ACYm, and Gatak are some of the several malware families.

3.2. Preprocessing of Data. Each file's binary information, excluding the PE header, is represented in hexadecimal form in raw data format. It is also possible to get a list of all metadata information collected from the binary, such as function calls and text values. The IDA disassembler was used to produce this. Malware dataset contains the following files.

Training File: - This file contains the raw data for the training set (MD5 hash = 4fedb0899fc2210a6c843889a70952ed).

Testing File: -This file contains the raw data for the test set (MD5 hash = 84b6fbfb9df3c461ed2cbbfa371ffb43).

3.3. Labeling. We need labeled samples because we are implementing supervised learning for classification. So, for marking these samples, we used the Microsoft Official dataset provided by Kaggle for competition. We have to supply the binary's MD5 hash for this.

TrainLabels.csv—the training set's class designations.

3.4. File Separation Process. In the dataset, we have the number of files having .asm as shown in Figure 2 and byte types as shown in Figure 3. We need to separate these files, so we use python language and PyCharm tool to code for this, and via code, we can split the byte and.asm files. We need the byte files for future use.

3.5. Byte to Image Conversion. Bytes files are included in our dataset. The raw data for each file provide the file's hexadecimal representation sans the file's PE header. So, initially, we transformed each hexadecimal representation into its decimal equivalent. There are 8-bit integers stored in a byte file. This one-dimensional vector can be easily turned into a two-dimensional array. An 8-bit grayscale picture with each pixel spanning from 0 to 8 may be easily viewed from 0 (black) to 255 (white) as shown in Figure 4.

3.6. Normalization of Image. Preprocessing the data before they are sent to the network is normalization. They are created by malware binaries and have no predetermined dimensions, making them difficult to categorize. To solve this issue, we initially reduced the size of the virus images to 224×224 pixels. Thus, the malware pictures were standardized and prepared to enter the CNN as input information. The key advantages of the normalizing method were that it reduced the size of the input photographs and made them more suitable for network training. The dimensionality reduction process also omitted several important features. We found that most malware pictures in our collection

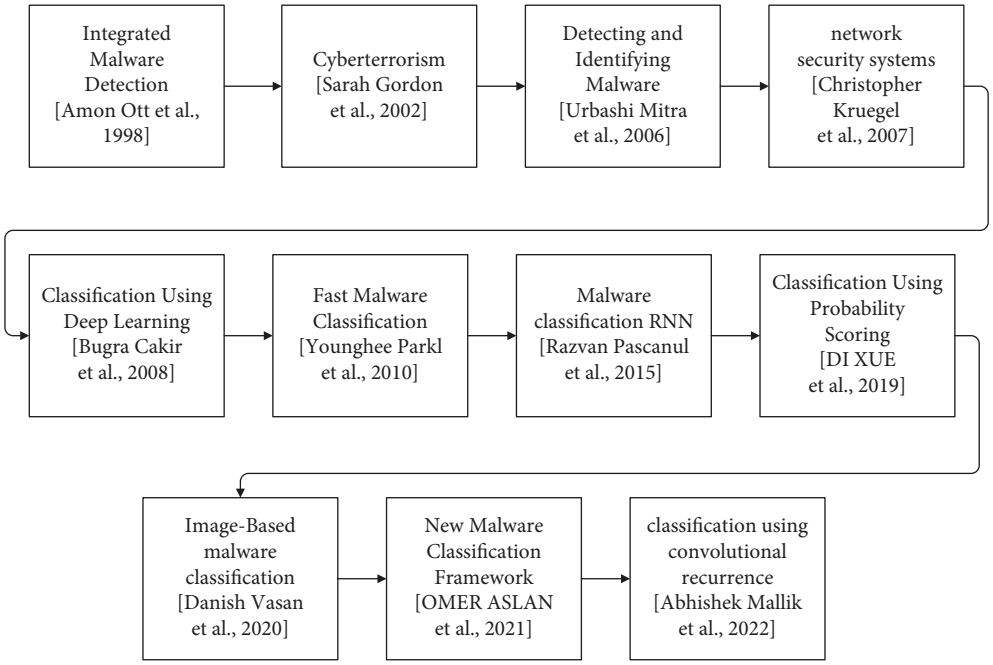
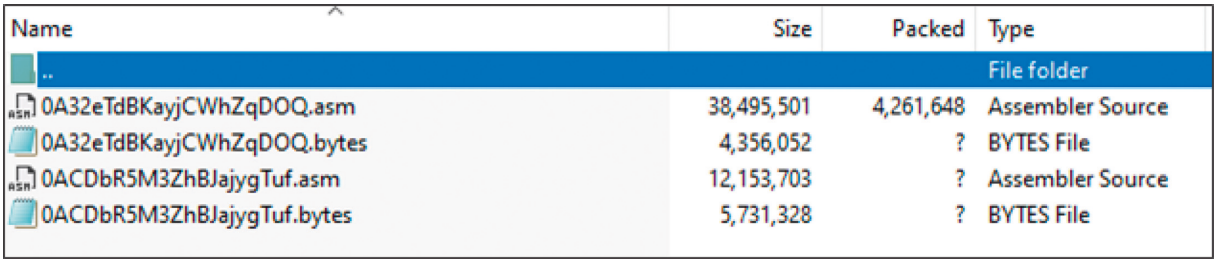


Figure 1: The evolution process in malware classifications.



Name	Size	Packed	Type
..			File folder
0A32eTdBKayjCWhZqDOQ.asm	38,495,501	4,261,648	Assembler Source
0A32eTdBKayjCWhZqDOQ.bytes	4,356,052	?	BYTES File
0ACDbR5M3ZhBJajygTuf.asm	12,153,703	?	Assembler Source
0ACDbR5M3ZhBJajygTuf.bytes	5,731,328	?	BYTES File

Figure 2: Original dataset file having both byte and .asm files.

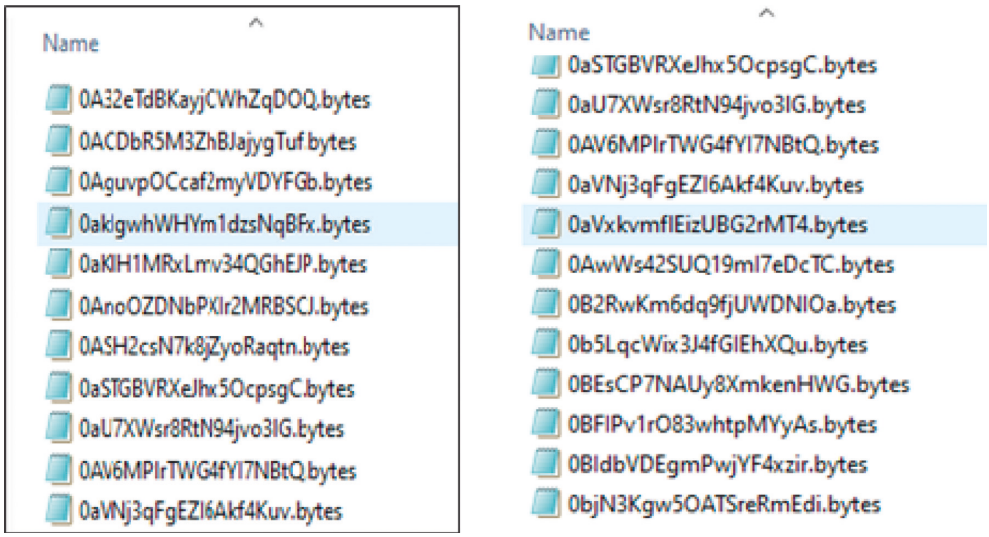


Figure 3: Byte files are taken from the dataset. Each file shows a different identifier, and it has its class label.

maintained their texture after the normalization process. Here are some image representations of malware from different families as shown in Figure 5.

3.7. Feature Extraction. This stage is critical for classifying deep learning models with the needed features. When dealing with enormous amounts of data, it is sometimes

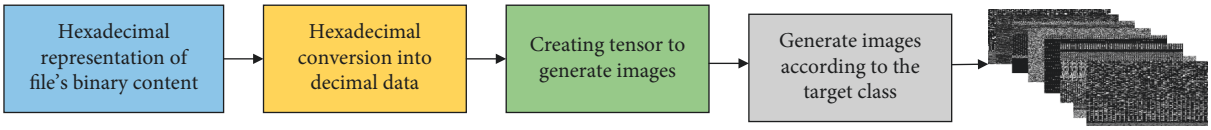


Figure 4: The image conversion process from hexadecimal binary content to 3-channel images.

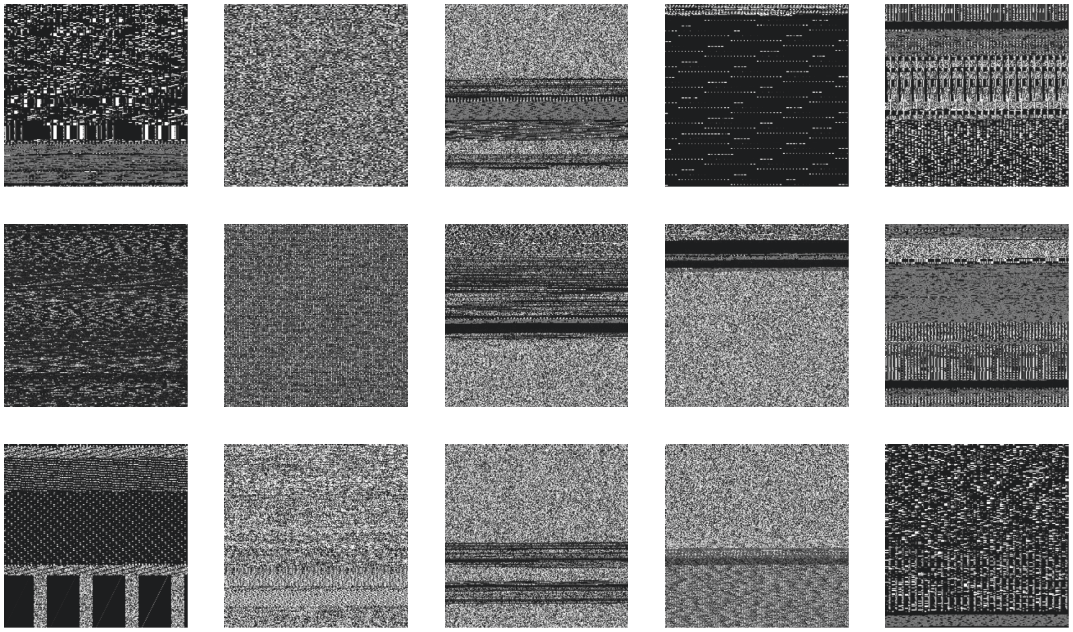


Figure 5: Malwares' families are represented in different image forms.

necessary to reduce the data to a more manageable number of feature representations. Resizing the image is part of data preprocessing. Apart from that, features can be extracted after and before data preprocessing according to the system design. Images are created from the binary file to the byte file and in the images. The image feature set can be desired while working on image data. Image tensors store a chunk of images for model training and testing phases. A feature contains information on an image's dimensions, texture, color, and shape. Local, global, and textural characteristics were all utilized in this case. A data stream is used to extract features in two steps in the proposed framework. Its durability and cheap processing cost make it an excellent first-stage tool for extracting texture features from grayscale photographs. Using the pretrained CNNs, a robust classification model is generated by extracting additional deep features.

3.8. Implementation. We adopt a different approach to analyzing and classifying malware than previous approaches. To address this issue, we turn to a convolutional neural network (CNN), a machine learning architecture that uses deep learning techniques, as shown in Figure 6. Deep learning has recently provided maximum efficiency across various applications and scenarios in numerous domains, including natural language processing, computer vision, speech recognition, and bioinformatics. However, using CNNs in many other fields has not been well investigated. Cyber security is one industry that could significantly benefit from developments in deep learning. With the recent success of deep understanding (particularly CNNs) in numerous classification tasks, we believe it can classify malware superior to support-vector machines in terms of accuracy. For image-processing issues, CNNs, in particular, have shown to be particularly successful.

For this reason, we turn malware classification into an image classification issue that can be tackled with CNNs. We design a generic malware classification architecture based on a deep convolutional neural network (CNN) instead of the current approaches. Existing high-accuracy approaches are frequently adapted to a given dataset. On the other hand, in the suggested process, the discriminative representation is directly learned from the data, rather than through handcrafted feature descriptors, making it data independent. In the first step, raw data for each file provide the file's binary content sans the PE header in hexadecimal. So, initially, we transformed each hexadecimal representation into its decimal equivalent. Using each unit's top and lower nibbles as indices when creating a two-dimensional color map, we may generate a series of RGB (pixel value) values for each hexadecimal digit. Image representations were then created by concatenating this sequence of pixel values to produce a two-dimensional matrix. A dataset containing photographs of malware is obtained in this way. For each image, it is adjusted to fit 224 rows of columns. A total of 10,000 samples

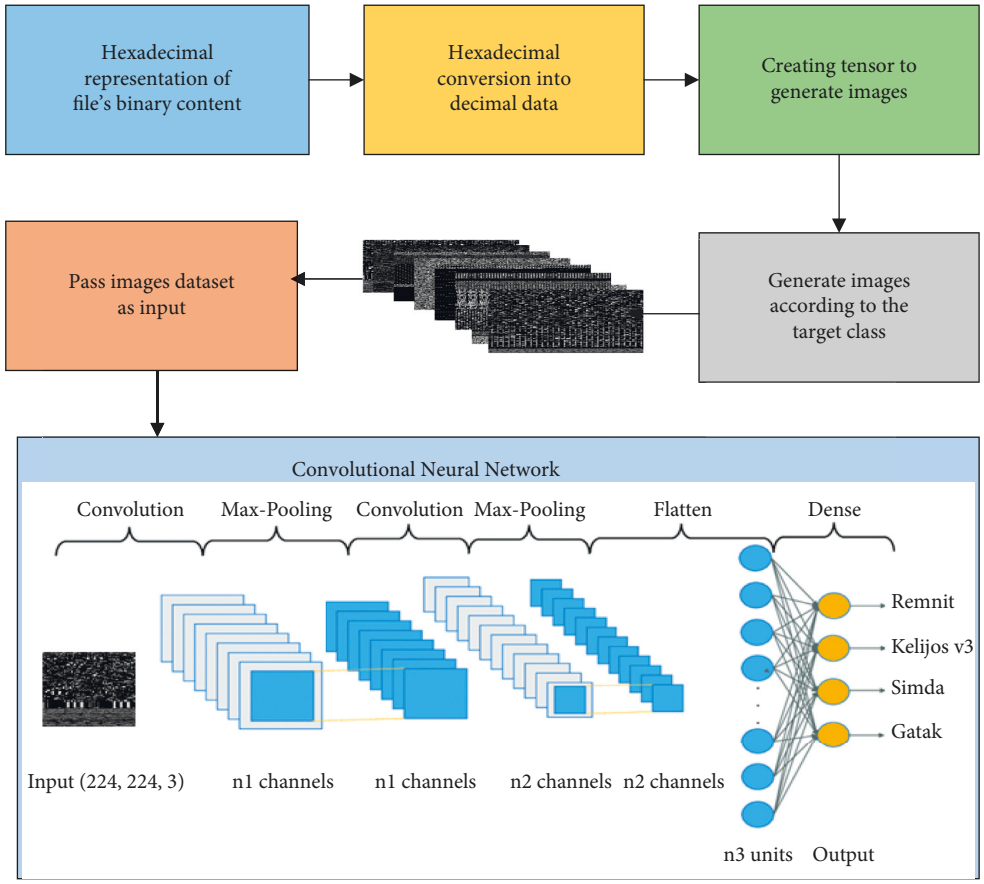


Figure 6: Proposed model.

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})}$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})}$$

$$\text{Accuracy} = \frac{(\text{TP}+\text{TN})}{(\text{TP}+\text{TN}+\text{FP}+\text{FN})}$$

Figure 7: Formulas of precision, recall, and accuracy.

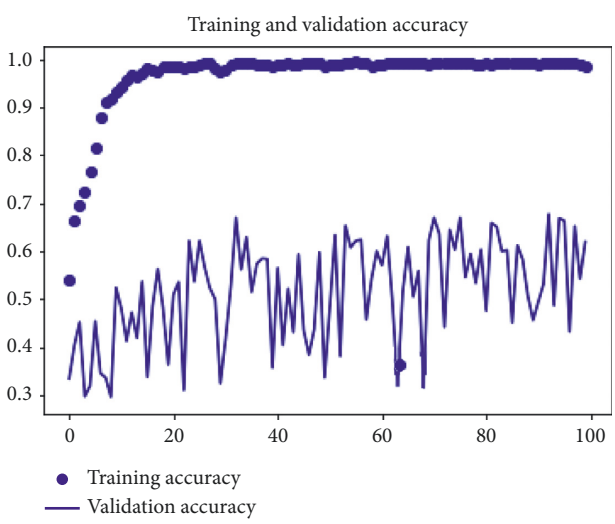


Figure 8: Training and validation accuracy.

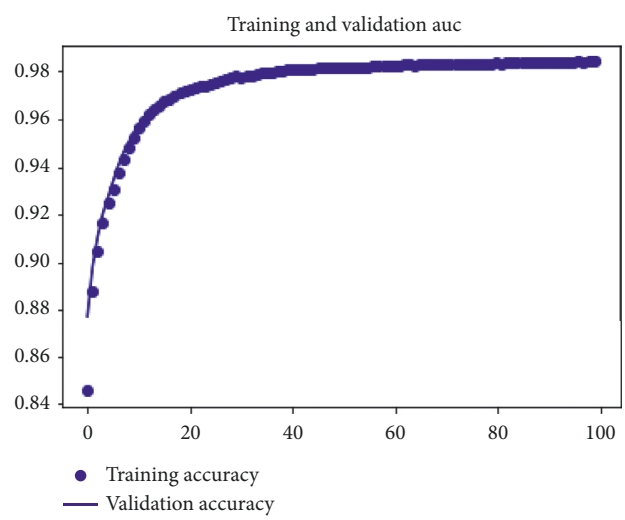


Figure 9: Training and validation of AUC.

were utilized for training and validation from a dataset made up of data from nine different classes. A twelve-layer residual network processes the training set samples. The model comprises two layers of convolution, followed by a layer of max pooling, followed by two layers of convolution, a layer of max pooling, and so on by adding flattened and dense layers.

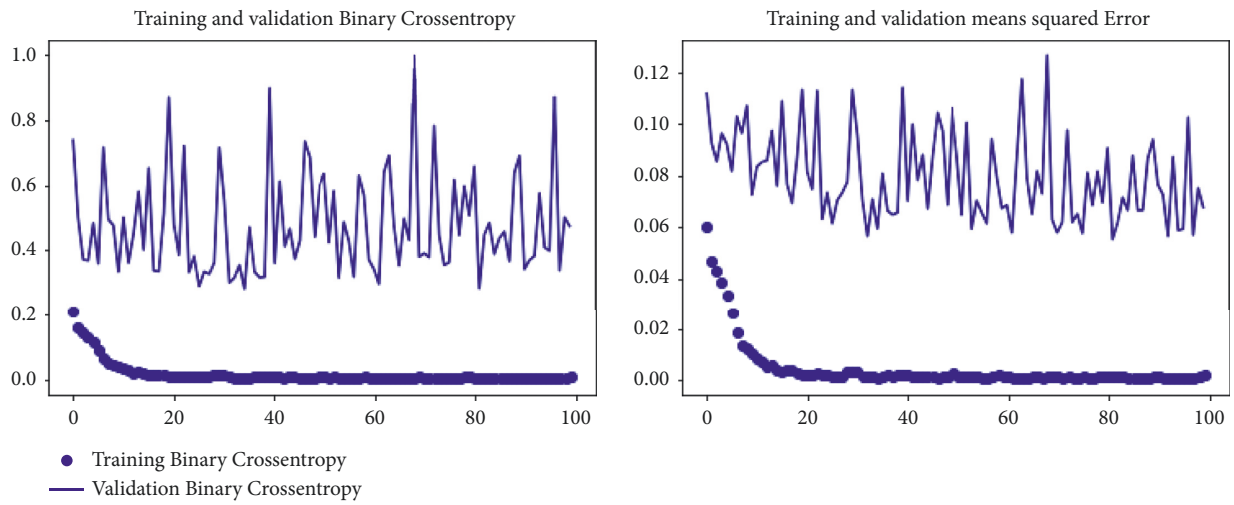


Figure 10: Training and validation binary cross-entropy and training and validation mean squared error.

4. Results and Discussion

Experimental results are shown in this section of the article. Texture-based malware classification is resistant to obfuscation methods and to enhance accuracy, according to the literature. Over 80% of malware binaries adopt this technique. The model is evaluated using a Microsoft Kaggle malware dataset, which was further converted into 224×224-pixel grayscale images from byte files using a deep convolutional neural network (CNN) approach. There were 100 epochs of training and testing with 9 classes of images with 10,000 samples, and our proposed model's accuracy was 99.97%. There are nine distinct malware programs in our dataset; however, the suggested study still outperforms them all.

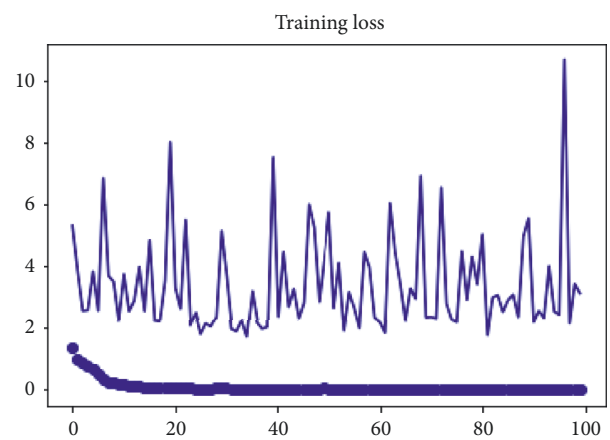


Figure 11: Training loss.

4.1. Evaluation Criterion. The experiments were implemented in Colab and Kaggle 64 bits on computer servers having 16 core CPU Xeon processors each of 3.2 GHz. 32 GB of RAM and 8 GB GPU. For the validation, an evaluation process of the implemented model testing process for the model is executed. The preliminary performance evaluation matrixes are the time and integrity of the predicted data. We will show the model training and testing evaluation based on accuracy, precision, and recall for both training and testing.

The performance of the prior system model is compared to our new, improved system scheme and machine learning model. The criteria listed below are used to assess the accuracy and other performance characteristics, as shown in Figure 7, where TP means true positive, FP means false positive, TN means true negative, and FN means false negative.

There were 100 epochs for the dataset that was used for training and testing. Figure 8 demonstrates the training and validation accuracy values, which are 99.97% as shown in the graph. The validity accuracy value grew in our model while the training loss value declined. In fact, depending on how some parameters are designed, different models may be able to increase verification capabilities. On the other hand, we have been unable to validate alternative models using Microsoft's dataset in our method.

Figure 8 depicts the validation performance results, and it shows that training and validation are almost identical curves for our model. Figure 9 shows that the former had a greater accuracy and a lower loss since it had a bigger number of samples. This finding demonstrates that severely lowering sample data by undersampling affects classification accuracy.

Figure 10 describes the binary cross-entropy results as shown in the below graph. After training the model on binary cross-entropy on the dataset, we validated the model, showing 99.97% results. Training and validation mean squared error is showing that our model performance is good on all 100 epochs.

Training loss is very small in our model, which works fine. In Figure 11, training loss is very small and validation is high, which evaluates our proposed model's accuracy and performance. The results of the experiments confirmed that our suggested technique is resistant to polymorphic obfuscation and that texture-based malware detection works.

Table 1: Validation of new algorithms against the Microsoft malware dataset.

Method	Epochs	Dataset	Accuracy
Vinayakumar et al. [46]	100	Microsoft malware dataset	91.27
Cui et al. [47]	10	Microsoft malware dataset	93.4
Luo and Lo [11]	60	Microsoft malware dataset	93.57
Singh et al. [48]		Microsoft malware dataset	94.24
Gilbert [49]	25	Microsoft malware dataset	94.64
Aslan et al. [44]		Microsoft malware dataset	94.88
Proposed method	100	Microsoft malware dataset	99.97

Table 1 provides the model's statistical values predicted during the evaluation process. Our proposed model achieved an accuracy value, which is 99.97% as compared to the other state-of-the-art methods.

5. Conclusions and Future Work

This manuscript explains the malware detection process by using deep learning techniques. Our proposed methods show the highest accuracy values. The accuracy comparison of the proposed model is the highest one, i.e., 99.97% compared to the already existing algorithms. In our model's simulation, the validity accuracy value increased, but the training loss value decreased. Because of this, various models may be able to improve their ability to verify the information. On the other hand, we have been unable to validate alternative models using the Microsoft dataset in our method. Even though we have seen some experimental proof of the recommended approach's effectiveness, more research in the following directions is required. We will improve a method for detecting and categorizing malware using GPUs and other parallelization methods to achieve high-performance computing. In the future, the large-scale application situations are implemented, the suggested approach. More research is needed to efficiently detect malware with antidissembling, antidebugging, and antipacking methods. We intend to investigate the fundamental reasons for the deteriorating problem in the future and build more effective malware detection tools.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was funded by the Taif University Researchers Supporting Project number (TURSP-2020/215), Taif University, Taif, Saudi Arabia.

References

[1] D. Gibert, C. Mateu, and J. Planes, "HYDRA: a multimodal deep learning framework for malware classification," *Computers & Security*, vol. 95, p. 101873, 2020.

[2] D. Farhat and M. S. Awan, "A brief survey on ransomware with the perspective of internet security threat reports," in *Proceedings of the 2021 9th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1–6, Elazig, Turkey, June 2021.

[3] B. Yadav and S. Tokekar, "Recent innovations and comparison of deep learning techniques in malware classification: a review," *International Journal of Information Security Science*, vol. 9, pp. 230–247, 2021.

[4] R. Agrawal, J. W. Stokes, K. Selvaraj, and M. Marinescu, "Attention in recurrent neural networks for ransomware detection," in *Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3222–3226, Brighton, UK, May 2019.

[5] U. A. Butt, M. Mehmood, S. B. H. Shah et al., "A review of machine learning algorithms for cloud computing security," *Electronics*, vol. 9, no. 9, p. 1379, 2020.

[6] H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, "Smart home security: challenges, issues and solutions at different IoT layers," *The Journal of Supercomputing*, vol. 77, no. 12, pp. 14053–14089, 2021.

[7] D. Vasan, M. Alazab, S. Wassan, B. Safaei, and Q. Zheng, "Image-Based malware classification using ensemble of CNN architectures (IMCEC)," *Computers & Security*, vol. 92, p. 101748, 2020.

[8] M. Jain, W. Andreopoulos, and M. Stamp, "Convolutional neural networks and extreme learning machines for malware classification," *Journal of Computer Virology and Hacking Techniques*, vol. 16, no. 3, pp. 229–244, 2020.

[9] R. Mitsuhashi and T. Shinagawa, "High-accuracy malware classification with a malware-optimized deep learning model," 2020, https://arxiv.org/abs/2004.05258.

[10] M. Nisa, J. H. Shah, S. Kanwal et al., "Hybrid malware classification method using segmentation-based fractal texture analysis and deep convolution neural network features," *Applied Sciences*, vol. 10, no. 14, p. 4966, 2020.

[11] J.-S. Luo and D. C.-T. Lo, "Binary malware image classification using machine learning with local binary pattern," in *Proceedings of the 2017 IEEE International Conference on Big Data (Big Data)*, pp. 4664–4667, Boston, MA, USA, December 2017.

[12] A. Dey, S. Bhattacharya, and N. Chaki, "Byte label malware classification using image entropy," in *Advanced Computing and Systems for Security*, pp. 17–29, Springer, Berlin, Germany, 2019.

[13] D. Xue, J. Li, T. Lv, W. Wu, and J. Wang, "Malware classification using probability scoring and machine learning," *IEEE Access*, vol. 7, pp. 91641–91656, 2019.

[14] S. Kumar, "MCFT-CNN: malware classification with fine-tune convolution neural networks using traditional and transfer learning in internet of things," *Future Generation Computer Systems*, vol. 125, pp. 334–351, 2021.

[15] P. Prajapati and M. Stamp, "An empirical analysis of image-based learning techniques for malware classification," in *Malware Analysis Using Artificial Intelligence and Deep Learning*, pp. 411–435, Springer, Berlin, Germany, 2021.

[16] S. Z. M. Shaid and M. A. Maarof, "Malware behaviour visualization," *Jurnal Teknologi*, vol. 70, 2014.

[17] H. Naeem, B. Guo, and M. R. Naeem, "A light-weight malware static visual analysis for IoT infrastructure," in *Proceedings of the 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pp. 240–244, Chengdu, China, May 2018.

[18] D. Vasan, M. Alazab, S. Wassan, H. Naeem, B. Safaei, and Q. Zheng, "IMCFN: image-based malware classification using fine-tuned convolutional neural network architecture," *Computer Networks*, vol. 171, p. 107138, 2020.

[19] K. Barure, Z. Shaikh, S. More, S. Kalbhor, and Y. Ingle, *Malware Classification Using Deep Learning*, Spriger, Berlin, Germany, 2020.

[20] L. Nataraj, V. Yegneswaran, P. Porras, and J. Zhang, "A comparative assessment of malware classification using binary texture analysis and dynamic analysis," in *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*, pp. 21–30, Chicago Illinois USA, October 2011.

[21] K. Han, B. Kang, and E. G. Im, "Malware analysis using visualized image matrices," *TheScientificWorldJOURNAL*, vol. 2014, p. 132713, 2014.

[22] L. Ghouti and M. Imam, "Malware classification using compact image features and multiclass support vector machines," *IET Information Security*, vol. 14, no. 4, 2020.

[23] D. L. Vu, T. K. Nguyen, T. V. Nguyen, T. N. Nguyen, F. Massacci, and P. H. Phung, "HIT4Mal: hybrid image transformation for malware classification," *Transactions on Emerging Telecommunications Technologies*, vol. 31, p. e3789, 2020.

[24] A. McDole, M. Gupta, M. Abdelsalam, S. Mittal, and M. Alazab, "Deep learning techniques for behavioral malware analysis in cloud iaas," in *Malware Analysis Using Artificial Intelligence and Deep Learning*, pp. 269–285, Springer, Berlin, Germany, 2021.

[25] J. S. Sraw, "Using static and dynamic malware features to perform malware ascription," *SPAST Abstracts*, vol. 1, 2021.

[26] S. Yoo, S. Kim, S. Kim, and B. B. Kang, "AI-HydRa: advanced hybrid approach using random forest and deep learning for malware classification," *Information Sciences*, vol. 546, pp. 420–435, 2021.

[27] P. Wang, Z. Tang, and J. Wang, "A novel few-shot malware classification approach for unknown family recognition with multi-prototype modeling," *Computers & Security*, vol. 106, Article ID 102273, 2021.

[28] M. Usman, R. Amin, H. Aldabbas, and B. Alouffi, "Lightweight challenge-response authentication in SDN-based UAVs using elliptic curve cryptography," *Electronics*, vol. 11, no. 7, p. 1026, 2022.

[29] N. Usman, S. Usman, F. Khan et al., "Intelligent dynamic malware detection using machine learning in IP reputation for forensics data analytics," *Future Generation Computer Systems*, vol. 118, pp. 124–141, 2021.

[30] A. McDole, A. Brown, and P. Brown, "Malware classification using deep learning in cloud environments," *Proceedings of Student Research and Creative Inquiry Day*, vol. 5, 2021.

[31] G. Suarez-Tangil and G. Stringhini, "Eight years of rider measurement in the android malware ecosystem: evolution and lessons learned," 2018, https://arxiv.org/abs/1801.08115.

[32] H. Cai, "Embracing mobile app evolution via continuous ecosystem mining and characterization," in *Proceedings of the IEEE/ACM 7th International Conference on Mobile Software Engineering and Systems*, pp. 31–35, Seoul Republic of Korea, July 2020.

[33] K. Xu, Y. Li, R. Deng, K. Chen, and J. Xu, "Droidevolver: self-evolving android malware detection system," in *Proceedings of the 2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 47–62, Stockholm, Sweden, June 2019.

[34] H. Cai, "Assessing and improving malware detection sustainability through app evolution studies," *ACM Transactions on Software Engineering and Methodology*, vol. 29, no. 2, pp. 1–28, 2020.

[35] H. Cai and J. Jenkins, "Towards sustainable android malware detection," in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings*, pp. 350-351, Gothenburg Sweden, May 2018.

[36] X. Fu and H. Cai, "On the deterioration of learning-based malware detectors for Android," in *Proceedings of the 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, pp. 272-273, Montreal, QC, Canada, May 2019.

[37] A. Ott, S. Fischer-Hübner, and M. Swimmer, "Approaches to integrated malware detection and avoidance," in *Proceedings of the 3rd Nordic Workshop on Secure IT Systems*, Trondheim, 1998.

[38] S. Gordon and R. Ford, "Cyberterrorism?" *Computers & Security*, vol. 21, no. 7, pp. 636–647, 2002.

[39] U. Mitra, A. Ortega, J. Heidemann, and C. Papadopoulos, "Detecting and identifying malware: a new signal processing goal," *IEEE Signal Processing Magazine*, vol. 23, no. 5, pp. 107–111, 2006.

[40] M. Bailey, J. Oberheide, J. Andersen, Z. M. Mao, F. Jahanian, and J. Nazario, "Automated classification and analysis of internet malware," in *Proceedings of the International Workshop on Recent Advances in Intrusion Detection*, pp. 178–197, Berlin, Germany, 2007.

[41] Y. Park, D. Reeves, V. Mulukutla, and B. Sundaravel, "Fast malware classification by automated behavioral graph matching," in *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research*, pp. 1–4, Oak Ridge Tennessee USA, April 2010.

[42] R. Pascanu, J. W. Stokes, H. Sanossian, M. Marinescu, and A. Thomas, "Malware classification with recurrent networks," in *Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1916–1920, South Brisbane, QLD, Australia, pril 2015.

[43] B. Cakir and E. Dogdu, "Malware classification using deep learning methods," in *Proceedings of the ACMSE 2018 Conference*, pp. 1–5, Richmond Kentucky, March 2018.

[44] Ö. Aslan and A. A. Yilmaz, "A new malware classification framework based on deep learning algorithms," *IEEE Access*, vol. 9, 2021.

[45] A. Mallik, A. Khetarpal, and S. Kumar, "ConRec: malware classification using convolutional recurrence," *Journal of Computer Virology and Hacking Techniques*, pp. 1–17, 2022.

[46] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, "Robust intelligent malware detection

using deep learning," *IEEE Access*, vol. 7, pp. 46717–46738, 2019.

[47] Z. Cui, F. Xue, X. Cai, Y. Cao, G.-g. Wang, and J. Chen, "Detection of malicious code variants based on deep learning," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3187–3196, 2018.

[48] A. Singh, A. Handa, N. Kumar, and S. K. Shukla, "Malware classification using image representation," in *Proceedings of the International Symposium on Cyber Security Cryptography and Machine Learning*, pp. 75–92, June 2019.

[49] D. Gibert, *Convolutional Neural Networks for Malware Classification*, University Rovira i Virgili, Tarragona, Spain, 2016.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3502992, 8 pages
https://doi.org/10.1155/2022/3502992

Research Article

Recommendation System for Privacy-Preserving Education Technologies

Shasha Xu and Xiufang Yin

Zhengzhou Preschool Education College, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Shasha Xu; xushasha1984@126.com

Received 2 March 2022; Revised 7 March 2022; Accepted 17 March 2022; Published 16 April 2022

Academic Editor: Konstantinos Demertzis

Considering the priority for personalized and fully customized learning systems, the innovative computational intelligent systems for personalized educational technologies are the timeliest research area. Since the machine learning models reflect the data over which they were trained, data that have privacy and other sensitivities associated with the education abilities of learners, which can be vulnerable. This work proposes a recommendation system for privacy-preserving education technologies that uses machine learning and differential privacy to overcome this issue. Specifically, each student is automatically classified on their skills in a category using a directed acyclic graph method. In the next step, the model uses differential privacy which is the technology that enables a facility for the purpose of obtaining useful information from databases containing individuals' personal information without divulging sensitive identification about each individual. In addition, an intelligent recommendation mechanism based on collaborative filtering offers personalized real-time data for the users' privacy.

1. Introduction

Artificial intelligence-based educational techniques have advanced significantly in recent years, and their applications in various academic fields have increased. Implementing artificial intelligence in education encompasses a broad range of intelligent instructional and evaluation methods, including intelligent tutoring systems, intelligent performance assessment, intelligent virtual agents, talking robots, humanized chatbots, and any other approach based on artificial intelligence [1]. These classroom innovations can benefit a diverse range of students, particularly those with disabilities. Thanks to new intelligence technologies, these students now have a more flexible and personalized educational solution.

In general, artificial intelligence can be combined with other methods (e.g., speech recognition, machine vision, and disability assistant) to develop advanced tutor systems that can help students learn more effectively [2]. Furthermore, approaches based on artificial intelligence can be used to create adaptive and personalized learning systems that are tailored to the unique characteristics of each individual student.

Nevertheless, as AI models reflect the data over which they were trained, data that may have privacy or other sensitivities associated with it, they are vulnerable. This work proposes a privacy-preserving [3] recommendation system that uses differential privacy in this spirit. Differential privacy [4] is a technology that allows researchers and database analysts to acquire useful information from databases that contain people's personal information without disclosing the unique identify of the persons who have provided the information. Achieving this can be accomplished by including the bare minimum of distractions in the information provided by the database system. The amount of distraction introduced is significant enough to protect privacy while still allowing for the provision of information to analysts to continue to be valid. Differential privacy, in its most basic sense, is the process of forming data anonymously by deliberately adding noise into a dataset. Data analysts are capable of doing any and all possible (functional) statistical analysis without revealing any personal information.

Specifically, this study presents an innovative privacy-preserving recommendation system for educational technologies. It is a fully automated intelligent system that can categorize trainees based on their requirements and special skills. The abilities of each student are automatically categorized into one of several categories. Using a directed acyclic graph machine learning method, the model uses differential privacy in order to protect the private information of each individual learner. Also, an intelligent module based on collaborative filtering offers personalized real-time privacy recommendations.

Afterward, in Section 2, we learn about the proposed system's technique. Exemptions for applying the proposed method are outlined in Section 3. Section 4 concludes by summarizing the findings and drafting the following potential directions for the work.

2. Proposed Methodology

A directed acyclic graph (DAG) [5] is used to express a probabilistic representation of the data structure created from the model and their putative independence. The classification method is then utilized to validate the whole combined probability distributions in the DAG [5]. The goal is to categorize an X sample into one of the supplied categories $C_1, C_2, \ldots, C_n$ using a probability model constructed according to Bayes theory in order to get the desired result. Overall, this is a first-level classification based on probabilities rather than predictions, a fact that has been demonstrated experimentally to be more useful, faster, and more efficient. In this case, projections are made to a certain extent, and the goal is to keep costs as low as possible. Each category is distinguished by a probability distribution that has occurred in the past. We assume that the sample X belongs to a class Ci, and we calculate the probability [5, 6] using the definitions and Bayes theory, respectively. To put it in another way, the initial step in the procedure is to understand how the pupils are dependent on one another and then assign probabilities to them, insuring how likely it is that their ability will change over time. As a result, the proposed system incorporates prior knowledge gathered from the model into the model learning process through a probabilistic representation of the data structure that arises for each learner, hence, enhancing the overall effectiveness of the system. A further consideration is the uncertainty in the model parameters that have been generated, which may be caused by noise such as a random or deceptive evaluation procedure, among other things.

In order to assess the overall performance, the following criteria of the DAG algorithm were used [7–9]:

(1) Overall accuracy (OvAc): this metric reflects the proportion of correctly identified samples in relation to the total number of test samples in a given period.

(2) Average accuracy (AvAc): this indicator displays the average accuracy of the different categories.

(3) Kappa rate: using the following function, we can determine how well the truth map and the final categorization map agree on various statistical criteria:

$$K = \frac{p_0 - p_e}{1 - p_e} = 1 - \frac{1 - p_0}{1 - p_e}, \tag{1}$$

where p_o is the correlation between actual agreement and p_e is the theoretical likelihood of random agreement.

(4) McNemar test: to evaluate the significance of categorization accuracy derived from different methodologies, a McNemar test was used:

$$z_{12} = \frac{f_{12} - f_{21}}{\sqrt{f_{12} + f_{21}}}, \tag{2}$$

where f_{ij} samples accurately categorized in classification and i mistakenly classified in the other one j.

(5) Coefficient of determination, R^2: use it to express correlation between two variables in percentage terms. The coefficient of determination is a measure of the degree to which the values of X and Y are correlated and calculated as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(Y_i - \widehat{Y}_i\right)^2}{\sum_{i=1}^{n} \left(Y_i - \underline{Y}_i\right)^2}, \tag{3}$$

where Y_i are the actual values of the dependent variable, $\widehat{Y}_i$ have been calculated based on our best estimates for this dependent variable, and Y is computed by taking the observed data and averaging it the number of observations.

(6) Root relative squared error (RRSE): in order for a model to be considered successful, the absolute correlation between predicted and actual values must be equal to zero:

$$\text{RRSE} = \frac{\sum_{j=1}^{n} \left(P_{(ij)} - T_j\right)^2}{\sum_{j=1}^{n} \left(T_j - \underline{T}\right)^2}, \tag{4}$$

where $P_{(ij)}$ is the anticipated value for a simple hypothesis that the algorithm generates j and T_j and T are the desired value for the simple hypothesis j, with the following connection being used to determine:

$$\underline{T} = \frac{1}{n} \sum_{j=1}^{n} T_j. \tag{5}$$

In addition, the proposed method uses differential privacy. Before being shared through the suggested technique, personal data might be obscured by statistical noise that has been slanted in a certain direction. It is possible to see relevant information emerge when a huge number of people contribute the same information. Three ingredients—sensitive

data, curators who need to provide statistics, and adversaries who want to retrieve the sensitive data—can all be solved through differential privacy. This reverse engineering is a type of privacy breach [3, 4, 10].

Finally, an intelligent recommendation memory-based approach was used to measure user privacy and compute the similarity between users [11, 12]. Finding persons with similar interests may be accomplished using the locality-sensitive hashing, which utilizes the closest neighbor algorithm in a linear time frame. A set of privacy restrictions is then proposed based on the k most comparable users and their related user-item matrices. Easy construction and usage, easy facilitation of new data, content-independent of the items being recommended, and effective scalability with co-rated goods are some of the advantages that this technique has to offer [13].

An abstract illustration of the proposed architecture is presented in Figure 1, which depicts as parts of a flowchart the basic steps of how the proposed system works.

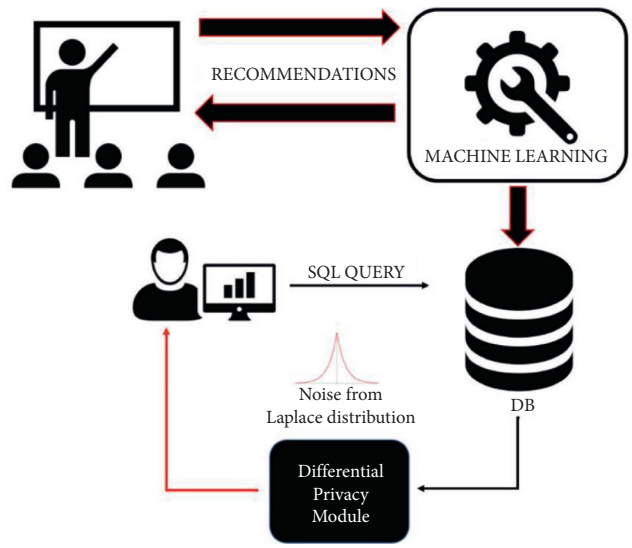


Figure 1: The proposed architecture.

3. Experiments

A preliminary exam for categorizing pupils' ability in their various level departments is the subject of this scenario. Students take this simple examination to determine their fitness to continue in higher-level education studies. It includes a set of questions or exercises evaluating skill or knowledge based on a scientific standard that can identify the real learning abilities of each learner [14–16]. Specifically, the preliminary test includes psychometric questionnaires and the purpose is to detect misunderstandings, ambiguities, disabilities, or other learning difficulties that may have the students.

The outcomes of this preliminary test are the dataset used by the classification algorithm. The dataset is used to contain ten questions that come from 350 volunteer students. Table 1 presents the statistical analysis of the preliminary test used in this study.

The questionnaire is satisfactorily reliable in measuring the determination of students' moods and corresponding abilities and can be used for further processing by the proposed learning system [17, 18].

3.1. Step 1: Classification Process and Results. This model's probabilistic values and the abovementioned statistical analysis of the questions [19] map each student's reply to the DAG as a pair of variables based on these criteria [6] in form $B = \langle G, \Theta \rangle$, where G is the nodes $X_1, X_2, \ldots, X_n$. In this form, each question in the questionnaire is represented as a probability value, along with its corresponding edge (the answers to each question). Graph G conveys the assumption that each variable X_i is independent of the inheritance assumed by G. Θ identifies the parameters of the network. Specifically, this set contains the parameter $\theta_{x_i|\pi_i} = P_B(x_i \mid \pi_i)$ for each x_i implementation of X_i in the condition π_i, for the set of X_i parents in G. Therefore, B defines a unique probability distribution over the variables, namely [5, 7, 20],

$$P_B = (X_1, X_2, \ldots, X_n). = \prod_{i=1}^{n} P_B(\pi_i) = \prod_{i=1}^{n} \theta_{X_i} \mid \pi_i \Big). \tag{6}$$

There are three internally distinct paths linking two vertices u and v such that neither of them has the same orientation, or there are two directed cycles with a common vertex if there is a strong component, that is, neither a cycle or a single vertex. Number of predicted components is capped above [5, 20, 21]:

$$2\binom{n}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\binom{n}{i}i!p^{i+1}\binom{n}{j}j!p^{j+1}\binom{n}{k}k!p^{k+1} + \binom{n}{1}\sum_{i=2}^{n}\sum_{j=2}^{n}\binom{n}{i}i!p^{i+1}\binom{n}{j}j!p^{j+1} \le \frac{\lambda^3}{n}\sum_{i=0}^{\infty}\sum_{j=0}^{\infty}\sum_{k=0}^{\infty}\lambda^{i+j+k} + \frac{\lambda^2}{n}\sum_{i=0}^{\infty}\sum_{j=0}^{\infty}\lambda^{i+j} = O\left(n^{-1}\right). \tag{7}$$

Table 1: Statistical analysis of the preliminary test.

Quest	Mean	S	Sδ	r_δ	R^2	Cronbach a
Q1	3.425	1.659	5.456	0.799	0.887	0.879
Q2	3.376	1.544	5.433	0.711	0.806	0.806
Q3	3.125	1.355	5.562	0.798	0.890	0.811
Q5	2.788	1.678	6.226	0.542	0.651	0.870
Q7	3.115	1.454	5.987	0.794	0.874	0.806
Q8	3.089	1.599	5.998	0.789	0.799	0.798
Q9	3.341	1.473	5.887	0.801	0.888	0.783
Q10	3.184	1.932	5.752	0.732	0.801	0.797

Based on the Markov inequality, there are no such components. So, we can bound the expected number of cycles of length larger than ω by

$$\sum_{k=\omega}^{n}\binom{n}{k}(k-1)!p^k = \sum_{k=\omega}^{n}\frac{\prod_{i=0}^{k-1}(n-i)}{n^k\,\dfrac{\lambda^k}{k}\leq\sum_{k=\omega}^{n}\lambda^k = O(\lambda^\omega).} \tag{8}$$

To compute the expectation of X_n, we have

$$\mathbb{E}[X_n] = \sum_{k=3}^{n}\binom{n}{k}(k-1)!p^k. \tag{9}$$

It follows that

$$\lim_{n\longrightarrow\infty}\mathbb{E}(X_n) = \lim_{n\longrightarrow\infty}\sum_{k=3}^{n}\frac{\prod_{i=0}^{k-1}(n-i)}{n^k\,\dfrac{\lambda^k}{k}\sim\sum_{k=3}^{\infty}\dfrac{\lambda^k}{k} = -\log(1-\lambda)-\lambda-\dfrac{\lambda^2}{2} = a(\lambda).} \tag{10}$$

The rth factorial moment of X_n is

$$\mathbb{E}[(X_n)_r] = \widehat{\theta}\sum_{k_1=3}^{n}\sum_{k_2=3}^{n-k_1}\cdots\sum_{k_r=3}^{n-\sum_{i=1}^{r-1}k_i}\binom{n}{k_1,k_2,\ldots,k_r,n-k_1-\ldots-k_r}\prod_{i=1}^{r}(k_i-1)!p^{k_i}. \tag{11}$$

The Hessian matrix of second-order partial and cross-partial derivatives determines whether or not the likelihood equations indicated root is in fact a (local) maximum [21, 22]:

$$H(\widehat{\theta}) = \begin{bmatrix} \dfrac{\partial^2\ell}{\partial\theta_1^2}\Big|_{\theta=\widehat{\theta}} & \dfrac{\partial^2\ell}{\partial\theta_1\partial\theta_2}\Big|_{\theta=\widehat{\theta}} & \cdots & \dfrac{\partial^2\ell}{\partial\theta_1\partial\theta_k}\Big|_{\theta=\widehat{\theta}} \\ \dfrac{\partial^2\ell}{\partial\theta_2\partial\theta_1}\Big|_{\theta=\widehat{\theta}} & \dfrac{\partial^2\ell}{\partial\theta_2^2}\Big|_{\theta=\widehat{\theta}} & \cdots & \dfrac{\partial^2\ell}{\partial\theta_2\partial\theta_k}\Big|_{\theta=\widehat{\theta}} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2\ell}{\partial\theta_k\partial\theta_1}\Big|_{\theta=\widehat{\theta}} & \dfrac{\partial^2\ell}{\partial\theta_k\partial\theta_2}\Big|_{\theta=\widehat{\theta}} & \cdots & \dfrac{\partial^2\ell}{\partial\theta_k^2}\Big|_{\theta=\widehat{\theta}} \end{bmatrix}. \tag{12}$$

In order to optimize the problem, we use bordered Hessian:

$$\begin{aligned} H(\Lambda) &= \begin{bmatrix} \dfrac{\partial^2\Lambda}{\partial\lambda^2} & \dfrac{\partial^2\Lambda}{\partial\lambda\,\partial x} \\ \left(\dfrac{\partial^2\Lambda}{\partial\lambda\,\partial x}\right)^{\top} & \dfrac{\partial^2\Lambda}{\partial x^2} \end{bmatrix} \\ &= \begin{bmatrix} 0 & \dfrac{\partial g}{\partial x} \\ \dfrac{\partial g}{\partial x}\Big)^{\top} & \dfrac{\partial^2\Lambda}{\partial x^2} \end{bmatrix}. \end{aligned} \tag{13}$$

Table 2 presents the results of the classification process:

For each variable (response), a probability value is generated, revealing the degree to which it is interdependent with its class and hence the direction in which each question has an effect. In other words, a first classification of the responses into distinct categories can define the options and skills of each student. In this example, based on the

Table 2: Classification results.

	OvAc (%)	AvAc (%)	Kappa	McNemar	R^2	RRSE
Class_1	99.44	98.67	0.8992	30.172	0.989	0.0459
Class_2	98.37	97.52	0.8885	29.674	0.981	0.0518
Class_3	99.12	98.33	0.8973	30.029	0.987	0.0479

questionnaire, three classes were used (theoretical direction, positive direction, and technological direction), where the students were classified based on their answers and the algorithm of the DAG used.

3.2. Step 2: Differential Privacy and Results. On the contrary, in order to protect an individual who is deciding to allow their data to be included in the repositories that proposed the method, we use differential privacy. Let q be a counting query. Trying to protect privacy by adding noise results in [3, 23–25],

$$M(x) = q(x) + \text{noise}. \tag{14}$$

The Laplace distribution with scale parameter $b > 0$ (assuming position parameter 0) is defined as the distribution with probability density function:

$$\text{Lap}(x \mid b) = \frac{1}{2b} e^{-|x|/b}. \tag{15}$$

So, it turns out

$$\begin{aligned} \Pr[M(x) \in S] &= \Pr[M(x) \in S|\text{enoughnoise}] \\ &\quad + \Pr[M(x) \in S|\text{notenoughnoise}] \\ &\le e^{\varepsilon} \Pr[M(y) \in S] + \delta. \end{aligned} \tag{16}$$

The l 1-sensitivity of a function f is calculated as

$$\Delta = \max_{x \sim x'} f(x) - f(x'). \tag{17}$$

For example, compare the x database with the test scores and the query for the average score:

$$q(x) = \frac{\sum_{i=1}^{n} x_i}{n}. \tag{18}$$

If we use a neighborhood type relationship such as $|x_i - x_i'| \le x_{\max}$, then the sensitivity of the question will be

$$\begin{aligned} \Delta &= \max_{x \sim x'} |q(x) - q(x')| \\ &= \frac{1}{n} \max_{x_i, x_i'} |x_i - x_i'| i \in [1, n]. \end{aligned} \tag{19}$$

According to the above equation, the differential privacy mechanism will be

$$M(x) = \frac{\sum_{i=1}^{n} x_i}{n} + \text{Lap}\left(\frac{x_{\max}}{n \in}\right). \tag{20}$$

Finally,

$$\begin{aligned} Pr[\text{notenoughnoise}] &\le Pr[\text{Lap}(\mu, b) \le 2] \\ &\quad + Pr[\text{Lap}(\mu, b) \le 1] = \\ &= \frac{1}{2} \exp\left(\frac{2-\mu}{b}\right) + \frac{1}{2} \exp\left(\frac{1-\mu}{b}\right) \\ &\le \exp\left(\frac{2-\mu}{b}\right). \end{aligned} \tag{21}$$

To prove that the proposed differential privacy system is secure against level 2 attacks, we need to prove that it does not allow distance calculation. Specifically, assuming that a DRE E is used to encrypt the DB to get E(DB), a level 2 attacker with $H = \langle$E (DB), P, I$\rangle$ can retrieve DB if P contains at least $d+1$ points x_i $(1 \le i \le d+1)$ so that the set of vectors $\{x_j - x_1 | 2 \le j \le d+1\}$ is linearly independent. A hash function used by the Distributed Hash Table (DHT) to assign file ownership to network nodes which generates a key of 256 bits, which is enough to withstand the level 2 attack on the DRE. This system's encryption function hides the distance between two points in a database table; therefore, it must be determined which of the two points is closest to a query point q, and it must also be implemented [4, 26]:

$$d(p1, q) \ge d(p2, q) \sqrt{\|p1\|^2 - 2p1 * q + \|q\|^2} \ge \sqrt{\|p2\|^2 - 2p2 * q + \|q\|^2} \|p1\|^2 - \|p2\|^2 + 2(p1 - p2) * q \ge 0. \tag{22}$$

where ||p|| represents the Euclidean norm of p, represents the gradient system, and ||p||2 can be represented by $p\,p$. As a result, the problem of inequality can be broken down into a slew of gradient calculations. This shows that Espe's product conservation is being assessed in terms of encryption, i.e., ∀p1, p2 ∈ DB, p1 $p2$ = Espe (p1, K) Espe (p2, K), to calculate k-NN [24, 27].

The attacker cannot increment the estimate of P to diminish the likelihood of a collision and rehash the attack as, within the proposed design, the item maintenance encryption is not remotely retrievable as [28, 29]

$$f(p1', p2') = \sqrt{p1' * p1' - 2(p1' * p2') + p2' * p2'} \ne d(p1', p2'). \tag{23}$$

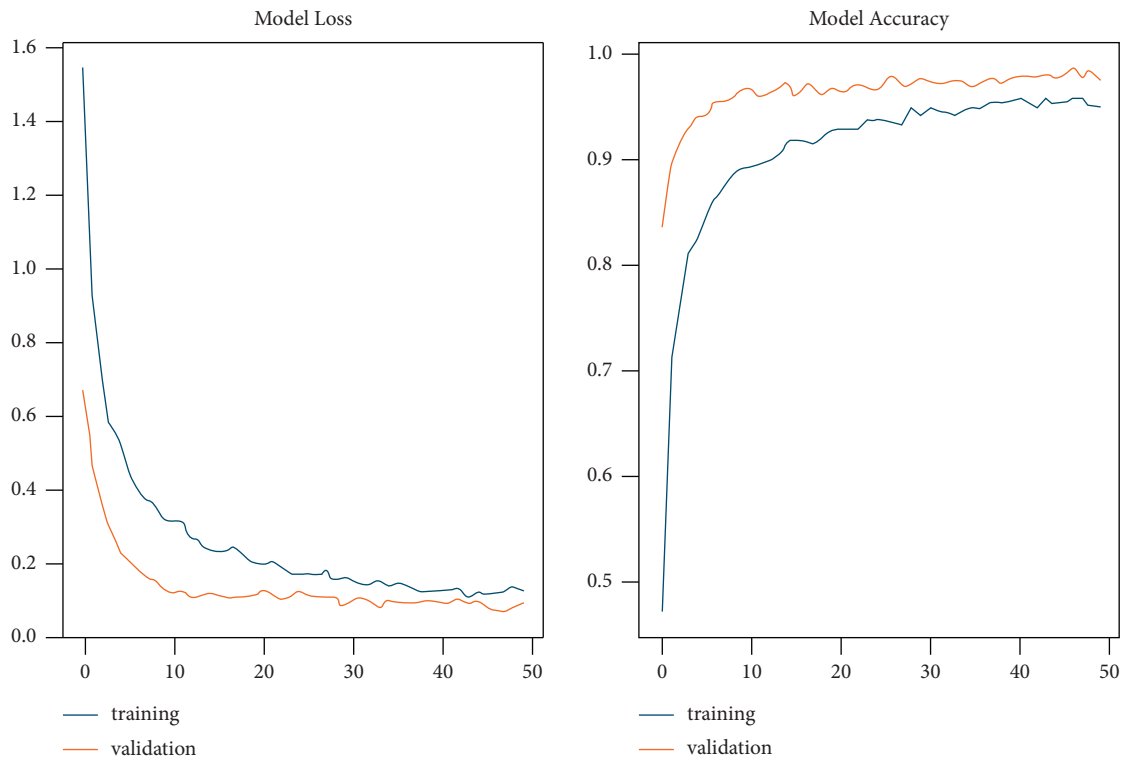


Figure 2: Proposed model loss and accuracy.

To put it another way, if the encryption E (i.e., E is DRE), then a computing technique f such that, for all points in time, the differential privacy function ET (i.e., ET is DRE) cannot be remotely retrieved, p_1 and p_2 and any encryption key K_1; it holds that $a_1 = E(p_1, K_1)$ and $a_2 = E(p_2, K_1)$; we have $f(a_1, a_2) = d(p_1, p_2)$. That is, the distance d (p1, p2) may be determined from the encrypted values a1 and a2 regardless of the encryption key.

3.3. Step 3: Recommendation System. Finally, an intelligent recommendation memory-based approach was used to measure user privacy and compute the similarity between users. It is a neighborhood-based collaborative filtering approach to produce recommendations [11–13]:

$$r_{u,i} = \mathrm{aggr}_{u' \in U} r_{u',i}. \tag{24}$$

The top N most comparable users to user u who share the same level of privacy as user i are denoted by U. The aggregation function includes

$$r_{u,i} = \frac{1}{N} \sum_{u' \in U} r_{u',i} r_{u,i} = k \sum_{u' \in U} \mathrm{simil}(u, u') r_{u',i}, \tag{25}$$

where r_u is the average privacy of user u for all the users rated by u.

The suggested technique determines the cosine similarity between two users in a neighborhood-based approach [4, 6, 30]:

$$\begin{aligned} \mathrm{simil}(x, y) &= \cos(\vec{x}, \vec{y}) \\ &= \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \times \|\vec{y}\|} \\ &= \frac{\sum_{i \in I_{xy}} r_{x,i} r_{y,i}}{\sqrt{\sum_{i \in I_x} r_{x,i}^2} \sqrt{\sum_{i \in I_y} r_{y,i}^2}}. \end{aligned} \tag{26}$$

Figure 2 shows the performance results of the proposed method.

As seen in the information supplied above, these findings demonstrate a solid solution to the challenging problem of grouping students to execute tailored educational programs. With the widespread usage of intelligent approaches such as those used in this study, small and heterogeneous student groups can form with members of each group sharing comparable characteristics of student ability, learning difficulties, and psychosocial and cognitive profile. By quickly managing the student potential in their class, as well as being aware of each group's unique characteristics such as their interests, unique experiences, learning rhythms, and

learning styles, the teacher can easily manage the student potential of their class and offer high-quality education, taking into account the specific educational needs and capabilities of each group. In addition, the algorithm may be utilized in traditional classrooms and digital or e-learning programs, facilitating the teaching role, as it can compensate for challenges in multicriteria grouping and differentiation of students in a wide range of subject areas. Additionally, it can be utilized with many pupils and produce results in a short amount of time, assuming that the required data is available. Another presumption supporting this idea is that the amount of data that can be regarded as quantitative data or the number of evaluable criteria that come from a comprehensive evaluation of a student is limitless. Finally, each student's talents are automatically classified into various groups. They were applying an AI technique known as directed acyclic graph learning. Each learner's private information is protected using differential privacy in the model. Collaborative filtering's intelligent module provides customized real-time privacy advice.

4. Conclusions

This study presented an innovative recommendation system for privacy-preserving education technologies. It is a hybrid intelligent computing system that can create learning programs based on the unique needs of each learner. It is based on advanced machine learning techniques for performing high-level privacy-preserving analyses to create learning repositories adapted to the trainees' skills and experiences. The instructional material of educational systems may be successfully rearranged depending on assessment criteria using this novel and privacy-preserving approach. Specifically, using machine learning and differential privacy, this study provides a directed acyclic graph approach to automatically classify each student into a category based on their skills. Next, the model takes advantage of differential privacy. This technology makes it possible to gather relevant information from databases containing the personal information of individuals without disclosing sensitive identification about each individual. Personalized real-time data are also provided by an intelligent suggestion process based on collaborative filtering.

The proposed intelligent system achieved remarkable results in all cases of evaluation, always taking into account the modeling difficulties and uncertainty introduced by the subjective learning system. An important innovation is related to using privacy-preserving recommendations capable of solving multidimensional and complex problems. Also, an exciting finding is emerged from this research related to the possibility of applying to truly unstructured data, techniques, and methodologies and derived from fully theoretical computing, with fully exploitable and realistic results. Furthermore, the proposed method uses a neighborhood-based methodology to determine the cosine similarity between two users as a significantly innovative approach.

Humans are prone to errors or biases that might skew results while doing repetitive tasks such as reading and analyzing open-ended survey replies and other text data. A few simple steps are required for natural language processing (NLP)-powered tools to be taught to the language and criteria of the educational process. So, once they get the machines up and running, they perform far better than humans could ever hope to accomplish. To keep up with the changing marketplace or the language of their education, NLP can be used to investigate and extend the model, which will allow the automated system to take full advantage of modeling learning systems' wider dependencies with greater accuracy and efficiency. Also, text analysis on a large scale on a variety of papers, internal systems, emails, social media data, online reviews, and more will be made possible by NLP technology. Data can be processed in a matter of seconds or minutes, compared to the days or weeks it would take to analyze manually.

Data Availability

The data used to support the findings of the study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] M. Al-Dojayli and A. Czekanski, "Integrated engineering design education: vertical and lateral learning," *Journal of Integrated Design and Process Science*, vol. 21, no. 2, pp. 45–59, 2017.

[2] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.

[3] G. D'Acquisto, J. Domingo-Ferrer, P. Kikiras, V. Torra, Y.-A. de Montjoye, and A. Bourka, "Privacy by design in big data: an overview of privacy enhancing technologies in the era of big data analytics," *ArXiv151206000 Cs*, 2015.

[4] Z. Shen and T. Zhong, "Analysis of application examples of differential privacy in deep learning," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 4244040, 15 pages, 2021.

[5] I. M. del Águila and J. del Sagrado, "Bayesian networks for enhancement of requirements engineering: a literature review," *Requirements Engineering*, vol. 21, no. 4, pp. 461–480, 2016.

[6] A. B. Mrad, V. Delcroix, S. Piechowiak, P. Leicester, and M. Abid, "An explication of uncertain evidence in Bayesian networks: likelihood evidence and probabilistic evidence," *Applied Intelligence*, vol. 43, no. 4, pp. 802–824, 2015.

[7] G. Canbek, S. Sagiroglu, T. T. Temizel, and N. Baykal, "Binary classification performance measures/metrics: a comprehensive visualized roadmap to gain new insights," in *Proceedings of the 2017 International Conference on Computer Science and Engineering (UBMK)*, pp. 821–826, Antalya, Turkey, October 2017.

[8] O. O. Koyejo, N. Natarajan, P. K. Ravikumar, and I. S. Dhillon, "Consistent binary classification with

generalized performance metrics," p. 9, 2014, https://dblp.org/rec/conf/nips/KoyejoNRD14.html.

[9] N. Elmrabit, F. Zhou, F. Li, and H. Zhou, "Evaluation of machine learning algorithms for anomaly detection," in *Proceedings of the 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1–8, Dublin, Ireland, June 2020.

[10] J. Bringer, H. Chabanne, and A. Patey, "Privacy-preserving biometric identification using secure multiparty computation: an overview and recent trends," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 42–52, 2013.

[11] C. C. Aggarwal, "Neighborhood-based collaborative filtering," in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed., Springer International Publishing, Manhattan, NY, USA, pp. 29–70, 2016.

[12] M. Deschênes, "Recommender systems to support learners' Agency in a Learning Context: a systematic review," *International Journal of Educational Technology in Higher Education*, vol. 17, no. 1, p. 50, 2020.

[13] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261–273, 2015.

[14] B. Alallawi, L. Denne, M. M. Apanasionok, C. F. Grindle, and R. P. Hastings, "Special educators' experiences of a numeracy intervention for autistic students," *European Journal of Special Needs Education*, no. 0, pp. 1–14, 2021.

[15] A. Demetriou, G. Spanoudis, and A. Mouyi, "Educating the developing mind: towards an overarching paradigm," *Educational Psychology Review*, vol. 23, no. 4, pp. 601–663, 2011.

[16] A. Klašnja-Milićević and M. Ivanović, "E-learning personalization systems and sustainable education," *Sustainability*, vol. 13, no. 12, 12 pages, Article ID 6713, 2021.

[17] H. J. Cha and M. L. Ahn, "Development of design guidelines for tools to promote differentiated instruction in classroom teaching," *Asia Pacific Education Review*, vol. 15, no. 4, pp. 511–523, 2014.

[18] C. Korkmaz and A.-P. Correia, "A review of research on machine learning in educational technology," *Educational Media International*, vol. 56, no. 3, pp. 250–267, 2019.

[19] J. T. Abbitt and W. J. Boone, "Gaining insight from survey data: an analysis of the community of inquiry survey using Rasch measurement techniques," *Journal of Computing in Higher Education*, vol. 33, no. 2, pp. 367–397, 2021.

[20] J. O. Berger, "Bayesian analysis," in *Statistical Decision Theory and Bayesian Analysis*, J. O. Berger, Ed., Springer, New York, NY, USA, pp. 118–307, 1985.

[21] M. S. Mahmud, J. Z. Huang, S. Salloum, T. Z. Emara, and K. Sadatdiynov, "A survey of data partitioning and sampling methods to support big data analysis," *Big Data Mining and Analytics*, vol. 3, no. 2, pp. 85–101, 2020.

[22] S. Blyumin, A. Pogodaev, and E. Khabibullina, "Graph-structural modeling of some special organizational systems," in *Proceedings of the 2020 2nd International Conference on Control Systems, Mathematical Modeling Automation and Energy Efficiency (SUMMA)*, pp. 279–283, Lipetsk, Russia, August 2020.

[23] T. Alshalali, K. M'Bale, and D. Josyula, "Security and privacy of electronic health records sharing using hyperledger fabric," in *Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 760–763, Las Vegas, NV, USA, September 2018.

[24] R. Hou, F. Tang, S. Liang, and G. Ling, "Multi-Party Verifiable Privacy-Preserving Federated k-Means Clustering in Outsourced Environment," *Security and Communication Networks*, vol. 2021, pp. 1–11, 2021.

[25] Z. Li, W. Xu, H. Shi, Y. Zhang, and Y. Yan, "Security and privacy risk assessment of energy big data in cloud environment," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–11, Article ID 2398460, 2021.

[26] S. Behera and J. R. Prathuri, "Application of homomorphic encryption in machine learning," in *Proceedings of the 2020 2nd PhD Colloquium on Ethically Driven Innovation and Technology for Society (PhD EDITS)*, pp. 1-2, Bangalore, India, August 2020.

[27] X. Su, T. M. Khoshgoftaar, and R. Greiner, "Imputed neighborhood based collaborative filtering," in *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 1, pp. 633–639, Sydney, NSW, Australia, December 2008.

[28] B. Bordel, R. Alcarria, and T. Robles, "Lightweight encryption for short-range wireless biometric authentication systems in Industry 4.0," *Integrated Computer-Aided Engineering*, vol. 29, no. 2, pp. 153–173, 2022.

[29] M. Iezzi, "Practical privacy-preserving data science with homomorphic encryption: an overview," in *Proceedings of the 2020 IEEE International Conference on Big Data (Big Data)*, pp. 3979–3988, Atlanta, GA, USA, September 2020.

[30] Q. Fu, Y. Tian, and J. Sun, "Modeling and simulation of dynamic lane reversal using a cell transmission model," *Journal of Intelligent Transportation Systems*, no. 0, pp. 1–13, 2021.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 6591140, 17 pages
https://doi.org/10.1155/2022/6591140

Research Article

Network Intrusion Detection Method Based on FCWGAN and BiLSTM

Zexuan Ma,[1] **Jin Li**,[1] **Yafei Song**,[1] **Xuan Wu**,[1] **and Chen Chen**[1,2]

[1]*College of Air and Missile Defense, Air Force Engineering University, Xi'an 710051, China*
[2]*Xi'an Satellite Control Center, Xi'an 710043, China*

Correspondence should be addressed to Yafei Song; yafei_song@163.com

Received 13 February 2022; Revised 11 March 2022; Accepted 15 March 2022; Published 13 April 2022

Academic Editor: Konstantinos Demertzis

Imbalanced datasets greatly affect the analysis capability of intrusion detection models, biasing their classification results toward normal behavior and leading to high false-positive and false-negative rates. To alleviate the impact of class imbalance on the detection accuracy of network intrusion detection models and improve their effectiveness, this paper proposes a method based on a feature selection-conditional Wasserstein generative adversarial network (FCWGAN) and bidirectional long short-term memory network (BiLSTM). The method uses the XGBoost algorithm with Spearman's correlation coefficient to select the data features, filters out useless and redundant features, and simplifies the data structure. A conditional WGAN (CWGAN) is used to generate a small number of samples in the dataset, add them to the original training set to supplement the dataset samples, and apply BiLSTM to complete the training of the model and realize the classification. In comparative tests based on the NSL-KDD and UNSW-NB15 datasets, the accuracy of the proposed model reached 99.57% and 85.59%, respectively, which is 1.44% and 2.98% higher than that of the same type of CWGAN and deep neural network (CWGAN-DNN) model, respectively.

1. Introduction

The continuous development of computer and network technology has greatly improved people's lives, but with it come a variety of attacks and threats at the network level, making network security an unavoidable and urgent problem. As an effective method to detect and defend against network attacks, the intrusion detection system (IDS) has been widely used. It monitors network traffic in real time, classifies it as normal or malicious, and provides information necessary to intrusion prevention systems. In recent years, machine learning and deep learning have been widely used for intrusion detection. However, since real-life network traffic data are unbalanced and relatively little has malicious attack attributes, the training sets of such methods are severely unbalanced. Hence, while existing network intrusion detection systems have high resolution accuracy for whether there is an attack, the detection accuracy of various samples is still low, especially for minority-class attacks, resulting in the misclassification of such traffic as other traffic, and the failure to meet performance analysis requirements. Therefore, it is important to solve the network data imbalance problem and improve the performance of model intrusion detection.

The class imbalance problem is commonly solved by enhancing the model training effect by increasing the number of samples in datasets, and much research has been conducted based on this method. Maryam Yousefnezhad et al. [1] proposed a feature extraction ensemble classification method based on deep learning. Firstly, the feature selection algorithm based on ensemble margin is used to select the samples, and the deep learning method is used to extract the sample features. Finally, the outputs of multiple KNN and SVM are combined according to Dempster–Shafer method. This method uses the method of ensemble learning, which can improve the detection rate of attack types to a great extent. At the same time, feature selection based on ensemble margin can remove the useless data in the original dataset, and improve the overall detection accuracy, and shorten the training time to a certain extent. However,

the structure is complex, there are many classifiers, and the overall calculation cost is high. Meanwhile, this method uses KNN and SVM as classifiers to classify samples, and the overall classification accuracy of the model has a large space for improvement. Considering the complexity of dimensions and the low efficiency of traditional algorithms, a chaotic cuckoo optimization algorithm with levy flight, disruption operator, and opposition-based learning (CCOALFDO) is proposed by Kelidari and Hamidzadeh [2]. The algorithm combines levy flight, disruption operator and opposition-based learning to select the optimal feature subspace for classification. Levy flight can deal with uncertainty and better update the cuckoo steps in high-dimensional space. The opposition-based learning and disruption operator can improve the search ability of the algorithm and ensure the diversity of the population. The algorithm proposed in this paper combines the above advantages, which can greatly reduce the randomness of feature selection and avoid falling into the local optimal solution. At the same time, due to the elimination of some redundant features, the classification accuracy can be greatly improved. However, the combination of multiple algorithms leads to the increase of the overall computational complexity of the algorithm, which requires higher computational cost, slows down the convergence speed and increases the computational time. Gonzalez-Cuautle et al. [3] proposed a resampling method that integrates the synthetic minority oversampling technique (SMOTE) and grid search algorithms to solve the problems of overfitting and low classification accuracy. This method improved the classification results of the intrusion detection system (IDS) dataset by merging synthetically generated balanced data and adjusting different supervised learning algorithms. SMOTE can oversample the data sample and increase the number of minority data. The grid search algorithm can automatically optimize the parameters, and find the parameters with the best detection effect and apply them to the model structure, and avoid falling into the local optimal solution, which ensures the optimality of the model detection effect. However, SMOTE randomly synthesizes the original data according to the k-nearest neighbor principle, does not learn the essence of the original data, and the quality of the generated samples is poor. At the same time, the grid search algorithm searches every parameter, which leads to too large calculation cost, too long calculation time, and there is a large space for improvement. Lee and Park [4] proposed AE-CGAN-RF, a model to solve the data imbalance problem by using an autoencoder to reduce the dimension of the network traffic and a conditional generative adversarial network (CGAN) to generate data samples, which were passed to a random forest (RF) to complete the intrusion detection classification. The model could greatly improve the accuracy of minority class sample detection, and reduces the data dimension, which reduces the time required for training and reduces the calculation cost. However, the use of RF as a classifier led to a low overall detection accuracy because of RF's weak classification ability. Lee and Park [5] proposed a detection model using a generative adversarial network (GAN) to generate minority class attack samples and RF for classification. This method increases the minority samples of CIC-IDS2017 dataset and improves the detection ability of the model for minority attack samples, so that the model can achieve better detection effect. At the same time, the structure is simple and the detection speed is fast. However, only ordinary GAN is used for sample generation, without considering the instability of the GAN, there are hidden dangers in the process of sample generation, and other datasets and models were not used to further validate its feasibility, which is not convincing. Liu et al. [6] proposed a GAN-FS method to address feature redundancy. The model can select dataset features based on feature variance, eliminate the impact of redundant data and useless data on the model detection effect to a great extent, improve the accuracy and speed of detection, and uses a GAN to generate samples, which increase the number of samples and enhance the training effect. Comparative experiments confirmed that the method could effectively improve model detection performance, but the method does not consider the degrees of freedom of GAN training, and the generated data are unsupervised and uncontrollable. Compared with CGAN, it is less targeted. At the same time, it only selects the features according to the feature variance, and the detection method is not comprehensive, which has certain limitations. He [7] addressed the low accuracy of class imbalance data detection and proposed a model using a conditional Wasserstein generative adversarial network (CWGAN) to generate minority class attack samples and a Deep Neural Networks (DNN) as a classifier for network intrusion detection, which improves the detection effect compared to a DNN. However, only using DNN as classifier to identify intrusion behavior, there is still a large gap in detection accuracy compared with other deep learning methods. At the same time, the high dimensionality of data is not considered, and the use of the network intrusion detection system in a large-scale network environment will be limited by time and space complexity because the data have high dimensionality and nonlinear characteristics. Therefore, dimensionality reduction for high-dimensional data is a key step to improve detection speed and performance.

To solve the above problems, this paper combines feature selection with a CWGAN. The feature selection-based dimensionality reduction of high-dimensional data can filter out redundant and useless features, simplify the data structure, improve intrusion detection performance, and decrease training time. The CWGAN oversamples the minority class data to supplement the samples and balance the data distribution, thus improving detection performance. A bidirectional long short-term memory network (BiLSTM) is used to extract and classify the features from the time series. The loss function and optimization algorithm are analyzed to select the most suitable hyperparameters.

This paper makes the following contributions:

(1) We propose FCWGAN-BiLSTM, a network intrusion detection system based on FCWGAN and a BiLSTM network, to alleviate the impact of class imbalance on detection performance and improve

the overall performance of a network intrusion detection model

(2) We use XGBoost and Spearman correlation coefficients for feature selection to filter out redundant and useless data and simplify the feature structure, which reduces computational difficulty and improves detection accuracy

(3) We apply CWGAN to generate minority class samples to supplement the dataset, enhance the model training effect, reduce the impact of class imbalance on the detection rate, and improve detection performance

(4) A BiLSTM network captures information in network traffic data with long-term dependency, extracts network traffic feature extraction based on time series, and effectively uses future moment information to improve the model classification effect

(5) Model performance analysis experiments, model ablation experiments, and comparison experiments with different data augmentation algorithms and classification algorithms demonstrate the performance of the proposed model

The rest of this paper is organized as follows. Section 2 presents the background and related work. Section 3 presents the proposed model, Section 4 provides experimental results and analysis, and Section 5 presents the conclusions.

2. Background and Related Work

2.1. Feature Selection. Feature selection is a method of selecting relevant features of a dataset by obtaining a subset from the original feature set based on specific criteria. Data dimensionality reduction is often applied to high-dimensional complex data [8]. Unlike feature extraction, feature selection preserves the physical meaning of the original features by retaining some of the data, and thus makes the model more readable and interpretable [9, 10]. In the field of intrusion detection, where datasets are characterized by a large volume of data and high dimensionality, feature selection reduces computational difficulty and eliminates data redundancy [11], thereby improving the detection rate of the model and reducing false positives. For example, a firefly algorithm was used for feature selection and to pass the generated features through a classifier based on C4.5 and a Bayesian network (BN) to complete the classification for intrusion detection [12]. The method selected important features in the KDD CUP 99 dataset and reduced the 41-dimensional features to 10 dimensions, which achieved better detection performance and reduced computation. However, the method suffers from a low discovery rate and slow solution speed, which leads to long calculation times. Le et al. [13] proposed SFSDT, a feature selection model that combines a hybrid sequence forward selection (SFS) algorithm with a decision tree (DT) model to select the best feature subset from the complete set of features in a dataset. The CF function in the SFS algorithm is adjusted, and the accuracy and error score of the DT model on each feature subset are generated by the SFS. SFSDT starts from an empty set and sequentially adds features to enhance the accuracy of the DT model until it is maximized on a validation dataset (feature subset). The algorithm reduces execution time and required memory, and significantly improves detection performance. However, SFS can only add features, and cannot remove them, and it tends to fall into local optima. Thus, it requires a large number of experiments to obtain the best subset. Considering the above problems, we use XGBoost and the Spearman correlation coefficient for dataset feature selection.

2.1.1. XGBoost. Proposed by Chen in 2015, XGBoost (eXtreme Gradient Boosting) is a model framework based on the idea of the gradient boosting decision tree (GBDT) [14]. It has the advantages of high speed, high efficiency, and strong performance, and has been widely used to solve classification and regression problems. The core idea is to generate a new tree by splitting the features in a dataset, and then to add new trees. It fits the residual of its last prediction to obtain a new function and improves performance through iteration. The traditional GBDT algorithm uses only first-order derivative information, while XGBoost uses a second-order Taylor expansion of the loss function and a regular term to speed up training and prevent overfitting. We use this method to rank the importance of features in the dataset [15].

2.1.2. Spearman Correlation Coefficient. We use the Spearman correlation coefficient to measure the correlation between features. Proposed by Spearman in 1904, it measures the strength of the relationship between two variables [16], and it takes values in the range (–1, 1). The Spearman correlation coefficient between variables x_i and y_i is calculated as

$$\rho = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i (x_i - \overline{x})^2 \sum_i (y_i - \overline{y})^2}}, \tag{1}$$

where $x_i\,(i = 1, 2, \ldots, n)$ and $y_i\,(i = 1, 2, \ldots, n)$ are elements of the vectors X and Y, respectively. A value of ρ close to ± 1 indicates a strong association; hence one of the features can be filtered out. A value close to 0 indicates that there is no association between them, and both should be retained.

2.2. CWGAN. A GAN is a deep learning model inspired by the two-person zero-sum game in game theory and is used to simulate complex high-dimensional distributions of real-world data. It consists of a generator (G) and discriminator (D) [17], which are both neural networks. The generator captures the potential distribution of real data samples and generates new data samples. The discriminator is a binary classifier used to determine whether the input sample is real or generated data. The classification results are passed back to the generator and discriminator through updates of the

weighted loss. The above networks are trained until the discriminator can no longer distinguish between real and generated samples [18]. Its optimization process is a minimax game problem with the goal to achieve a Nash equilibrium so that the generated network can estimate the distribution of the data samples [19]. The objective function for generating the adversarial network is

$$\min_G \max_D V(D,G) = E_{x\sim p_{\text{data}}(x)}[\log D(x)] + E_{z\sim p_{\text{data}}(z)} [\log(1 - D(G(z)))], \tag{2}$$

where p_{data} denotes the distribution of real samples, the function $G(z)$ maps noise z to the data space, and $D(x)$ is the probability that sample x is real data. To distinguish between real and generated data, $D(x)$ should be as large as possible, and $D(G(z))$ as small as possible.

The CGAN is based on a GAN, where category information and noise are merged with the original data as the input to the generator and discriminator [20], with loss function

$$\min_G \max_D V(D,G) = E_{x\sim p_{\text{data}}(x)}[\log D(x|y)] + E_{z\sim p_{\text{data}}(z)}[\log(1 - D(G(z|y)))], \tag{3}$$

where y represents the category information, and other parameters are the same as in (2).

A GAN is different from ordinary oversampling, as it generates new samples by obtaining the potential distribution of the original data and passing it randomly into the generator. By training the generator and discriminator, the generated samples are similar to the original sample distribution with high confidence. GANs are used to generate samples for minority classes and to expand datasets. For example, the SIGMA method [21] generates new samples to enhance the ability of IDSs to resist new types of attacks, combining a GAN with hybrid local search and genetic algorithms to iteratively generate new samples to retrain the intrusion detection system based on machine learning until the detection rate converges. AEGAN [22] is a hybrid model consisting of adversarial environment reinforcement learning (AE-RL) and a CGAN, whose model is trained on a network intrusion detection dataset to generate synthetic samples to deal with class imbalance problems. The above methods can improve the performance of network intrusion detection systems, but none considers the vanishing gradient problem that might occur during the training of GANs.

GANs and CGANs can generate samples and reduce class imbalance problems. However, their use of Jensen-Shannon scatter requires overlap between the distributions of real and generated samples, which is nonexistent or negligible when the discriminator is trained to be optimal, which can lead to model collapse and vanishing gradient problems [23].

To solve the above problems, we introduce the Lipschitz limit and Wasserstein distance to CGAN to realize CWGAN for the dataset samples, with the workflow shown in Figure 1.

We fix the discriminator, input the noise vector and labels to the generator, and train it to simulate the real data distribution. We use the discriminator to judge the real and generated samples. If it cannot distinguish between them, we fix the generator and train the discriminator, and if it can, we fix the discriminator and train the generator. We repeat these steps until the loss function of the discriminator is stabilized at about 0.5, at which time we generate attack samples and add them to the training set.

Through the above method, the model can generate data of a specified pattern to supplement the dataset, while effectively avoiding the vanishing gradients caused by the failure of the discriminator to converge during training. The objective function of CWGAN is

$$V(D,G) = \max_D \left\{ E_{x\sim p\text{data}}[D(x|y)] - E_{x\sim p_g}[D(x|y)] - \lambda E_{x\sim p_{\text{Penaty}}}\left[\left\|\nabla_x D(x|y)\right\| - 1\right]^2 \right\}, \tag{4}$$

where λ is an artificial parameter, $\|\nabla_x D(x)\|$ is the calculation paradigm for x in $D(x)$, and $x \sim p_{\text{Penaty}}$ is the middle position of the line connecting points on p_r and p_g.

2.3. BiLSTM. The model in a traditional neural network focuses only on the processing of the current moment, while a recurrent neural network (RNN) can use information processed at the current moment at the next moment [24]. Considering the problem of the vanishing gradient and gradient explosion during the training of an RNN, Hochreiter et al. proposed the long short-term memory network (LSTM) [25], which adds a gate mechanism and a memory unit on the basis of the RNN and memory unit to effectively solve the problems of RNNs, and better solves the longer distance dependence problem [26]. LSTM has input, forget, and output gates, as shown in Figure 2.

The LSTM structure is described as

$$\begin{cases} f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \\ i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \\ \widetilde{C}_t = \tanh\left(W_C \cdot [h_{t-1}, x_t] + b_C\right) \\ C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t \\ o_t = \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \\ h_t = o_t \cdot \tanh\left(C_t\right) \end{cases}, \tag{5}$$

where f_t is the forget gate; i_t is the input gate; $\widetilde{C}_t$ and C_t are the current input and unit state, respectively; σ is the sigmoid function; W_f, W_i, W_o, and W_C are the weight matrices of the forget gate, input gate, output gate, and current input unit state, respectively; $[h_{t-1}, x_t]$ denotes the concatenation of the two vectors; and b_f, b_i, b_o, and b_C are the bias terms of the forget gate, input gate, output gate, and current input unit state, respectively. The above parameters change continuously during training.

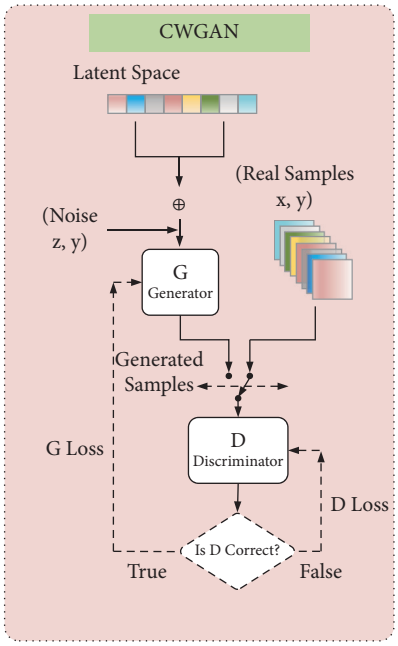


Figure 1: CWGAN workflow diagram.

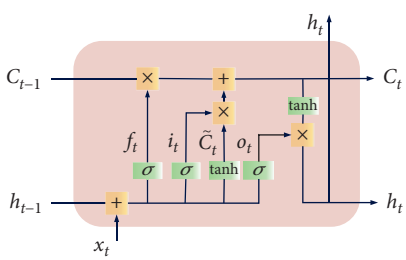


Figure 2: LSTM structure diagram.

Considering the distinct temporal characteristics of network traffic data, the use of RNN-like approaches to deal with network intrusion problems has unique advantages. For example, in [27], a deep learning-based intrusion detection system, DL-IDS, uses a hybrid network of convolutional neural networks (CNNs) and LSTM to extract the spatiotemporal characteristics of network traffic data, thus providing a better intrusion detection system. However, it was not considered that the unidirectional LSTM can only read sequence data from one direction and cannot exclude the influence of subsequent information on the detection results. Thus, BiLSTM was used instead of LSTM to process incoming data [28].

BiLSTM combines forward and backward LSTM to learn from forward and backward time-series data. The hidden layer contains two units with the same input that are connected to the same output, where one processes the forward time series, and the other the backward time series, increasing the time series involved in training by learning features better, thus providing higher accuracy for longer time series. The BiLSTM process is shown in Figure 3.

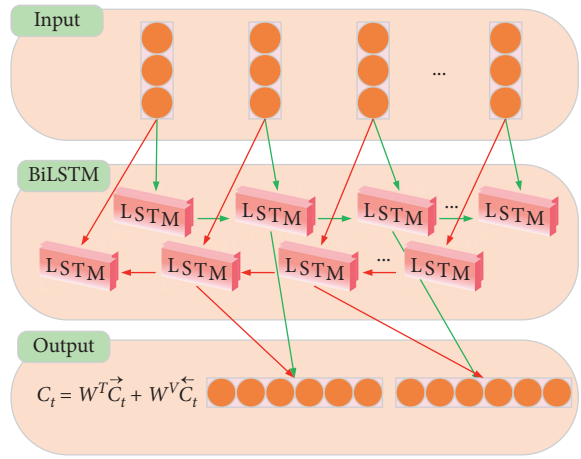


Figure 3: BiLSTM process diagram.

The process is

$$\begin{cases} \overrightarrow{C}_t = \mathrm{LSTM}\left(x_t, \overrightarrow{h}_{t-1}, \overrightarrow{C}_{t-1}\right) \\ \overleftarrow{C}_t = \mathrm{LSTM}\left(x_t, \overleftarrow{h}_{t-1}, \overleftarrow{C}_{t-1}\right) \\ C_t = W^T\overrightarrow{C}_t + W^V\overleftarrow{C}_t \end{cases}, \tag{6}$$

where the LSTM function represents the nonlinear transformation of the input feature, which is encoded as the corresponding hidden state of the LSTM ((5) and W^T and W^V are the weight coefficients corresponding to the forward and backward moment unit state, respectively.

3. Network Intrusion Detection Method Based on FCWGAN and BiLSTM

We propose a network intrusion detection method based on FCWGAN and BiLSTM. XGBoost is used in the feature selection stage to rank the importance of the features in the dataset, whose relevance is analyzed based on the Spearman correlation coefficient. Features with strong relevance and low importance are filtered out to simplify the feature structure. The selected features are passed into CWGAN together with the labels, and minority class samples in the training set are generated in a controlled manner. Generated samples are passed into BiLSTM together with the original data in the training set for training, and the model is validated on a test set. The intrusion detection process includes stages of data preprocessing, feature selection, sample generation, feature extraction and training, and testing, as shown in Figure 4.

3.1. Data Preprocessing. Tag encoding was used to convert the string-type features in the NSL-KDD and UNSW-NB15 datasets to numeric-type. It was judged whether there was a null value in the dataset, and if there was none, the data were normalized by Min – Max,

$$x = \frac{x - M_{\min}}{M_{\max} - M_{\min}}, \tag{7}$$

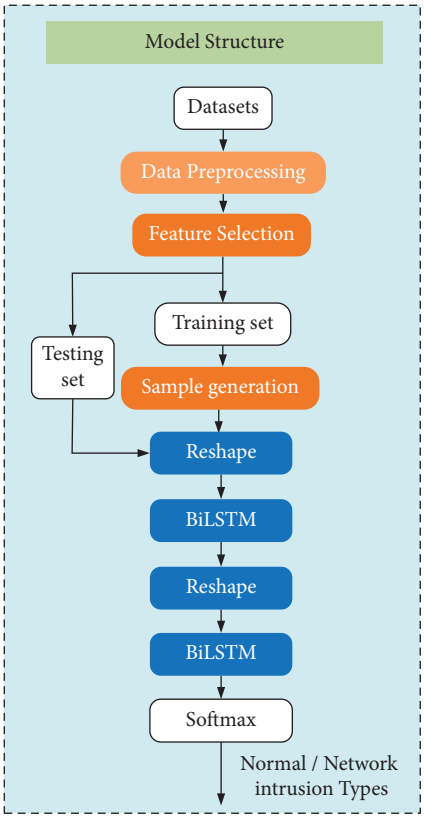


Figure 4: Schematic diagram of model structure based on FCWGAN and BiLSTM.

where $M_{\min}$ and $M_{\max}$ are the minimum and maximum values, respectively, of the dimension.

3.2. Feature Selection. In the feature selection stage, we used XGBoost to rank the feature importance, and Spearman's correlation coefficient to analyze the feature relevance. Irrelevant and redundant features were filtered out, and important features were retained to improve detection speed and enhance detection results.

XGBoost obtains a new function by fitting the residuals of the last prediction of the model and iterates to improve model performance [29]. Unlike the traditional GBDT algorithm that uses only first-order derivative information, the XGBoost algorithm performs a second-order Taylor expansion on the loss function and adds a regularization term to improve the model training speed and prevent overfitting. The target loss function of the XGBoost algorithm is

$$\begin{cases} \mathrm{Obj} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \\ \Omega(f_k) = \gamma T + \frac{1}{2}\lambda \|\omega\|^2 \end{cases}, \tag{8}$$

where $l(y_i, \hat{y}_i)$ is the loss function, which represents the difference between the predicted value $\hat{y}_i$ and true value y_i; and $\Omega(f_k)$ aims to prevent overfitting, where T is the number of child nodes, ω denotes the leaf weights, γT reduces the number of leaf nodes in the tree, γ is the penalty coefficient, $\lambda\|\omega\|^2$ is the regularization term, and λ is the regularization coefficient.

XGBoost requires several iterations to continuously generate the tree [30], assuming that the t-th iteration produces the tree, and the objective function of the t-th iteration is

$$\mathrm{Obj}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t). \tag{9}$$

where $\Omega(f_t)$ is a function to prevent overfitting.

We can evaluate the reasonableness of the decision tree structure based on the structure loss,

$$\mathrm{Obj}^{(t)}(p) = -\frac{1}{2}\sum_{j=1}^{T}\left(\frac{\sum_{i\in I_j} g_i)}{\sum_{i\in I_j} h_i + \lambda + \gamma T},\right. \tag{10}$$

where g_i and h_i are the first- and second-order derivatives of the loss function to the predicted values after iteration t-1, $I_j = \{i | p(x_i) = j\}$ is the index of leaf node j, and a smaller structural loss indicates a better decision tree structure.

If the tree splits at node j, the structure gain of the leaf node is

$$\mathrm{Obj}_s = \mathrm{Obj}(p_{\mathrm{before}}) - \mathrm{Obj}(p_{\mathrm{after}}) = \frac{1}{2}\left(\frac{\left(\sum_{i\in I_j} g_i\right)^2}{\sum_{i\in I_j} h_i + \lambda} + \frac{\left(\sum_{i\in I_L} g_i\right)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{\left(\sum_{i\in I_R} g_i\right)^2}{\sum_{i\in I_R} h_i + \lambda}\right) - \gamma, \tag{11}$$

where γ is the split coefficient, which can reduce the complexity of the model and prevent overfitting. This split gain is used to judge the quality of the split node.

Based on the above formulas, the importance of the features was ranked, and their relevance was analyzed through the Spearman correlation coefficient.

The importance of the features is sorted according to formula (11), and the Spearman correlation coefficient is used to analyze the feature correlation. The two are combined to eliminate irrelevant and redundant features, filter out key features, and pass them to the GAN for minority class sample generation.

3.3. Sample Generation. In the sample generation process, CWGAN was trained using noise and data samples that underwent feature selection and preprocessing [31], as shown in Table 1.

In the process of training CWGAN, the generator and discriminator were trained in turn, as follows:

(1) The discriminator is fixed and the generator is trained to simulate the distribution of the real data

(2) The generator is fixed, and the discriminator is trained until it cannot distinguish whether samples are from the real dataset or the generator

(3) The discriminator is fixed, and the generator is trained until the discriminator cannot distinguish samples by successive training

(4) Steps 1–3 are repeated until the loss value of the discriminator reaches 0.5

(5) The generator is used to generate attack samples, and these are added to the training set to complete sample generation

Table 1: CWGAN training algorithm.

Algorithm 1: minority class sample generation based on CWGANs
Input: $s = (z, y)$, where z is noise data, y is class label
Output: $s_G = [G(z, y'), y']$
(1) **While** D **does not approach** 0.5 /*CWGAN training*/
(2) **for** $t = 1, \ldots, n$ **do** /*optimize discriminator*/
(3) Sampling $\{(x_i, y_i)\}_{i=1}^{n_z}$ from $p_{\text{data}}(x, y)$
(4) Sampling $\{(z_i)\}_{i=1}^{n_z}$ from $p_z(z)$
(5) $\eta_{\theta_D} \leftarrow \nabla\theta_D \left[1/n_z \sum_{i=1}^{n_z} \left\{ \begin{array}{c} D(x_i, y_i) - D(G(z_i, y_i'), y_i') - \\ \lambda E_{(x,y)\sim p_{\text{Penaty}}} [\lVert \nabla_{(x,y)} D(x, y) \rVert - 1]^2 \end{array} \right\} \right]$
(6) $\theta_D \leftarrow \theta_D + \alpha_D \cdot \text{Adam}(\theta_D, \eta_{\theta_D})$
(7) **end**
(8) from $p_z(z)$ sample $\{(z_i)\}_{i=1}^{n_z}$ /*optimize generator*/
(9) $\eta_{\theta_G} \leftarrow \nabla\theta_G [1/n_z \sum_{i=1}^{n_z} (D(G(z_i, y_i'), y_i'))]$
(10) $\theta_G \leftarrow \theta_G - \alpha_G \cdot \text{Adam}(\theta_G, \eta_{\theta_G})$
(11) **end**
(12) **return** u /*generate samples*/

where θ_G, η_{θ_G}, θ_D and η_{θ_D} respectively denote the network parameters and gradients of the generator and discriminator.

3.4. Feature Extraction and Training. In the feature extraction stage, a BiLSTM layer learned the long-term temporal features in the dataset, Nadam optimization was applied to the neural network [32], a dropout layer alleviated overfitting, and a softmax classifier was used for network attack classification.

3.5. Testing. The trained model was used to classify the test set to obtain the prediction type. To ensure credible test results, the model was tested by k-fold cross-validation. The softmax function,

$$\sigma(x)_j = \frac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}} \quad j = 1, \ldots, K, \tag{12}$$

was used to calculate the probability of the classification and compare it with the original labels.

4. Experiment and Result Analysis

4.1. Experimental Settings. The performance of network intrusion detection methods based on FCWGAN and BiLSTM were evaluated according to the following experiments:

Experiment 1: model performance analysis

Experiment 2: model noise robustness

Experiment 3: model ablation

Experiment 4: comparison of data enhancement algorithms

Experiment 5: comparison of classification algorithms

Experiment 6: comparison of intrusion detection models

The experimental environment was a 64-bit Windows 10 operating system with TensorFlow learning framework, an AMD Ryzen 9 5900HX with Radeon Graphics at 3.30 GHz, and 32 GB RAM.

A Bayesian optimization algorithm was used for automatic optimization of model parameters, whose settings are shown in Table 2.

The categorical cross-entropy loss function is

$$\text{oss} = -\frac{1}{N} \sum_{i=0}^{N} \left(y_i \log y_i + (1 - y_i) \log(1 - y_i) \right). \tag{13}$$

4.2. Dataset and Experimental Evaluation Criteria. The proposed model was evaluated on the NSL-KDD and UNSW-NB15 datasets.

The NSL-KDD dataset was obtained by Tavallaee et al. in 2009 by eliminating duplicate instances in the KDD99 dataset and enabling a more objective reflection of the detection accuracy of the model [33]. It includes DoS, Probe, R2L, and U2R attack types, and has 41 attributes, but the data are extremely unbalanced. It has far fewer attack instances than normal instances, with only 995 R2L attacks and 52 U2R attacks.

The UNSW-NB15 dataset was created by the Cyber Range Lab of the Australian Cyber Security Centre, and includes attack types other than NSL-KDD, i.e., Fuzzers, Analysis, Backdoor, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. Similarly, there are far fewer attack instances than normal instances.

The distributions of training set types for the NSL-KDD datasets are shown in Figure 5.

The distributions of training set types for the UNSW-NB15 datasets are shown in Figure 6.

Comparative experiments used classification accuracy, precision, recall, and F1-score to judge the classification effectiveness of the models. The classification confusion matrix is shown in Table 3.

Table 2: Model parameter settings.

Parameter	Setting
XGBoost maximum depth	12
XGBoost gamma value	0
CWGAN learning rate	0.0001
CWGAN training iterations	200
Noise dimension	32
Batch size setting	1024
Loss function	Categorical cross-entropy
Optimizer	Nadam
Optimizer learning rate	0.001
BiLSTM cell count	64/128
Dropout rate	0.5

The four evaluation criteria are as follows:

$$\begin{aligned} \text{accuracy} &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \\ \text{recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ F1 - \text{score} &= 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \end{aligned} \tag{14}$$

4.3. Experimental Results and Analysis

4.3.1. Model Performance Analysis Experiment. To verify the effectiveness of the proposed model at network intrusion detection, we set up performance analysis experiments on network intrusion detection methods based on FCWGAN and BiLSTM.

FCWGAN was used to select the features of the training set samples of the NSL-KDD and UNSW-NB15 datasets, filter out redundant and useless samples, and simplify the data structure. The feature importance was judged using XGBoost, and the feature importance scores were obtained as shown in Figures 7 and 8.

The feature importance score in Figures 7 and 8 selects the total splitting gain, which can better reflect the importance of variables to the model.

From Figure 7, one can see that among the features of NSL-KDD datasets, the "dst_host_srv_count" is the most important and the "su_attempted" is the lowest; Similarly, it can be seen from Figure 8 that among the features of UNSW-NB15 datasets, the "dur" is the most important and the "ct_ftp_cmd" is the lowest. At the same time, it can be seen that in the above two datasets, the importance of different features varies greatly, and the importance of individual features is close to 0, which has little influence on the discrimination of sample types. Therefore, these useless features with low importance can be screened out to simplify the feature structure.

The feature correlations were analyzed using the Spearman correlation coefficient; the correlation between individual features is strong, and redundant features can be filtered out (Figures 9 and 10).

We combined the feature importance and correlation for analysis, and the filtered features are shown in Table 4.

Training set samples were then generated based on the selected features. We expanded the training set samples and combined the generated and original samples. The data distribution of the combined training set is shown in Tables 5 and 6.

Finally, the training set was passed into the BiLSTM network for training, and the test data were passed into the completed model to evaluate the model detection effect. The trends of model detection accuracy and average loss with the number of iterations are shown in Figures 11 and 12.

The trends of various class detection rates with the number of iterations are shown in Figures 13 and 14.

From Figures 11 and 12, one can see that the accuracy of the model increases rapidly with the number of iterations at the early stage of training, and gradually stabilizes; the average loss decreases rapidly with the number of iterations, and can reach a stable state quickly. Using the proposed model to perform multiclassification on the NSL-KDD and UNSW-NB15 datasets, the best accuracy rates are 99.57% and 85.59%, respectively. This shows that the model can distinguish types of network intrusion attacks well, thus obtaining high detection accuracy and a better detection effect.

From Figures 13 and 14, it can be seen that the proposed model can accurately identify normal and majority class attacks on both datasets, and the detection rate for minority class attacks can also reach a high standard, showing that the minority class samples generated by the model largely alleviate the impact of the class imbalance problem, thus improving the overall detection effect.

4.3.2. Model Noise Robustness Experiment. In recent years, the network environment has become more and more complex. In addition to a large number of redundant and useless data, there are also noise data in the network data, which will lead to the low robustness of the intrusion detection system [34]. In order to verify the robustness of the model proposed in this paper to noise, this section sets up a noise robustness experiment for network intrusion detection methods based on FCWGAN and BiLSTM.

Different levels of Gaussian white noise are added to NSL-KDD and UNSW-NB15 datasets, which obey $N(0, 0.02)$, $N(0, 0.04)$, $N(0, 0.06)$ and $N(0, 0.08)$, respectively. The detection accuracy of the model under the influence of different noise levels is shown in Table 7.

From Table 7, it shows that the accuracy of the two datasets decreases to a certain extent with the increase of the noise level. However, the range of change did not exceed 1.5%. This shows that the model proposed in this paper has strong robustness and stability to the interference of noise, and a small amount of noise data cannot have a significant impact on the performance of the model. At the same time, according to the conclusion of 3.3.1, different levels of Gaussian white noise are added to several features with

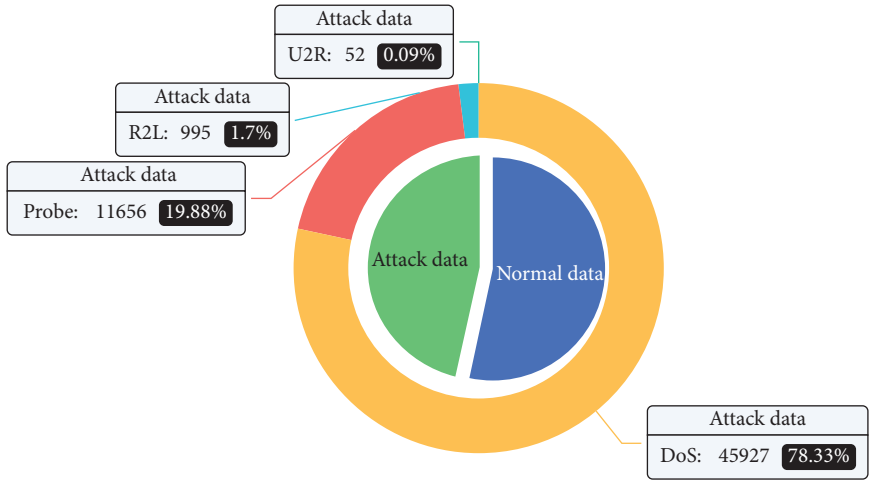


Figure 5: Distribution of NSL-KDD training set types.

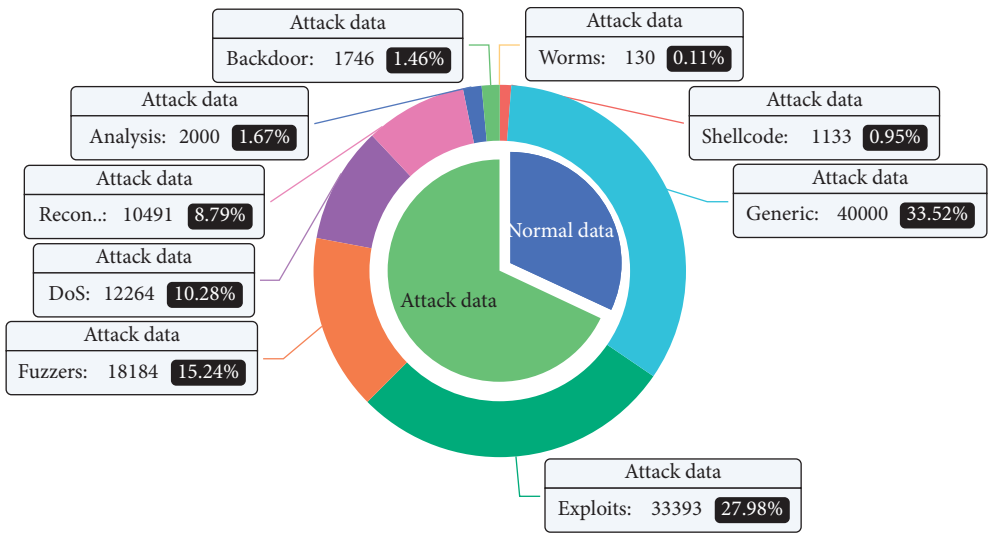


Figure 6: Distribution of UNSW-NB15 training set types.

strong correlation, middle correlation and weak correlation, which tests the accuracy of model detection. The result shows that adding noise to the features with stronger correlation has more obvious impact on the performance of the model, while the features with weaker correlation have little impact. It shows that when dealing with noise, it is not necessary to deal with all features, but only some noise sensitive features, which also confirms the necessity of feature selection.

4.3.3. Model Ablation Experiment. We set up model ablation experiments to verify the proposed feature selection and the ability of CWGAN to improve the detection effect of the model for minority samples.

Under the same experimental conditions, BiLSTM, GAN-BiLSTM, CWGAN-BiLSTM, and the model in this paper were compared on the NSL-KDD dataset. The detection rates of each model for various types of NSL-KDD datasets were evaluated, and are displayed in Table 8.

Table 3: Definition of classification confusion matrix.

		Predicted class	
		Normal	Abnormal
Actual class	Normal	TP	FN
	Abnormal	FP	TN

From Table 8, it shows that the feature extraction and the proposed CWGAN played a relatively significant role in the improvement of the detection rate for minority class samples. The reason is that real-world data contain many irrelevant, redundant, and noisy features, whose removal through feature selection can greatly reduce storage and computational costs, and can simplify the data structure and improve the detection results. The proposed feature selection method was used to directly select a subset of relevant features for the model, eliminate useless and redundant features, and improve the test effectiveness from the original

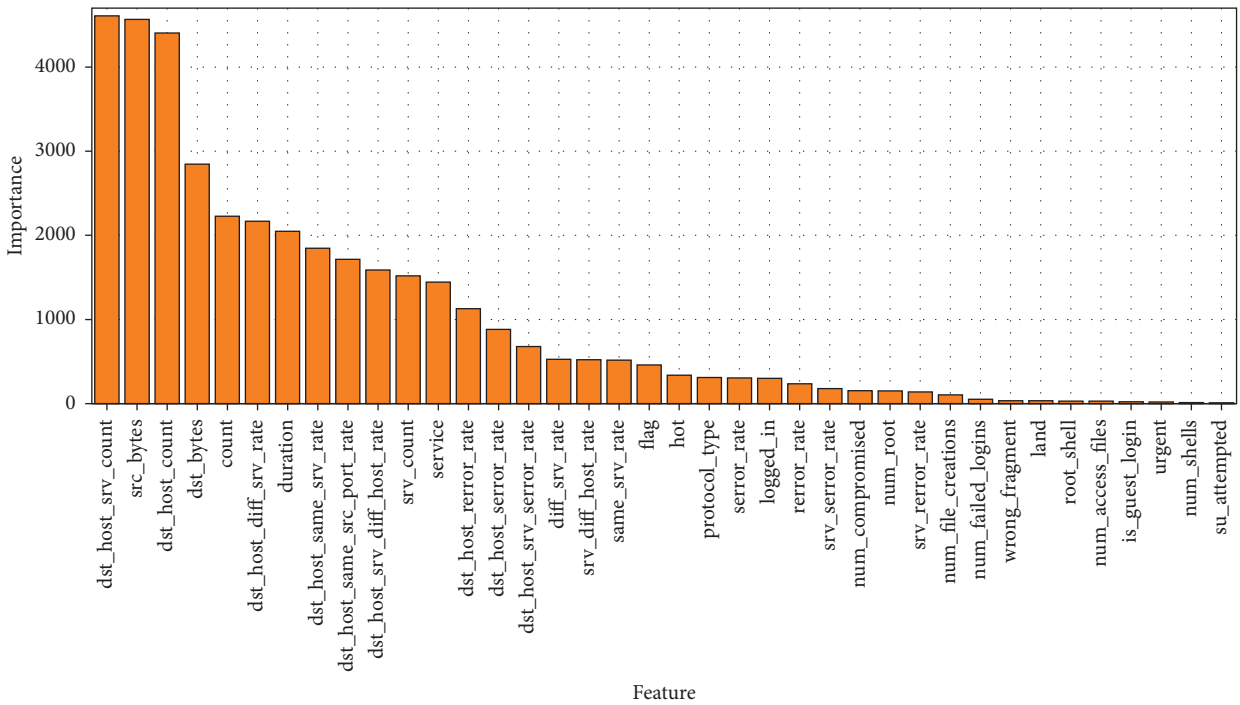


Figure 7: Feature importance map of NSL-KDD.

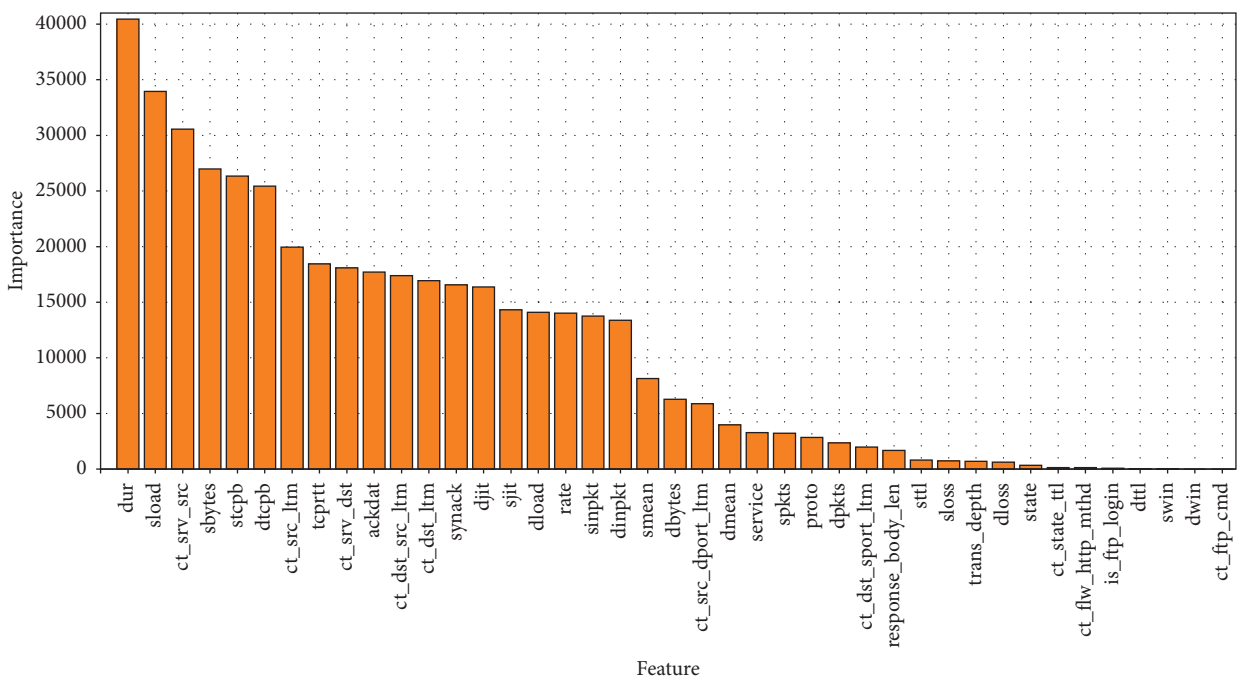


Figure 8: Feature importance map of UNSW-NB15.

dataset level. CWGAN achieved the controlled generation of minority samples by adding category information and the Wasserstein distance, while avoiding the vanishing gradient, and combining it with the original training set to increase the number of minority samples. Ultimately, this enhanced the training effect of the model. Therefore, we combined the two to process the dataset and improve the test effect of the model.

4.3.4. Comparative Experiments with Different Data Enhancement Algorithms. We set up comparison experiments

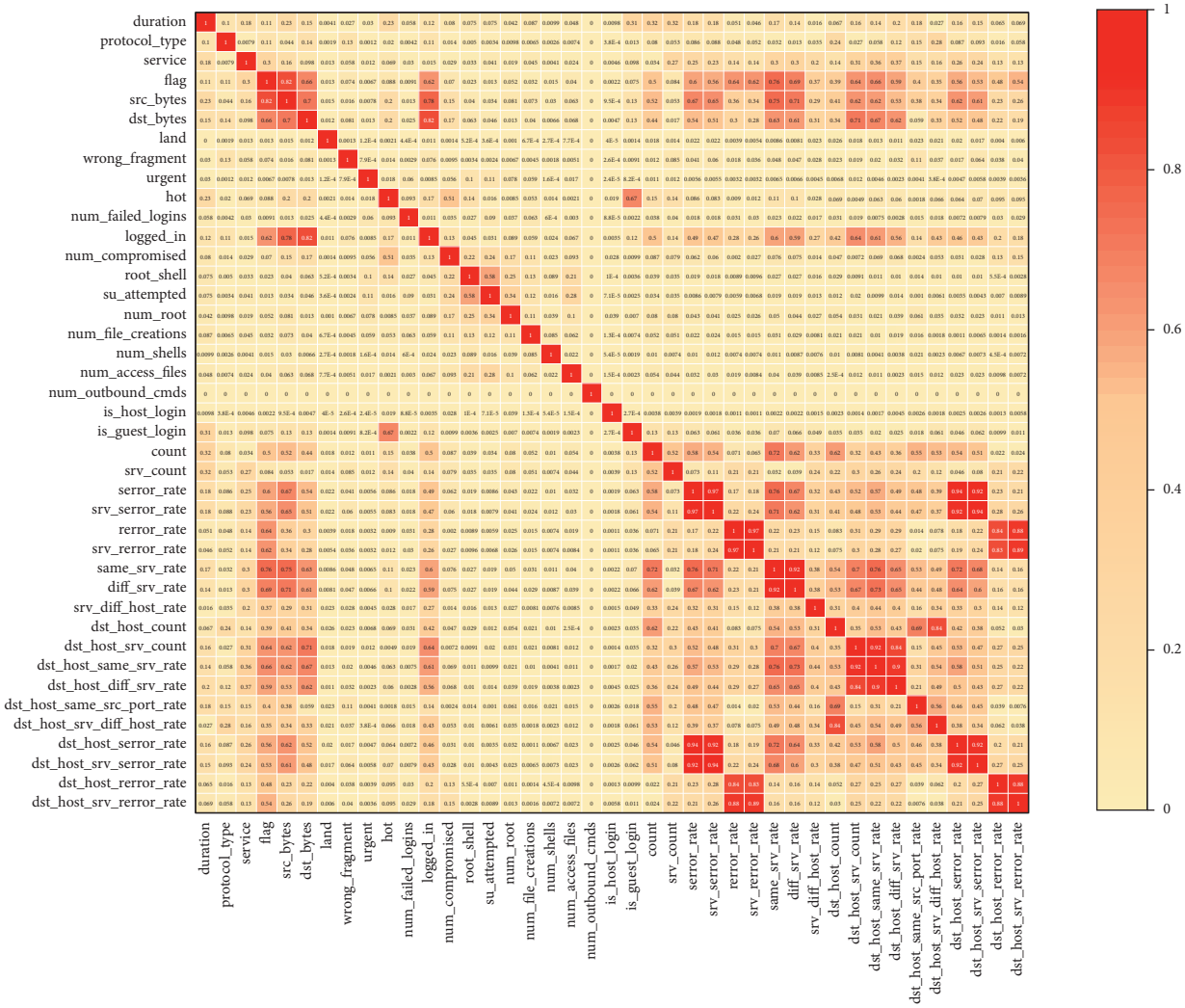


Figure 9: Feature correlation diagram of NSL-KDD.

to verify the superiority of the FCWGAN data enhancement algorithm at network intrusion detection.

Under the same experimental conditions, ROS, ADASYN, SMOTE, WGAN, and the proposed FCWGAN method were used for data enhancement on the NSL-KDD and UNSW-NB15 datasets, respectively, using BiLSTM as a classifier, with test results as shown in Tables 9 and 10.

From Tables 9 and 10, it can be seen that the proposed FCWGAN-BiLSTM achieved the best test results in terms of accuracy, precision, recall, and F1-score. Overall, FCWGAN was better for data enhancement. The time in the table is the training time of a single epoch. It can be found that the training time of the model in this paper is lower than that of other methods, indicating that the calculation speed of the model is the fastest and the calculation cost is the smallest. This is because ROS only performs a simple resampling of the original data, ADASYN and SMOTE perform a random synthesis of the original data based on the k-nearest neighbor principle, and neither learns the nature of the original data. In contrast, FCWGAN, which is based on deep learning, can acquire the potential distribution of the original data, randomly connect the data points with class labels, and pass them to the generator to generate new minority samples. Compared with WGAN, FCWGAN adds feature selection and simplifies the data structure, which calculation cost is reduced and the calculation speed is accelerated. At the same time, a gradient penalty term solves the vanishing gradient problem during training, so that FCWGAN can generate minority class samples that have higher quality and are more similar to the original samples.

4.3.5. Comparative Experiments with Different Classification Algorithms. We performed comparison experiments to verify that BiLSTM could achieve better results for the classification of network intrusions.

Under the same experimental conditions, the dataset was processed using FCWGAN, and was then trained on RF, DNN, LSTM, and BiLSTM. The results of different algorithms for network intrusion behavior were evaluated, and the results are shown in Tables 11 and 12.

From Tables 11 and 12, it can be seen that the proposed FCWGAN-BiLSTM achieved the best results in terms of accuracy, precision, recall, and $F1$-score. Moreover, BiLSTM

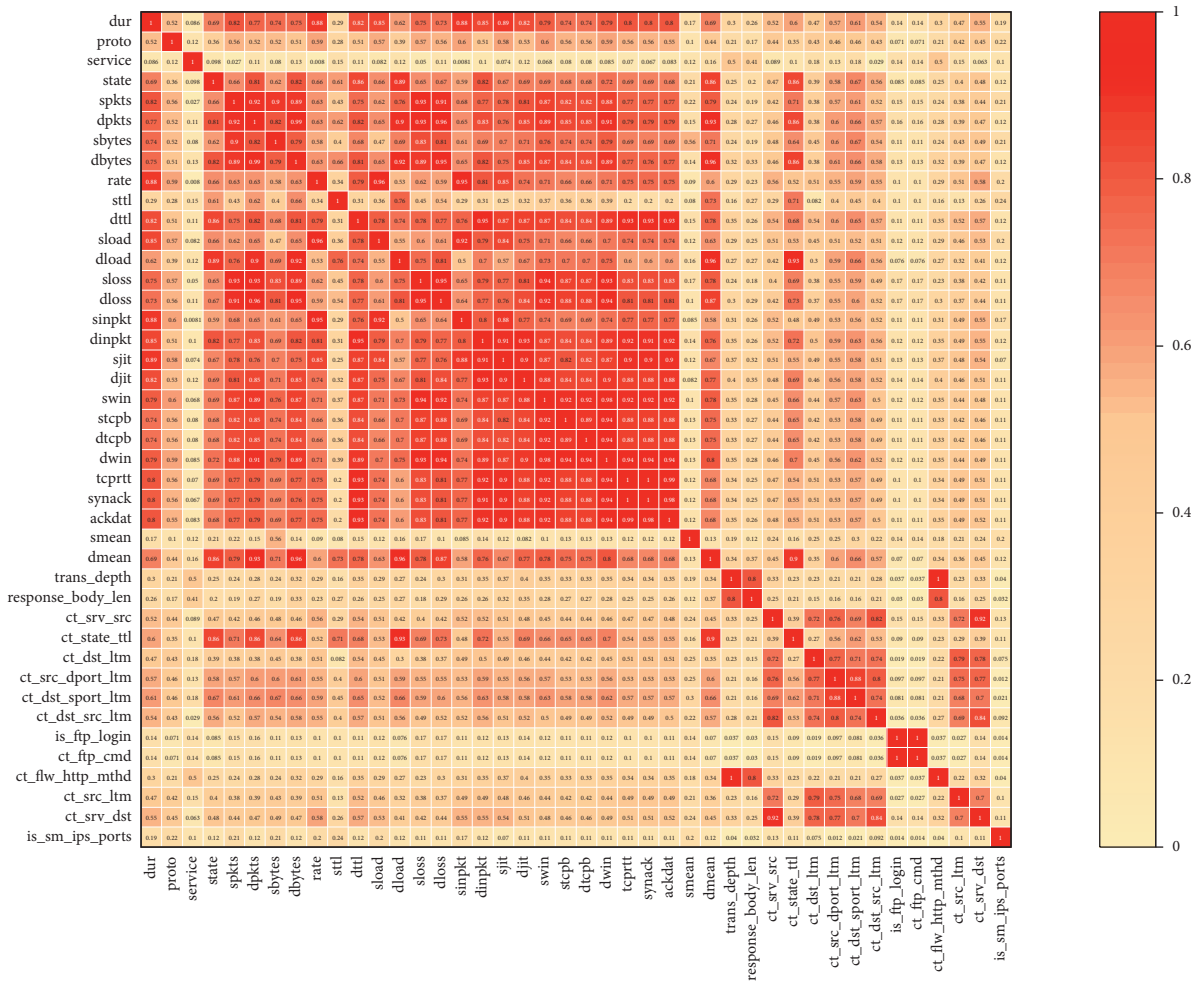


Figure 10: Feature correlation diagram of UNSW-NB15.

Table 4: Feature selection results.

Dataset	Feature selection	Number
NSL-KDD	duration, protocol_type, service, dst_host_srv_count, src_bytes, dst_host_count, dst_bytes, count, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, srv_count, dst_host_rerror_rate, dst_host_serror_rate, diff_srv_rate, srv_diff_host_rate, hot, serror_rate, rerror_rate, num_compromised, num_root	20
UNSW-NB15	dur, sload, ct_srv_src, sbytes, stcpd, ct_src_ltm, tcprtt, ct_srv_dst, ct_dst_src_ltm, ct_dst_ltm, djit, sjit, dload, smean, ct_src_dport_ltm, dmean, service, proto, response_body_len	19

Table 5: Distribution of NSL-KDD dataset before and after sample generation.

Class	Before sample generation	After sample generation
Normal	67343	67343
DoS	45927	45927
Probe	11656	11656
R2L	995	5995
U2R	52	5052

has advantages in network intrusion detection problem. The reason is that network traffic data have obvious time-series characteristics, while LSTM and BiLSTM have strong time-series processing capability and could perform deeper feature extraction on long-term time-series data. Therefore, this type of method can achieve good results at network intrusion detection. LSTM could only read sequence data from one direction and could not rule out the influence of subsequent information on the detection results. Thus, BiLSTM is used to process the incoming data to improve the

Table 6: Distribution of UNSW-NB15 dataset before and after sample generation.

Class	Before sample generation	After sample generation
Normal	56000	56000
Generic	40000	40000
Exploits	33393	33393
Fuzzers	18184	18184
DoS	12264	12264
Reconnaissance	10491	10491
Analysis	2000	7000
Backdoor	1746	6746
Shellcode	1133	6133
Worms	130	5130

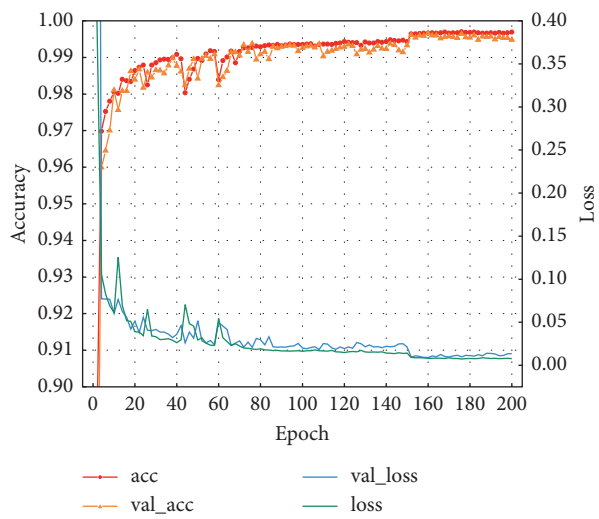


Figure 11: NSL-KDD accuracy curve.

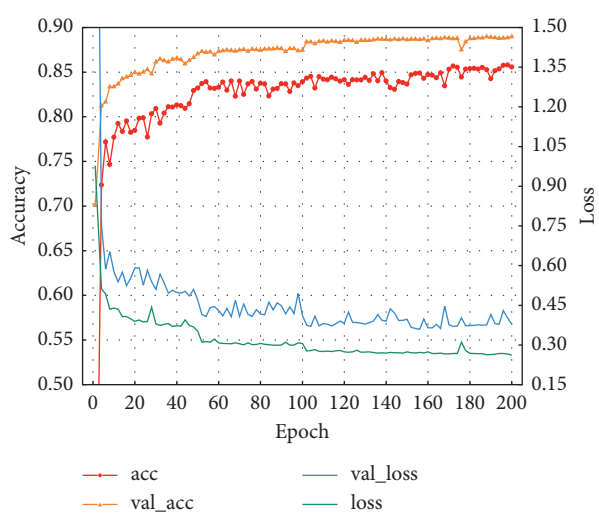


Figure 12: UNSW-NB15 accuracy curve.

detection effect. However, the training time of a single epoch of the model is higher than that of other detection methods, because BiLSTM is more complex than other classification algorithms.

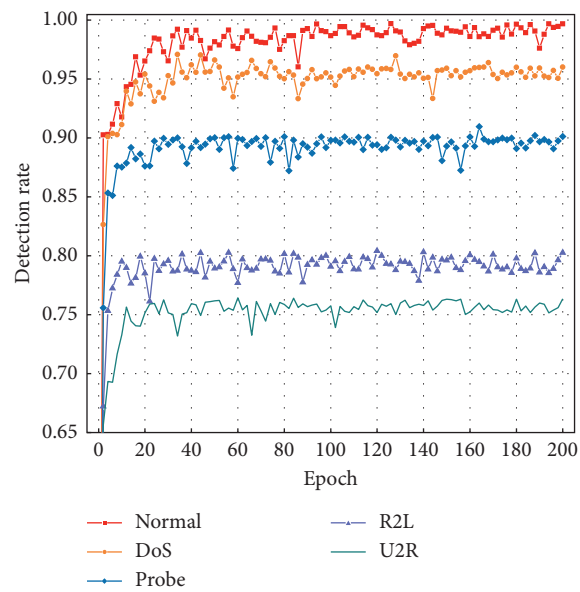


Figure 13: NSL-KDD class detection rate curve.

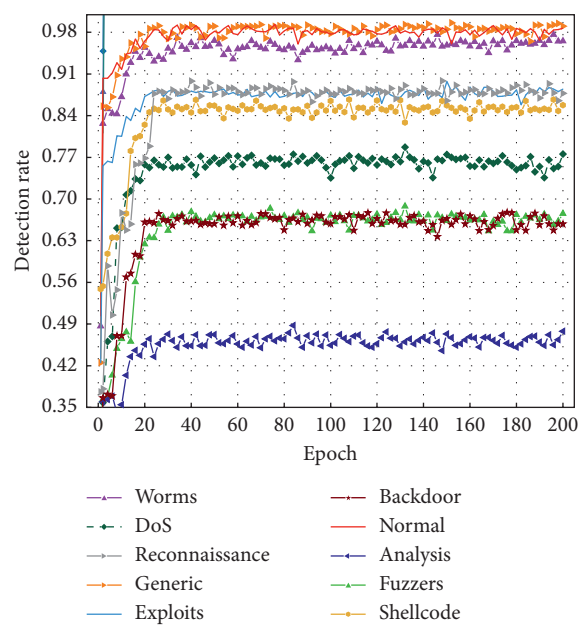


Figure 14: UNSW-NB15 class detection rate curve.

4.3.6. Comparative Experiment with Existing Intrusion Detection Models. Performance comparison experiments were conducted to further verify the comprehensive performance of network intrusion detection algorithms based on FCWGAN and BiLSTM.

Under the same experimental conditions, models with superior detection results in the literature, such as CNN-BiLSTM [35], SSAE-LSTM [36], CWGAN-DNN, and AE-CGAN-RF, were applied to the NSL-KDD and UNSW-NB15 datasets in accordance with their published descriptions and parameter settings, with results as shown in Tables 13 and 14.

Table 7: Detection accuracy under the influence of different noise levels.

Dataset	Noise level				
	0	0.02	0.04	0.06	0.08
NSL-KDD	99.57 ± 0.21	99.55 ± 0.22	99.45 ± 0.22	98.88 ± 0.24	98.27 ± 0.25
UNSW-NB15	85.59 ± 0.27	85.53 ± 0.29	85.28 ± 0.30	84.71 ± 0.31	84.15 ± 0.33

Table 8: Ablation experiment detection rate of various types.

Algorithm	Type of samples				
	Normal	DoS	Probe	U2R	R2L
BiLSTM	94.65 ± 0.21	88.24 ± 0.19	72.91 ± 0.23	46.81 ± 0.35	51.97 ± 0.30
GAN-BiLSTM	95.31 ± 0.25	92.18 ± 0.22	81.27 ± 0.30	60.33 ± 0.42	65.10 ± 0.37
CWGAN-BiLSTM	98.54 ± 0.19	94.60 ± 0.15	85.15 ± 0.23	70.20 ± 0.26	72.13 ± 0.25
Model in this paper	99.68 ± 0.14	96.01 ± 0.11	90.12 ± 0.15	76.35 ± 0.27	80.26 ± 0.19

Table 9: Comparison of data enhancement algorithms on NSL-KDD dataset.

Algorithm	Evaluation metrics				
	Accuracy	Precision	Recall	$F1$-score	Time (s)
ROS-BiLSTM	89.18 ± 0.35	90.34 ± 0.40	88.61 ± 0.35	89.46 ± 0.37	4
ADASYN-BiLSTM	92.95 ± 0.24	93.12 ± 0.27	92.61 ± 0.21	92.86 ± 0.25	5
SMOTE-BiLSTM	93.66 ± 0.28	94.63 ± 0.34	93.14 ± 0.26	93.88 ± 0.30	3
WGAN-BiLSTM	96.56 ± 0.23	96.71 ± 0.28	95.65 ± 0.21	96.20 ± 0.26	7
Model in this paper	99.57 ± 0.21	99.55 ± 0.20	99.47 ± 0.17	99.51 ± 0.18	2

Table 10: Comparison of data enhancement algorithms on UNSW-NB15 dataset.

Algorithm	Evaluation metrics				
	Accuracy	Precision	Recall	$F1$-score	Time (s)
ROS-BiLSTM	81.70 ± 0.43	79.32 ± 0.47	80.49 ± 0.41	79.90 ± 0.44	6
ADASYN-BiLSTM	83.65 ± 0.37	84.11 ± 0.40	82.14 ± 0.35	83.12 ± 0.37	6
SMOTE-BiLSTM	83.66 ± 0.31	84.28 ± 0.34	81.24 ± 0.27	82.73 ± 0.30	5
WGAN-BiLSTM	81.49 ± 0.30	84.71 ± 0.24	82.51 ± 0.28	83.60 ± 0.26	8
Model in this paper	85.59 ± 0.27	86.11 ± 0.21	85.57 ± 0.24	85.84 ± 0.22	4

Table 11: Comparison of classification algorithms on NSL-KDD dataset.

Algorithm	Evaluation metrics				
	Accuracy	Precision	Recall	$F1$-score	Time (s)
FCWGAN-RF	91.29 ± 0.27	90.24 ± 0.29	89.11 ± 0.21	89.67 ± 0.24	1
FCWGAN-DNN	95.11 ± 0.23	96.01 ± 0.22	94.98 ± 0.17	95.00 ± 0.19	2
FCWGAN-LSTM	98.29 ± 0.23	98.37 ± 0.21	98.14 ± 0.15	98.25 ± 0.18	2
Model in this paper	99.57 ± 0.21	99.55 ± 0.20	99.47 ± 0.17	99.51 ± 0.18	2

Table 12: Comparison of classification algorithms on UNSW-NB15 dataset.

Algorithm	Evaluation metrics				
	Accuracy	Precision	Recall	$F1$-score	Time (s)
FCWGAN-RF	81.00 ± 0.37	81.94 ± 0.33	80.97 ± 0.31	81.45 ± 0.32	1
FCWGAN-DNN	83.44 ± 0.31	84.12 ± 0.33	83.40 ± 0.27	83.76 ± 0.30	2
FCWGAN-LSTM	84.98 ± 0.30	85.44 ± 0.29	84.67 ± 0.25	85.05 ± 0.28	3
Model in this paper	85.59 ± 0.27	86.11 ± 0.21	85.57 ± 0.24	85.84 ± 0.22	4

Table 13: Comparison of multiclassification on NSL-KDD dataset.

Algorithm	Evaluation metrics				
	Accuracy	Precision	Recall	$F1$-score	Time (s)
CNN-BiLSTM	99.22 ± 0.31	99.18 ± 0.29	99.14 ± 0.24	99.15 ± 0.26	6
SSAE-LSTM	97.63 ± 0.34	97.91 ± 0.33	97.21 ± 0.28	97.56 ± 0.30	4
CWGAN-DNN	98.13 ± 0.26	99.03 ± 0.30	97.91 ± 0.25	98.46 ± 0.27	8
AE-CGAN-RF	98.53 ± 0.27	98.67 ± 0.28	98.31 ± 0.23	98.49 ± 0.25	7
Model in this paper	99.57 ± 0.21	99.55 ± 0.20	99.47 ± 0.17	99.51 ± 0.18	2

Table 14: Comparison of multiclassification on UNSW-NB15 dataset.

Algorithm	Evaluation metrics				
	Accuracy	Precision	Recall	$F1$-score	Time (s)
CNN-BiLSTM	82.08 ± 0.43	82.68 ± 0.43	80.00 ± 0.37	81.32 ± 0.40	10
SSAE-LSTM	82.31 ± 0.45	83.65 ± 0.44	81.94 ± 0.36	82.78 ± 0.41	7
CWGAN-DNN	82.61 ± 0.37	82.95 ± 0.41	82.11 ± 0.33	82.53 ± 0.38	14
AE-CGAN-RF	81.24 ± 0.39	83.47 ± 0.40	80.31 ± 0.35	81.86 ± 0.38	13
Model in this paper	85.59 ± 0.27	86.11 ± 0.21	85.57 ± 0.24	85.84 ± 0.22	4

From Tables 13 and 14, it can be seen that the proposed model achieved the best detection results on all metrics. Compared with CNN-BiLSTM and SSAE-LSTM, the proposed model uses FCWGAN to simplify the data features and reduce dataset dimensionality, which reduces the computational cost, while generating minority class samples to supplement the dataset, which alleviates the impact of class imbalance, and thus it could obtain better detection results. Compared with CWGAN-DNN and AE-CGAN-RF, the proposed model eliminates high-dimensional disasters and simplifies the data structure, while uses BiLSTM for feature extraction and classification, which can extract more in-depth and comprehensive data features from the time-series level, and thus obtains better multiclassification results.

5. Conclusion

To alleviate the impact of class imbalance on the accuracy of network intrusion detection models and improve their effectiveness at detecting network intrusion attacks, we proposed a network intrusion method based on FCWGAN and BiLSTM. The method uses XGBoost and Spearman correlation coefficients to process the dataset, which effectively filters out redundant and useless features, simplifies the data structure, which reduces the computational cost and training time, and avoids high-dimensional disasters. Minority class samples are generated using CWGANs to supplement the dataset and alleviate class imbalance. BiLSTM is used to extract the time-series features of data to complete the classification of network intrusions. Extensive experiments on the NSL-KDD and UNSW-NB15 datasets demonstrated that the model greatly improves the detection effect for minority class samples, has a strong feature extraction capability, high detection accuracy, and low false-positive rate when processing large-scale network data, and shows promise for real-time intrusion detection systems. However, the accuracy of this model on the UNSW-NB15 dataset demonstrated that there is room for improvement. Future work will focus on this deficiency, and we will investigate the construction of feature extraction and classification models to find ways to improve detection accuracy.

Data Availability

All data used in this paper can be obtained by contacting the authors of this study.

Ethical Approval

This article does not contain any studies with human or animal subjects performed by any of the authors.

Consent

Informed consent was obtained from all individual participants included in the study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61703426, 61806219, and 61876189), Youth Talent Promotion Plan of Shaanxi University Association for Science and Technology (no. 20190108), and Shaanxi Innovation Capability Support Plan (no. 2020KJXX-065).

References

[1] M. Yousefnezhad, J. Hamidzadeh, and M. Aliannejadi, "Ensemble classification for intrusion detection via feature extraction based on deep Learning," *Soft Computing*, vol. 25, no. 20, Article ID 12667, 2021.

[2] M. Kelidari and J. Hamidzadeh, "Feature selection by using chaotic cuckoo optimization algorithm with levy flight, opposition-based learning and disruption operator," *Soft Computing*, vol. 25, no. 4, pp. 2911–2933, 2020.

[3] D. Gonzalez-Cuautle, A. Hernandez-Suarez, G. Sanchez-Perez et al., "Synthetic minority oversampling technique for optimizing classification tasks in botnet and intrusion-detection-system datasets," *Applied Sciences*, vol. 10, no. 3, p. 794, 2020.

[4] J. Lee and K. Park, "AE-CGAN model based high performance network intrusion detection system," *Applied Sciences*, vol. 9, no. 20, p. 4221, 2019.

[5] J. Lee and K. Park, "GAN-based imbalanced data intrusion detection system," *Personal and Ubiquitous Computing*, vol. 25, no. 1, pp. 121–128, 2019.

[6] X. Liu, T. Li, R. Zhang, D. Wu, Y. Liu, and Z. Yang, "A GAN and feature selection-based oversampling technique for intrusion detection," *Security and Communication Networks*, vol. 2021, Article ID 9947059, 15 pages, 2021.

[7] J. He, "CWGAN-DNN: a method of conditional Wasserstein generation against network intrusion detection," *Journal of Air Force Engineering University (NATURAL SCIENCE EDITION)*, vol. 22, no. 5, pp. 67–74, 2021.

[8] A. Thakkar and R. Lohiya, "Attack classifcation using feature selection techniques: a comparative study," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, 2021.

[9] M. Di Mauro, G. Galatro, G. Fortino, and A. Liotta, "Supervised feature selection techniques in network intrusion detection: a critical review," *Engineering Applications of Artificial Intelligence*, vol. 101, Article ID 104216, 2021.

[10] S. M. Z. Kashani and J. Hamidzadeh, "Feature selection by using privacy-preserving of recommendation systems based on collaborative filtering and mutual trust in social networks," *Soft Computing*, vol. 24, no. 15, Article ID 11425, 2019.

[11] X. Li, P. Yi, W. Wei, Y. Jiang, and L. Tian, "Lnnls-KH: A Feature Selection Method for Network Intrusion Detection," *Security and Communication Networks*, vol. 2021, Article ID 8830431, 22 pages.

[12] B. Selvakumar and K. Muneeswaran, "Firefly algorithm based feature selection for network intrusion detection," *Computers & Security*, vol. 81, pp. 148–155, 2019.

[13] T. Le, Y. Kim, and H. Kim, "Network intrusion detection based on novel feature selection model and various recurrent neural networks," *Applied Sciences*, vol. 9, no. 7, p. 1392, 2019.

[14] B. S. Bhati, G. Chugh, F. Al-Turjman, and N. S. Bhati, "An improved ensemble based intrusion detection technique using XGBoost," *Transactions on emerging telecommunications technologies*, vol. 32, no. 6, 2021.

[15] S. T. Ikram, A. K. Cherukuri, B. Poorva et al., "Anomaly Detection Using XGBoost Ensemble of Deep Neural Network Models," *Cybernetics and information technologies*, vol. 21, no. 3, 2021.

[16] G. P. Dubey and D. R. K. Bhujade, "Optimal feature selection for machine learning based intrusion detection system by exploiting attribute dependence," *Materials Today Proceedings*, vol. 47, pp. 6325–6331, 2021.

[17] D. Liao, S. Huang, Y. Tan, and G. Bai, "Network Intrusion Detection Method Based on GAN Model," in *Proceedings of the 2020 International Conference on Computer Communication and Network Security (CCNS)*, August 2020.

[18] S. Huang and K. Lei, "IGAN-IDS: an imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks," *Ad Hoc Networks*, vol. 105, Article ID 102177, 2020.

[19] H. Chen and L. Jiang, "Efficient GAN-based Method for Cyber-Intrusion Detection," 2019, https://arxiv.org/abs/1904.02426.

[20] J. Ye, Y. Fang, and J. Ma, "Intrusion Detection Model Based on Conditional Generative Adversarial Networks," in *Proceedings of the 2019 Second International Conference on Algorithms, Computing and Artificial Intelligence*, ACM, Sanya, China, December 2019.

[21] S. Msika, A. Quintero, and F. Khomh, "SIGMA: Strengthening IDS with GAN and Metaheuristics Attacks," 2019, https://arxiv.org/abs/1912.09303.

[22] R. Ahsan, W. Shi, X. Ma, and W. L. Croft, "A Comparative Analysis of CGAN-based Oversampling for Anomaly Detection," *IET Cyber-Physical Systems: Theory & Applications*, vol. 7, no. 6, 2021.

[23] G. Zhang, X. Wang, R. Li, Y. Song, J. He, and J. Lai, "Network intrusion detection based on conditional Wasserstein generative adversarial network and cost-sensitive stacked autoencoder," *IEEE access*, vol. 8, Article ID 190431, 2020.

[24] R. Feng, "Uncertainty analysis in well log classification by Bayesian long short-term memory networks," *Journal of Petroleum Science and Engineering*, vol. 205, 2021.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] S. A. Althubiti, E. M. Jones, and K. Roy, "LSTM for anomaly-based network intrusion detection," in *Proceedings of the 2018 28th International Telecommunication Networks and Applications Conference (ITNAC)*, November 2018.

[27] P. Sun, P. Liu, Q. Li et al., "DL-IDS: Extracting Features Using CNN-LSTM Hybrid Network for Intrusion Detection System," *Security and communication networks*, vol. 2020, pp. 1–11, Article ID 8890306, 2020.

[28] S. Hao, J. Long and Y. Yingchuan, BL-IDS: "Detecting Web Attacks Using Bi-LSTM Model Based on Deep Learning," in *Proceedings of the Second EAI International Conference, SPNCE 2019*, Tianjin, China, April 2019.

[29] P. Devan and N. Khare, "An efficient XGBoost–DNN-based classification model for network intrusion detection system," *Neural Computing & Applications*, vol. 32, no. 16, Article ID 12499, 2020.

[30] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, Article ID 107247, 2020.

[31] M. Usama, M. Asim, S. Latif, J. Qadir, and A.-A. Fuqaha, "Generative Adversarial Networks for Launching and Thwarting Adversarial Attacks on Network Intrusion Detection Systems," in *Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, Tangier, Morocco, June 2019.

[32] M. Ishaque and L. Hudec, "Feature extraction using deep learning for intrusion detection system," in *Proceedings of the 2019 Second International Conference on Computer Applications & Information Security*, May 2019.

[33] C. Sarika and K. Nishtha, "Analysis of KDD-cup'99, NSL-KDD and UNSW-NB15 datasets using deep learning in IoT," *Procedia Computer Science*, vol. 167, 2020.

[34] X Zhu and X Wu, "Class noise vs. attribute noise: a quantitative study," *Artificial Intelligence Review*, vol. 22, pp. 177–210, 2004.

[35] S. Jay and M. Manollas, "Effective deep CNN-BiLSTM model for network intrusion detection," in *Proceedings of the Third International Conference on Artificial Intelligence and Pattern*

Recognition in 2020 2020, p. 9, Xiamen, Fujian, China, June 2020.

[36] Y. Lin, J. Wang, Y. Tu, L. Chen, and Z. Dou, "Time-related Network Intrusion Detection Model: A Deep Learning Method," in *Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, December 2019.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 2017366, 10 pages
https://doi.org/10.1155/2022/2017366

Research Article

Resiliency Assessment of Power Systems Using Deep Reinforcement Learning

Mariam Ibrahim,[1] **Ahmad Alsheikh**,[1,2] **and Ruba Elhafiz**[1]

[1]*Department of Mechatronics Engineering, German Jordanian University, Amman 11180, Jordan*
[2]*Department of Natural Science & Industrial Engineering, Deggendorf Institute of Technology, Deggendorf 94469, Germany*

Correspondence should be addressed to Mariam Ibrahim; mariam.wajdi@gju.edu.jo

Received 1 February 2022; Accepted 19 March 2022; Published 7 April 2022

Academic Editor: Konstantinos Demertzis

Evaluating the resiliency of power systems against abnormal operational conditions is crucial for adapting effective actions in planning and operation. This paper introduces the level-of-resilience (LoR) measure to assess power system resiliency in terms of the minimum number of faults needed to produce a system outage (blackout) under sequential topology attacks. Four deep reinforcement learning (DRL)-based agents: deep Q-network (DQN), double DQN, the REINFORCE (Monte-Carlo policy gradient), and REINFORCE with baseline are used to determine the LoR. In this paper, three case studies based on IEEE 6-bus test system are investigated. The results demonstrate that the double DQN network agent achieved the highest success rate, and it was the fastest among the other agents. Thus, it can be an efficient agent for resiliency evaluation.

1. Introduction

The deployment of recent technologies in communication, computing, and control of smart grids can be suitable for clients and electrical facilities. Energy infrastructures are natively connected to other areas of demanding infrastructures, and their supply breaking can have disastrous cascading results [1]. One of the important features that is essential in today's smart grids is to run resiliently when attacks/faults and other contingencies occur.

Determining the resilience of power systems (PSs) has been a subject of concern in latest years. Stochastic and statistical analysis techniques are used to evaluate power system resilience [2]. While these techniques can aid understanding the system resilience to large-scale contingencies, however, they are not always appropriate when evaluating resilience in the presence of malicious sources. These methods are based on the comparably simple DC model, which does not consider effects like voltage breakdown that may happen during a cascade. Also, there is a need to enhance their data, sampling ways, and the extent of models and effects represented [3]. Accordingly, it is essential to investigate new approaches to evaluate the resilience of the grid using the more realistic and scalable AC models.

The applications of machine learning (ML) algorithms are identified by Olowononi et al. [4] in the field of security and resiliency of the power grid. Their target is to effectively survey the interactions among resilient grid using ML and resilient ML when used in the grid. The power system's cybersecurity and ML have a wide range of interdisciplinary crossways between them. For instance, reinforcement and deep learning (DL) can be used to build smart models for applying malware classification, observing the use of the intrusion detection and prevention systems (IDS/IPS), and implementing threat intelligence sensing [5].

Reinforcement learning (RL) is one of the established ML approaches [6]. RL does not depend directly on data sets but has an agent that is placed in an anonymous environment and can receive feedbacks in form of rewards by making actions that can result in maximizing cumulative rewards, so the agent learns from its own experience. The agent focuses on finding an optimal policy rather than analyzing data as compared to supervised and unsupervised

learning. The environment usually has dynamics that are unknown to the agent.

The DL approaches grant computational models that are created of numerous processing layers to learn representations of data with various levels of abstraction. These approaches have effectively enhanced the visual object recognition, speech recognition, and numerous realms [7]. The combination of RL with DL techniques (DRL) is most useful in problems with a high dimensional state-space which makes it suitable for evaluating the resilience of power systems. Classical RL techniques has a complex design issue in the decision of features. Nonetheless, DRL has been rewarding in difficult assignments with a lower prior knowledge [8]. The recent advancement in DL techniques is summarized by Dick et al. [1] for creating machine vision models. The current applications of this technology are also investigated to improve the resiliency of critical infrastructure protection (CIP).

Several works investigated the cybersecurity of power grids using RL and DRL. For instance, Dibaji et al. [9] considered cyber physical systems' security from systems and control perspectives in general, and shortly discussed the possibilities of using RL and DRL to this purpose. Q-learning was proposed by Yan et al. [10] to interpret the transmission grid vulnerability against sequential topology attacks and determine critical attack sequences taking into account physical system behaviors. A modified Q-learning (termed the nearest sequence memory Q-learning) was adopted by Wang et al. [11] to evaluate threats imposed by false data injection attack on voltage control of a power system. Test results revealed that even if a few substations are attacked, a voltage collapse with its consequences can happen in the system.

Secure state estimation using multiagent reinforcement learning was dealt by He et al. [12] with the assumption that measurements are sent over a wireless network under jamming attacks. The antijamming game framework was used to determine the optimal path against an intelligent attacker. He et al. [13] considered secure-state estimation with risk-averse transmission path selection method that is based on RL concept. They demonstrated how the proposed approach can improve secure-state estimation robustness.

The use of RL was discussed by Oozeer et al. [14] in a general framework of cognitive risk control for cyber-attacks in smart grids. RL was presented by Chen et al. [15] to evaluate false data injection attacks against automatic voltage control of power systems (in normal operating states). A Q-learning algorithm with the nearest sequence memory was employed for online learning of attacking strategy. The optimal attack strategy was modelled as a partially observable Markov decision process. Based on kernel density estimation, a bad data detection and correction technique was presented to reduce the disruptive influence of the attacks. Table 1 shows some recent studies that were performed on smart grid system security using RL and DRL.

The novelty of this work lies in evaluating power system's resiliency level (LoR) under sequential topological attacks/faults using DRL techniques. The framework design methodology is based on using four DRL agents which are trained and optimized with the aim of determining the minimum number of faults required to black out the system. This number is used to determine the LoR for three different topologies of IEEE 6-bus system case study under single and three-phase attack scenarios. The performance of the tested DRL agents was compared. The double DQN agent was stable and achieved the highest success rate among all agents. Thus, it can be used for resilience studies that investigate the system's ability to withstand attacks/faults by aiding system designers to select the most resilient system's topology. The rest of the paper is straightened out as follows: Section 2 illustrates power system's topologies along with the attack/faults scenarios. Section 3 presents the resiliency measure formulation and the DRL techniques. Experimental results are shown in Section 4. Section 5 summarizes and presents certain future directions.

1.1. Acronyms and Notations. Table 2 illustrates the acronyms and notations used through the paper.

2. Preliminaries

2.1. Electric Power Grid Topology. An electrical power grid is a complementary network for carrying electricity from producers to consumers. Electrical grids differ in size from serving whole countries through national grids to cross-continents through transnational grids [21]. Three power system topologies were considered in this paper. These are PS1, PS2, and PS3, respectively, as shown in Figures 1 to 3. They have identical buses, generation, and load units. Each system is a three-phase electric power system that consists of three loads (each has an active power of 70 Mw), three generators (two photovoltaic (PV) generators and one swing) with active power of 50 Mw for each, six buses and 36 transmission lines. The power system PS1 is an IEEE 6-bus system introduced by Kennedy [22]. PS2 was generated by altering PS1's topology, while PS3 can be described as a fully connected system where all the RLC circuits are connected to each other.

In PS1, PS2, and PS3, the loads L1, L2, and L3 are connected to buses 4, 5, and 6, respectively. Nonetheless, the generators Swing, PV1, and PV2 are connected to buses 1, 2, and 3, respectively. The values of RLC of lines are also equal in all the three grids. The topology differences can be shown in the transmission line connections which resulted in altering the potential paths of current flow.

2.2. Faults Scenarios. Typically, a power system performs well under balanced conditions. However, the system might become unbalanced due to several reasons, such as natural disturbances (e.g., earthquakes, lightning, and high-speed winds), tree falling on the lines, and insulation failure. These reasons can lead to short-circuits or a fault in the lines [22]. The most harming faults in power systems are short-circuit faults because their occurrence can result into a significant increase in the electrical current. Nonetheless, there exists

Table 1: Recent studies on smart grid system security using RL and DRL.

Reference	System	Method	Attack	Recovery action	Aim	Limitations
[16]	Modified 9-bus system	Deep deterministic policy gradient (DDPG)	Multiswitch attacks and false data injection (FDI) attacks	Reclose the transmission lines lost in the cyber-attack by optimizing the reclosing time.	Reach the asynchrony in the power system by applying power blocking which will accelerate/decelerate the rotors of the generators	Owing to its continuous action space, it will not be suitable for topological resilience studies
[17]	IEEE 9, 14 and 30-bus systems	Deep Q-network (DQN)	Data integrity attacks	No recovery action	Evaluate the delay-alarm error rates, false-alarm error rates, and detect-failure rates for the systems	
[18]	IEEE 30-bus system	Deep Q-network (DQN)	Coordinated cyber physical topology (CCPT) attacks	Control center can detect the line outage by using phasor measurement units (PMU) data	Investigate the coordinated topology attacks in smart grid which combine a physical topology attack and a cyber-topology attack	DQN suffers from overestimation
[19]	Wood & Wollenberg 6-bus system and IEEE 30-bus system	Q-learning	Sequential attacks	Automatic generation control (AGC)	Identify the minimum number of attacks/actions to reach blackout threshold	
[20]	IEEE 14-bus system	SARSA	False data injection (FDI), jamming, and denial of service (DoS) attacks	No recovery action	Formulation an online cyber-attack detection as a POMDP problem and propose a solution based on the model-free RL for POMDPs	Q-learning and SARSA techniques are limited to systems with small state-action space
Our work	IEEE 6-bus system	Deep Q-network (DQN), double DQN, REINFORCE, and REINFORCE with baseline	Sequential attacks	Disconnecting the faulted transmission lines	Evaluating the resiliency of power systems against faults/attacks using DRL	Needs to investigate tabular methods such as Q-learning and SARSA to compare their performance with DRL methods

two types of short-circuit faults: symmetric and asymmetric [23].

In a symmetric fault, all the phases are short-circuited to each other and often to earth. Such a fault is balanced in the sense that the system remains symmetrical, or in other words, the lines are displaced by an equal angle. It is the most relentless type of faults, including the largest current. Yet, it rarely materializes [24], such as a three-phase line to the ground fault (L–L–L–G) where the fault occurs between the three phases and the ground of the system. The asymmetrical fault gives rise to asymmetrical current, that is, the current is differing in magnitude and phase in the three phases of the power system. When a short-circuit occurs, the current comes into its peak value rapidly, and then it reduces exponentially with time through three different states: sub-transient, transient, and permanent states [25]. Examples of asymmetrical faults are single line-to-ground (L–G) fault, line-to-line fault (L–L), and double line-to-ground (L–L–G) fault. In this work, the asymmetric (L–L–G) and symmetric (L–L–L–G) faults were considered against the three topologies.

3. Resiliency Measure and DRL Techniques

3.1. Resiliency Measure Formulation. LoR is the factor that is employed to hold the evolution of system's features through the variations of system's modes of operation under a sequence of fault and recovery actions. For a number of PSs under a sequence of faults/attacks (an attack scenario), suppose the resulting system modes are represented by $Z_0 \longrightarrow Z_1 \longrightarrow \ldots \longrightarrow Z_m$, where Z_0 is the initial mode, while Z_h is the mode after the hth fault and reconfiguration ($h = 1, \ldots, m$). A power system is more resilient if it needed a larger number of faults/attacks N over all possible attack scenarios M before its outage. This factor can be determined by using a reinforcement agent who finds the optimal number of faults

Table 2: Acronyms and notations used.

Category	Items/ symbols	Description
Acronyms	LoR	Level-of-resilience
	PS	Power system
	DRL	Deep reinforcement learning
	DQN	Deep Q-network
	ML	Machine learning
	CIP	Critical infrastructure protection
	PV	Photovoltaic generator
	DDPG	Deep deterministic policy gradient
	FDA	False data injection
	Q value	State-action value
	(L–G)	Single line-to-ground fault
	(L–L)	Line-to-line fault
	(L–L–G)	Double line-to-ground
Notations	$\pi(S)$	Agent's policy
	V(S)	Value function
	R	Reward
	G_t	Return
	γ	The discounting factor
	S	State
	A	Action
	ϵ	Probability of selecting an action
	θ, θ^{-}	Weights
	y_j	The value function target
	$\nabla_\theta J(\theta)$	Gradient
	$\pi_\theta(A\|S)$	Parameterized function with respect to θ
	$\delta(S, A)$	The advantage function
	$\mu(S)$	Actor policy
	S_T	Terminal state
	α, β	The learning rates
	Z_h	The mode after hth fault and reconfiguration
	M	A set of attack scenarios
	N	Number of faults/attacks

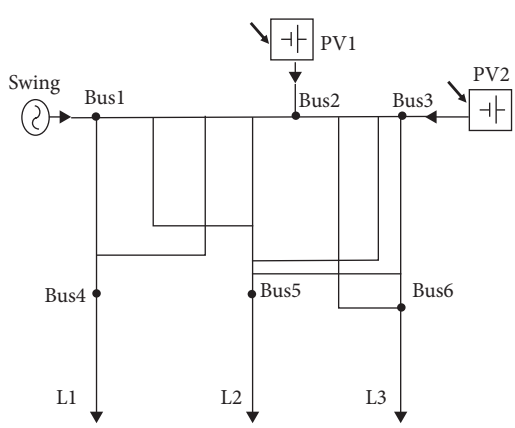


Figure 1: Power system PS1.

(by trial and failure) needed to produce a blackout. This is called an optimal policy.

Definition 1. Given a set of power systems with identical buses, generation, and load units, but with different topologies: PS ≡ ∪ PS_k; $k \in \{1, \ldots, y\}$, where y is the number of

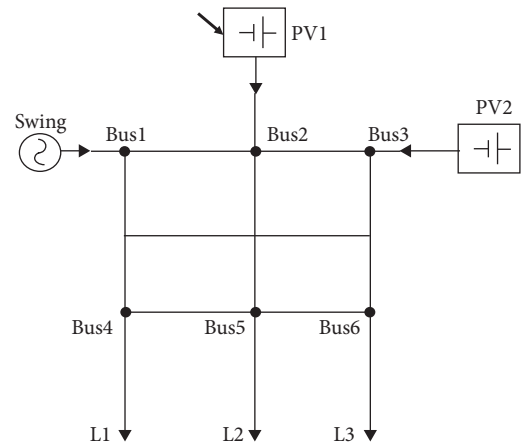


Figure 2: Power system PS2.

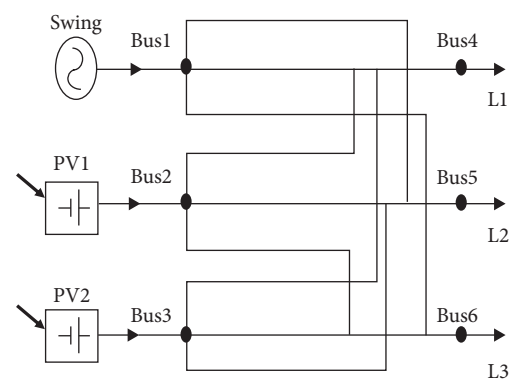


Figure 3: Power system PS3.

the power systems, and a set of attack scenarios M, we say that the $LoR(PS_i) > LoR(PS\text{-}PS_i)$ if:

$$N_{PSi} > N_{PS-PSi}$$

3.2. DRL Algorithms. When the agent begins to learn, the agent will be in a state S of the environment, by selecting an action A, the agent can switch from one state to another. The transition probability between states, that is, P, denotes the probability of the state to which the agent will arrive to. When the agent conducts an action, the environment delivers a reward R as feedback. The model describes the reward function and transition probabilities. The agent's policy $\pi(S)$ provides the strategy on which is the best/optimal action to be taken in a specific state with the aim of maximizing the cumulative rewards. Every state is identified with a value function $V(S)$ predicting the expected number of future rewards that the agent will obtain in this state by choosing an optimal action under the current/other policy. The future reward (also called return) G_t is the total sum of discounted rewards in the future as represented by:

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}, \quad (1)$$

where $\gamma \in [0, 1]$ is the discounting factor which penalizes the rewards in the future, so an agent can focus on the future reward rather than the immediate reward. Both policy and value functions are what the agent tries to learn in RL. The cooperation among the agent and the environment includes a sequence of actions and rewards evolving in time $t = 1, 2, \ldots, T$, where T is time step at which the termination state is reached. During this process, the agent gathers information about the environment and gives decisions on which action to take next to precisely learn the best policy. The state, action, and reward at time step t can be represented as S_t, A_t, and R_t, respectively. Therefore, the full cooperation sequence is represented by one episode (trajectory) and the sequence terminates at the terminal state: $S_1, A_1, R_1, S_2, A_2, R_2, \ldots, S_T$.

DQN was introduced by Mnih et al. [26] through a combination of Q-learning with a function approximator (neural network) to overcome the tabular limit of Q-learning. The algorithm was tested on Atari games and the agent was able to achieve the human level in Atari games. The inputs were raw pixels of the game so that the same agent can learn multiple games with no need for a special processing of the inputs. The past trials of combining Q-learning with function approximators in the past were not successful due to the deadly triad issue [27], where the model suffered from instability and divergence. This issue was solved by improving and stabilizing the training procedure of Q-learning using two methods of experience replay and periodically updated target. Here, DQN is a neural network model that receives states as inputs and produces action values $Q(S; \theta)$ for network parameters θ. The episode step $e_t = (S_t, A_t, R_t, S_{t+1})$ is stored in one replay memory $D_t = \{e_1, \ldots, e_t\}$, where D_t has experienced e_t tuples over many episodes. During Q-learning updates, samples are drawn randomly from the replay memory (called experience replay). Thus, one sample could be used many times. This was useful in reducing the correlation between samples, which resulted in a network that can learn without any overfitting. Moreover, the experience replay could reuse old experience, which resulted in a smooth learning and more efficient tuples samples.

In periodically updated target, DQN keeps a copy of the network with an identical architecture and initializes with the same parameters (weights values). The predicted Q from the target network will be used to update the main Q-network. The target network's parameters are not trained like the main network, instead they are periodically synchronized with the parameters of the main Q-network. The idea behind this is to serve the same goal as the experience buffer by reducing the correlation between samples using different parameters in the main Q-network with θ and θ^- for the target network. Thus, optimizing the Q values towards the target values. This has shown to stabilize the learning. Here, the target network with parameters θ^- is the same as the main Q-network except that its parameters are copied every C time steps. The C steps were chosen to be two steps so that $\theta_t^- = \theta_t$ and are kept fixed in all other steps. The main Q-network goal is to produce an estimation of the Q values for each action that can be taken from that state, but the objective is to find an optimal Q value that satisfy the Bellman optimality equation:

$$q_*(S, A) = \mathrm{E}\left[R_{t+1} + \gamma \max_{A'} q_*(S', A')\right]. \tag{2}$$

For any state-action pair (S, A) at time t, the expected return from starting in state S selecting action A and following the optimal policy q_* thereafter is going to be the expected reward we get from taking an action A in state S, which is R_{t+1} plus the maximum expected discounted return that can be achieved from any possible next state-action pair. Also, since the agent is following an optimal policy, the following state S' will be the state from which the best possible next action A' can be taken at time $t + 1$ and the $\max_{A'} q_*(S', A')$ is outputted from the target network. This will be used eventually to calculate the loss from the main Q-network which is calculated by comparing the generated Q values from the main Q-network to the target Q values from the right-hand side of the Bellman equation, where the objective here is to minimize this loss. After the loss is calculated, the parameters θ within the main Q-network are updated via Stochastic Gradient Descent (SGD) and back-propagation. This process is done repeatedly for each state in the environment until minimizing the loss and arriving to an approximate optimal Q value as follows:

$$\begin{aligned} &\text{Loss} = q_* - q, \\ &\text{Loss} = \mathrm{E}\left[R_{t+1} + \gamma \max_{A'} q_*(S', A')\right] - \mathrm{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}\right], \end{aligned} \tag{3}$$

which can be rewritten into the following equation:

$$\begin{aligned} \text{Loss} &= y_j - Q(\mathrm{S}_i, A_i|\theta), \\ y_j &= R_j + \gamma \max_{A'} Q^-(S', A'|\theta^-). \end{aligned} \tag{4}$$

However, DQN has the drawback of overestimation in most cases. Normally, the overestimation is caused by Q value update rule of taking the maximum Q value of the new state. Therefore, a double DQN was proposed by Hado et al. [28] to overcome the overestimation of the DQN. Double DQN improved Q value update rule by selecting the action corresponding to the maximum Q value of the current Q-network rather than using the maximum Q value of the target Q-network.

To make sure that the selected action for the next state is the action with the highest value function (highest Q value), the current Q network is used to find the best action with the highest Q value $(A_{\max})$, then the target network is used to calculate the target Q value (Q^-) of taking this action at the next state:

$$\begin{aligned} \text{Loss} &= y_j - Q(\mathrm{S}_i, A_i|\theta), \\ y_j &= R_j + \gamma Q^-(S', A_{\max}|\theta^-), \end{aligned} \tag{5}$$

where

$$A_{\max} = \arg\max_{A'} Q((S', A'|\theta)). \tag{6}$$

DQN and double DQN are concerned with learning a state-action value (Q value) function and then selecting actions based on this value, where the Q value indirectly evaluates the policy that the agent follows. On the other

hand, policy gradient methods instead learn the policy π directly by a parameterized function $\pi_\theta(A|S)$ with respect to θ, where the objective function value relies on the policy. Thus, the algorithm goal is to optimize θ to determine the optimal value of the function $\pi_\theta(A|S)$.

The REINFORCE [29] (Monte-Carlo policy gradient) is a model-free, online, on-policy reinforcement learning technique. REINFORCE depends on an estimated return by Monte-Carlo methods using episode samples to update the policy parameter θ. The policy gradient methods learn a policy function directly (instead of a Q function). On-policy, means that REINFORCE learns from trajectories generated by the current policy. The objective function for policy gradients is defined as follows:

$$J(\theta) = \mathbb{E}\left[\sum_{t=1}^{T-1} R_{t+1}\right]. \tag{7}$$

A useful way to learn an approximation policy is by directly maximizing the expected reward using a gradient method (i.e., policy gradient). It describes the gradient of the expected reward with respect to the parameters, where the objective function J is calculated to learn a policy that maximizes the cumulative future reward R to be received starting from any given time t until the terminal time T. The policy optimization process uses a gradient ascent with the partial derivative of the objective with respect to the policy parameter θ to maximize the objective function:

$$\theta \leftarrow \theta + \frac{\delta J(\theta)}{\delta \theta}. \tag{8}$$

REINFORCE works because the expectation of the sample gradient is equal to the actual gradient as shown in the consecutive equation:

$$\begin{aligned} \nabla_\theta J(\theta) &= \mathbb{E}_\pi\{Q^\pi(s,a)\ln\ \pi_\theta(A|S)\}, \\ &= \mathbb{E}_\pi[G_t \nabla_\theta \ln\ \pi_\theta(A_t|S_t)]. \end{aligned} \tag{9}$$

Here, one can measure G_t from real sample full trajectories and employ it to update the policy gradient. A commonly used modification of REINFORCE is to subtract a baseline value from the return G_t to decrease the variance of gradient estimation, while keeping the bias unchanged. For example, a common baseline is to subtract state-value from action-value, and if adapted, one could use the advantage $\delta(S,A) = \mathrm{Q}(\mathrm{S},\mathrm{A}) - \mathrm{V}(\mathrm{S})$ in the gradient ascent update. While training the agent for each training episode, the agent generates episode experience by following actor policy $\mu(S)$. The agent conducts actions until it arrives at the terminal state S_T. The episode experience includes the sequence S_1, A_1, R_2, S_2,. . ., S_{T-1}, A_{T-1}, R_T, S_T. Then, the agent calculates the return G_t each time step. In case a baseline was used, then the advantage function δ_t is calculated employing the baseline value function estimated from the critic as given by:

$$\delta(s,a) = G_t - V(S_t|\theta_v). \tag{10}$$

In fact, the REINFOR.

CE-with-baseline technique learns both a policy and a state-value function, but according to Sutton et al. [29], it will not be considered as an actor-critic method because the state-value function is used only as a baseline, not as a critic. This means that the critic will not be used for bootstrapping that illustrates updating the value estimate for a state from the estimated values of subsequent states. However, REINFORCE applies the state-value function only as a baseline for the state whose estimate is being updated. Afterwards, in reinforce with baseline, the agent accumulates the gradients for the actor network and critic network as represented by Wang et al. (11) and He et al. (12):

$$\mathrm{d}\theta_\mu = \sum_{t=1}^{T-1} \delta_t \nabla_{\theta_\mu} \ln\ \mu\left(\mathrm{S}_t|\theta_\mu\right), \tag{11}$$

$$d\theta_v = \sum_{t=1}^{T-1} \delta_t \nabla_{\theta_v} \mathrm{V}(\mathrm{S}_t|\theta_v). \tag{12}$$

Finally, the agent will update the actor parameter θ_μ, and the state-value θ_v in case of a baseline, as shown by He et al. (13) and Oozeer and Haykin (14), respectively, where α and β are the learning rates.

$$\theta_\mu = \theta_\mu + \alpha \mathrm{d}\theta_\mu, \tag{13}$$

$$\theta_v = \theta_v + \beta \mathrm{d}\theta_v. \tag{14}$$

3.3. Agents Features. To train the agents, the topological line states were given as inputs (also called observations). The distribution of the faults for the three topologies PS1, PS2, and PS3 has resulted in 12 faults in L–L–L–G case and 36 faults for L–L–G case. Each fault is placed at each possible line where the current can flow through. Therefore, let $I = \{1, 2, \ldots, 12\}$ and $K = \{1, 2, \ldots, 36\}$. For every time step t, an agent is given an observation $s_t(I) = \{s_t(1), s_t(2), \ldots, s_t(12)\}$ (for L–L – L–G case) or $s_t(K) = \{s_t(1), s_t(2), \ldots, s_t(36)\}$ (for L–L–G case). The initial state of each observation is $s(I \vee K) = 1$ which means that the line is not faulted (in service), the current is available and can flow through the line. However, when a line is faulted (out of service), the line's state is switched to $s(I \vee K) = 0$, which means that the line is faulted, and the current cannot flow through this line as illustrated by:

$$s_t(I \vee K) = \begin{cases} 1, & \text{if line}\,(I \vee K)\text{is in service at time}\,t, \\ 0, & \text{if line}\,(I \vee K)\text{is out of service at time}\,t. \end{cases} \tag{15}$$

Likewise, in every time step t, the agent selects to defect one line out of the I or K possible faults, where $A_t\ (I \vee K) = 1$. Once a fault is selected, the faulted line is disconnected, and the current is rerouted into other possible paths (if exists) toward loads. In addition, the reward function R is defined as follows:

$$R_{t+1}(S_t, A_t) = \begin{cases} -10, & \text{each}\,t\,\text{step}, \\ -10, & \text{if}\ A_t \in We_t, \\ 0, & \text{otherwise}. \end{cases} \tag{16}$$

Each time step the agent selects a line to attack, the agent receives a negative reward. Therefore, the number of faults that is needed to cause an outage of the system equals to the time steps in this episode. Also, during an episode, the actions that are taken by the agent will be stored in a buffer $We_t = (A_1, A_2, \ldots, A_{T-1})$. The current action A_t taken by the agent at time t is compared to We_t to prevent the agent from repeating an action that was taken previously in the episode. By doing so, the agent can be trained with the aim of determining the minimum number of faults required to black out the system.

3.4. Networks Parameters. The DQN and double DQN agents were implemented by first defining the critic networks that get the observations as inputs. A critic network has two hidden layers each with 24 hidden neurons, and each hidden layer is connected with a rectified linear activation function (RELU) and passed to the output layer to find the Q value for each defined action. The optimizer for the critic network is ADAM, with a learning rate of 0.001. The gradient threshold parameter was set up and defined to be 1 to prevent any gradient explosion when the network back propagates to update the network weights. This usually happens when the gradients increase in magnitude exponentially, which results in an unstable training and can diverge within a few iterations. Gradient clipping can prevent gradient explosion by stabilizing the training at higher learning rates and in the presence of outliers. Gradient clipping enables networks to be trained faster and does not often affect the accuracy of the learned task [30].

Adding a regularization (L2 regularization factor) term for the weights to the loss function is one way to reduce overfitting [31]. Another parameter that is needed to train the agent is the experience buffer that is assigned with size of 3000 since the model is relatively small. The agent computes updates using a mini batch of experiences randomly sampled from the buffer with size of 64 which is large enough to reduce the variance when computing gradients, but it increases the computational effort. The discount factor that applies to future rewards during training is 0.9.

The REINFORCE agent is composed of an actor that has two hidden layers with 24 hidden neurons, and each hidden layer is connected with an RELU activation function. Likewise, the REINFORCE with baseline agent, was constructed of an actor and a baseline network. The baseline has two hidden layers with 24 hidden neurons with a RELU function. Similar to DQN agent, the gradient threshold was set to 1. Alongside an ADAM optimizer with a learning rate of 0.005 and a discount factor of 0.9, the learning rates for the two REINFORCE agents were optimized with different values until 0.005 was found to produce better results.

4. Experimental Results

The four agents were implemented using Simulink (Simscape Electrical) environment for the three topologies for the two cases of L–L–L–G and L–L–G fault scenarios, respectively. These agents are DQN, double DQN, REINFORCE, and REINFORCE with baseline. The results for the case of symmetrical L–L–L–G fault scenarios are shown in Figure 4.

The figure shows the training progress of the four agents, where it points out the success rate with the number of episodes. Each episode describes a scenario of lines outages the agent applies to cause a complete system blackout. It can be observed that the DQN agent successfully found a policy that is able to outage the three topologies with a high success rate. It shows also that the DQN agent learned faster than the other agents and was stable during the learning. The double DQN agent was slightly slower at the start of the training but later was stable and achieved a higher success rate than the DQN agent in the three topologies. However, the REINFORCE and the REINFORCE with baseline were slower in learning. The REINFORCE failed in the three topologies to converge and had lots of spikes, which explains that the agent was not stable during the training process. The REINFORCE with baseline succeeded to stabilize in PS3. But in PS1 and PS2, it was improving slowly, which means that by letting the agent train in more episodes, it will converge to an optimal policy. The agent cannot explore the action-state space efficiently. Thus, it takes longer time to find a good policy. It is worth mentioning that all the attempts to optimize the REINFORCE agent by adjusting the learning rate and the number of hidden neurons in the actor network were not sufficient to stabilize the learning procedure and to find an optimal sequence of actions. Table 3 shows the minimum possible number of faults to outage the three systems PS1, PS2, and PS3, respectively, determined by the four agents. It can be shown that the double DQN was able to find a solution or a sequence of actions that results in system outage with a smaller number of faults as compared to the other agents in PS2.

Following Definition 1, the results illustrate that the third topology PS3 is the most resilient topology, as it needed 7 faults to black out the system. This is because PS3 has more redundant paths, so even if a line is faulted, the current can still flow through other paths towards the intended load.

For the second case of single-phase L–L–G fault scenarios, the results are illustrated in Figure 5. The results demonstrate that the double DQN network agent achieved a higher success rate, and it was faster than the other agents. Also, the agent was capable of finding the optimal number of faults for PS1, while the other agents could not find them. The results also illustrate that the REINFORCE agent failed once again to determine the optimal number of faults for the three topologies. Besides that, the agent was not stable, and the success rates were declining in PS1 and PS2, respectively. The REINFORCE with baseline was improving similar to symmetrical fault scenarios but needed longer training episodes to converge. The DQN agent had a similar behaviour to the double DQN agent but could not find the optimal number of faults in PS1.

Table 4 shows the minimum number of faults under single-phase L–L–G fault scenarios. It can be noted that the double DQN found a sequence of faults that was sufficient to outage PS1 with the minimum number of faults as compared to the other agents. The DQN, double DQN, and REINFORCE with baseline agents found the optimal solutions for

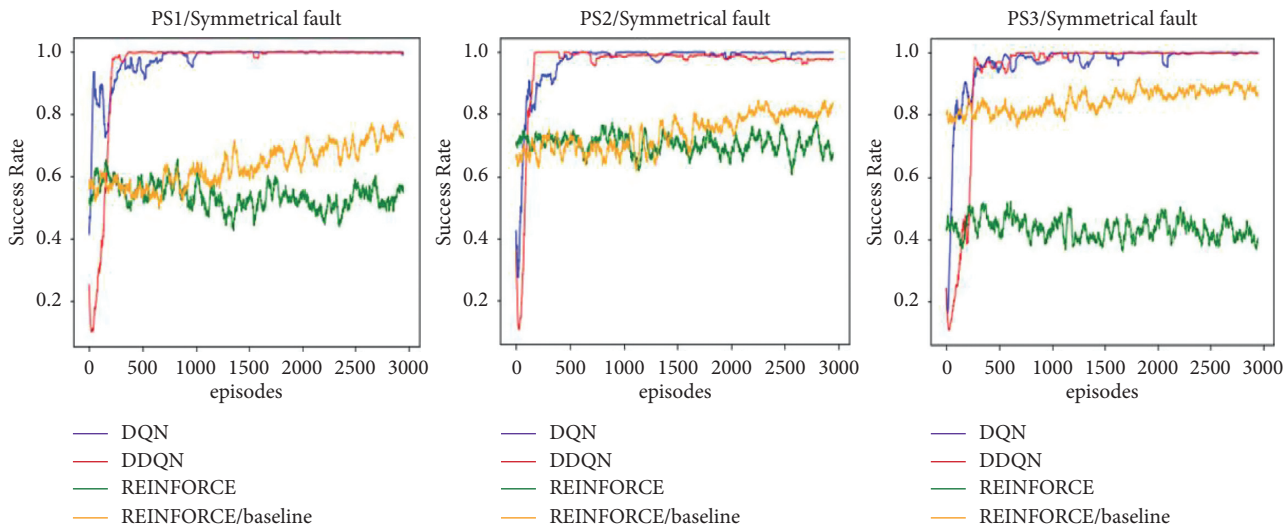


Figure 4: Results of DRL agents for the case of symmetrical L–L–L–G fault scenarios.

Table 3: Minimum number of faults under three-phase L–L–L–G fault scenarios.

PS/agent	DQN	Double DQN	REINFORCE	REINFORCE with baseline
PS1	6	6	6	6
PS2	6	5	6	6
PS3	7	7	7	7

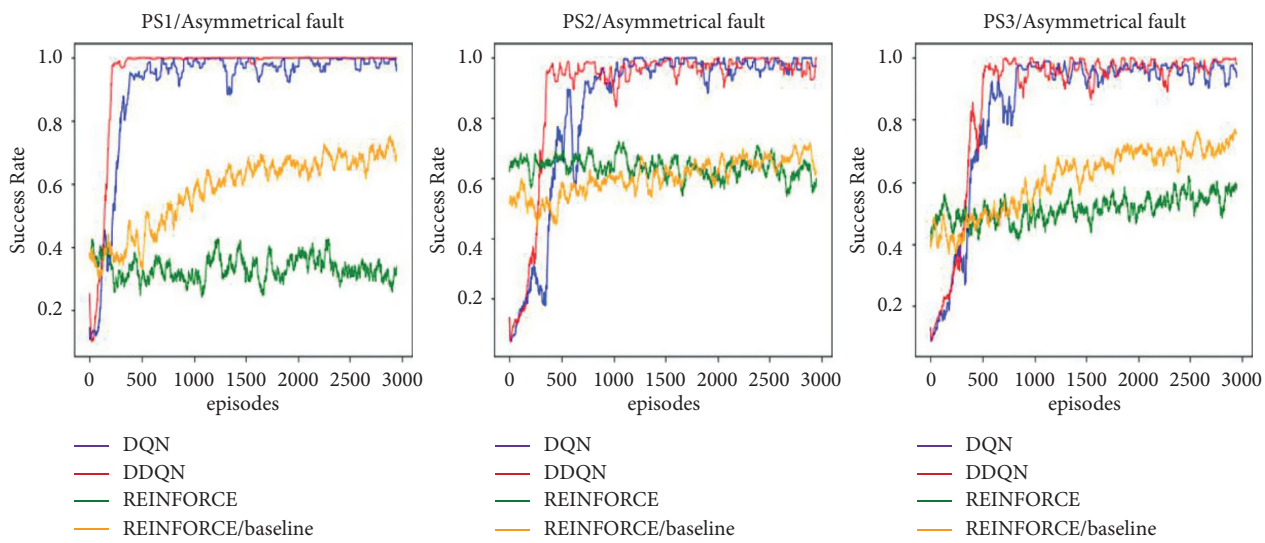


Figure 5: Results of DRL agents for the case of asymmetrical L–L–G fault scenarios.

Table 4: Minimum number of faults under single-phase L–L–G fault scenarios.

PS/agent	DQN	Double DQN	REINFORCE	REINFORCE with baseline
PS1	8	7	10	8
PS2	7	7	8	7
PS3	10	10	11	10

PS2 and PS3, respectively. However, the REINFORCE agent could not find the solution for the three topologies. Following Definition 1, the results show that the third topology PS3 is the most resilient topology.

These results demonstrate that the double DQN agent is a powerful tool for resilience studies that investigate the system's ability to withstand attacks/faults. The double DQN was used to avoid the DQN's overestimation issue, by

improving Q value updating rule when selecting the action corresponding to the maximum Q value of the current Q-network rather than using the maximum Q value of the target Q-network. In addition, the results illustrate for the REINFORCE agent how subtracting a baseline can help reduce the variance and stabilizing the agent. Yet, it needs more training episodes to converge.

5. Conclusion

A new measure for comparing the LoR was proposed for PSs) under attacks/faults. This measure is based on comparing the minimum number of faults that causes system outage by employing reinforcement learning approaches. The reinforcement learning agents were DQN, double DQN, the REINFORCE (Monte-Carlo policy gradient), and REINFORCE with baseline. The LoR of three different PS topologies under symmetrical and asymmetrical fault scenarios were compared. Experimental results showed that while the three PSs have the exact set of generators and have enclosed the same set of loads, yet, they had distinct resiliency levels due to their topological dissimilarity. The multipaths presented in PS3 topology supported the load's demands by the generation side. The results also showed that the double DQN agent was stable and achieved the highest success rate among all agents, as opposed to the REINFORCE agent that failed to determine the minimum number of faults for the three topologies under both symmetrical and asymmetrical faults. In this work, the agents were trained for a certain number of observations (current flow paths and lines availability states) and possible attacks/faults actions for three IEEE 6-bus topologies. However, investigating the LoR for other PSs topologies requires defining and training new agents properties with new observations and actions. As a future work, other factors need to be investigated like recovery time, stability, as well as checking the LoR of more topologies to determine the most resilient PS design. In addition to that further development on the resiliency enhancement can be obtained through the adaptation of DL and decision-making techniques.

Data Availability

The IEEE 6-Bus system load flow Simulink model was used from Mathworks (https://www.mathworks.com/matlabcentral/fileexchange/74690-ieee-6-bus-load-flow-simulink-model). It is provided free for academic research.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to acknowledge Deanship of Graduate Studies and Scientific Research at the German Jordanian University for the Seed fund SATS 03/2020 and Eng. Mohammad Alsheikh for the simulation support.

References

[1] K. Dick, L. Russell, Y. Souley Dosso, F. Kwamena, and J. R. Green, "Deep learning for critical infrastructure resilience," *Journal of Infrastructure Systems*, vol. 25, no. 2, 2019.

[2] A. Bernstein, D. Bienstock, D. Hay, M. Uzunoglu, and G. Zussman, "Power grid vulnerability to geographically correlated failures—analysis and control implications," in *Proceedings of the IEEE conference on computer communications*, pp. 2634–2642, Toronto, ON, Canada, 27 April-2 May 2014.

[3] M. R. Kelly-Gorham, P. D. H. Hines, K. Zhou, and I. Dobson, "Using utility outage statistics to quantify improvements in bulk power system resilience," *Electric Power Systems Research*, vol. 189, p. 106676, 2020.

[4] F. O. Olowononi, D. B. Rawat, and C. Liu, "Resilient Machine Learning for Networked Cyber Physical Systems: A Survey for Machine Learning Security to Securing Machine Learning for Cps," *IEEE Communications Surveys & Tutorials*, vol. 23, pp. 524–552, 2020.

[5] J.-h. Li, "Cyber security meets artificial intelligence: a survey," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 12, pp. 1462–1474, 2018.

[6] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Hoboken, New Jersey, 2002.

[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[8] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Foundations and Trends in Machine Learning*, Now Foundations and Trends, United States, 2018.

[9] S. M. Dibaji, M. Pirani, D. B. Flamholz, A. M. Annaswamy, K. H. Johansson, and A. Chakrabortty, "A systems and control perspective of CPS security," *Annual Reviews in Control*, vol. 47, pp. 394–411, 2019.

[10] J. Yan, H. He, X. Zhong, and Y. Tang, "Q-learning-based Vulnerability Analysis of Smart Grid against Sequential Topology Attacks," *iEEE Transactions on Information Forensics and Security*, vol. 12, pp. 200–210, 2016.

[11] Z. Wang, Y. Chen, F. Liu, Y. Xia, and X. Zhang, "Power System Security under False Data Injection Attacks with Exploitation and Exploration Based on Reinforcement Learning," *IEEE Access*, vol. 6, pp. 48785–48796, 2018.

[12] J. He, C. Chen, S. Zhu, B. Yang, and X. Guan, "Antijamming game framework for secure state estimation in power systems," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 2628–2637, 2018.

[13] J. He, C. Chen, S. Zhu, B. Yang, and X. Guan, "Risk-averse Transmission Path Selection for Secure State Estimation in Power Systems," *IEEE Internet of Things Journal*, vol. 6, pp. 3121–3131, 2018.

[14] M. I. Oozeer and S. Haykin, "Cognitive Risk Control for Mitigating Cyber-Attack in Smart Grid," *IEEE Access*, vol. 7, pp. 125806–125826, 2018.

[15] Y. Chen, S. Huang, F. Liu, Z. Wang, and X. Sun, "Evaluation of reinforcement learning-based false data injection attack to automatic voltage control," *IEEE Transactions on Smart Grid*, vol. 10, pp. 2158–2169, 2018.

[16] F. Wei, Z. Wan, and H. He, "Cyber-attack recovery strategy for smart grid based on deep reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2476–2486, 2019.

[17] D. An, Q. Yang, W. Liu, and Y. Zhang, "Defending against data integrity attacks in smart grid: a deep reinforcement

learning-based approach," *IEEE Access*, vol. 7, pp. 110835–110845, 2019.

[18] Z. Wang, H. He, Z. Wan, and Y. Sun, "Coordinated topology attacks in smart grid using deep reinforcement learning," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 1407–1415, 2020.

[19] Z. Ni, S. Paul, X. Zhong, and Q. Wei, "A reinforcement learning approach for sequential decision-making process of attacks in smart grid," in *Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1–8, Honolulu, HI, USA, 27 Nov.-1 Dec. 2017.

[20] M. N. Kurt, O. Ogundijo and C. Li, Online cyber-attack detection in smart grid: a reinforcement learning approach," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5174–5185, 2018.

[21] P. Negirla, R. Druță, and I. Silea, "Availability improvements through data slicing in PLC smart grid networks," *Sensors*, vol. 20, no. 24, p. 7256, 2020.

[22] C. Kennedy, *IEEE 6 Bus Load Flow Simulink Model*, p. 23, 2020, https://www.mathworks.com/matlabcentral/fileexchange/74690-ieee-6-bus-load-flow-simulink-model.

[23] A. Afwah, S. Mogadisho, E. D. Osman, and A. Abdirahman, *Three-Phase Fault Analysis on Transmission Line in MATLAB SIMULINK*, 2017, https://www.ijraset.com/fileserve.php?FID=36027.

[24] S. Paiva and L. F. Coelho, *Redes de Energia Eléctrica: uma análise sistémica*, Livros Recomendados, Cambridge, 2005.

[25] L. Hewitson, M. Brown, and R. Balakrishnan, *Practical Power System protection*, Elsevier, Amsterdam, Netherlands, 2004.

[26] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[27] R. Sutton and A. Barto, *Introduction to Reinforcement Learning*, MIT press, Cambridge, 2017.

[28] V. H. Hado, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Phoenix, Arizona USA, 2016.

[29] R. Sutton, M. A. David, S. P. Satinder, and M. Yishay, "Policy gradient methods for reinforcement learning with function approximation," *News in Physiological Sciences*, vol. 99, pp. 1057–1063, 1999.

[30] R. Pascanu, T. Mikolov, and Y. Bengio, "On the Difficulty of Training Recurrent Neural Networks," in *Proceedings of the International conference on machine learning*, pp. 1310–1318, Atlanta GA USA, June 2013.

[31] K. B. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT press, Cambridge, 2012.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 7402085, 9 pages
https://doi.org/10.1155/2022/7402085

Research Article

A Semisupervised Majority Weighted Vote Antiphishing Attacks IDS for the Education Industry

Xiaona Yin and Xingxing Zheng

Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Xiaona Yin; yinxiaona1985@126.com

Received 28 February 2022; Revised 7 March 2022; Accepted 8 March 2022; Published 31 March 2022

Academic Editor: Konstantinos Demertzis

Although the digital transformation is advancing, a significant portion of the population in all countries of the world is not familiar with the technological means that allow malicious users to deceive them and gain great financial benefits using phishing techniques. Phishing is an act of deception of Internet users. The perpetrator pretends to be a credible entity, abusing the lack of protection provided by electronic tools and the ignorance of the victim (user) to illegally obtain personal information, such as bank account codes and sensitive private data. One of the most common targets for digital phishing attacks is the education sector, as distance learning became necessary for billions of students worldwide during the pandemic. Many educational institutions were forced to transition to the digital environment with minimal or no preparation. This paper presents a semisupervised majority-weighted vote system for detecting phishing attacks in a unique case study for the education sector. A realistic majority weighted vote scheme is used to optimize learning ability in selecting the most appropriate classifier, which proves to be exceptionally reliable in complex decision-making environments. In particular, the voting naive Bayes positive algorithm is presented, which offers an innovative approach to the probabilistic part-supervised learning process, which accurately predicts the class of test snapshots using prerated training snapshots only from the positive class examples.

1. Introduction

The consequent increase in the popularity of online educational resources, combined with the lack of preparedness, has made the education sector an ideal target for digital phishing attacks [1]. Phishing is the most widespread technique where malicious users create fake websites that look like the official websites of legal organizations/companies/banks [2, 3]. They then send emails or SMS or create misleading messages that link to the misleading URL they have made. Users are asked to fill in confidential personal and financial data on these websites, including usernames, passwords, and bank card details. The main reasons cited by most phishing messages are a problem in the user's account, a confirmation of execution or cancellation of a transaction (which has never been done by the user), a service upgrade action, and so on [4].

A successful phishing attack is based on the victim's lack of knowledge, attention, and visual deception [3]. The average person knows how to handle the essential functions of the computer and the Internet without knowing the process by which it works. So, it cannot recognize traces of phishing, such as a varied e-mail address or a different URL. At the same time, due to ignorance of the risk, the user neglects antiphishing programs. Even in cases where the users have the appropriate knowledge to detect malicious elements, they often will not notice the signs, as they may be abstract or busy with something else. Thus, the user may not pay enough attention to the current security warnings or lack them. After all, the proper phishing technique hides most signs as a successful phishing attack is based mainly on visual deception. The aim is to convince the victim of the authenticity and reliability of the fraud, which is achieved by [5, 6] the following.

(1) *Misleading Text.* This text, which is usually misleading links, may use incorrect syntax or spelling, for example, www.fasebook.com, anagrams, e.g., and www.yutoube.com, or replace similar letters such as the English lowercase *l* (*L*) with the capital I (i).

(2) *Misleading Images.* These images may be visually the same as the images used by a website, for example, the Google logo, but when you click on them, they redirect you elsewhere. An equally standard method is images that mimic the computer operating system.

(3) *Misleading Design.* With the help of misleading text and images and the processing of the code of the original website, the malicious user can create an entire website with the same design as the original.

(4) *Threatening Message.* The message usually contains a threat or a problem that the user must deal with. For example, "if you do not follow the link, your account will be locked," or "as soon as a transaction was made from your account, click here to cancel it."

If a phishing campaign manages to combine all the above, it will be successful in most cases. The research community intensively deals with this cyber threat, while many of their research results have been presented in the international literature [6–10].

Section 2 includes an overview of approaches identified in the literature and associated with similar technical standardization. You will discover more about the suggested system's technique in Section 3. According to the dataset and findings presented in Section 4 of the proposed approach, there are no restrictions on applying it. Section 5 finishes with a summary of the findings and a list of possible next research directions.

2. Literature Review

The concept of phishing attack detection has been approached with various methods from the research community. During the last five years, especially, researchers have been evaluating machine learning approaches to face this rising problem better.

Cuzzocrea et al. [4] offered a machine learning-based approach for detecting the difference among phishing and authentic websites. They built signs to identify phishing activity using cutting-edge machine learning techniques. The suggested solution is based on a simple feature vector to collect and does not need extra processing. They stated that by evaluating a certain algorithm, they might get encouraging results in identifying phishing attempts.

Natural language processing methods were utilized by Peng et al. [11] to evaluate text (but not message metadata) and identify incorrect utterances indicative of phishing attempts. To identify harmful information, they used a semantic analysis of the text. Their strategy resulted in entirely text-based phishing emails, with no harmful attachments attached. They tested it with a huge batch of phishing emails and found that it had a high recall rate, proving that semantic information is a good predictor of social engineering.

Garces et al. [6] conducted a study on examining anomalous behavior associated with phishing online assaults and how machine learning methods may be used to combat the issue. This assessment was done using infected data sets and scripting language tools to establish machine learning for detecting phishing attacks throughout the analysis of URLs to determine if they were good or bad URLs based on specific characteristics of the URLs and to provide real-time information and making informed decisions that reduce the potential damage.

Basit et al. [2] conducted a study of Artificial Intelligence approaches in use, including spoofing attack mitigations tactics, data mining and heuristics, machine learning, and AI techniques. They also evaluated several research for each AI technology that detected phishing attacks and looked at the benefits and drawbacks of each methodology. Compared with other classification techniques such as random forest, support vector machine, decision tree, principal component analysis, and k-nearest neighbor, Machine learning processes provide the most significant results. Future study towards a more configurable strategy, including creative plugin solutions to tag or label whether a website is genuine or leading to a phishing attempt, is suggested.

Saha et al. [5] established a data-driven approach utilizing a feed-forward neural network to anticipate phishing websites. Their program was able to classify websites into three categories: phishing, suspicious, and authentic. The dataset was large, including data from hundreds of web pages, and their model had excellent training and test accuracy percentages. The difference between training and test accuracy was small, indicating that the proposed model learned from the dataset and was capable of quickly detecting unfamiliar web pages. The authentic website identification accuracy, on the other hand, was greater than the existing phishing detection method.

Using machine learning methods such as random forest and decision tree, Alam et al. [7] created a model to identify phishing assaults. To detect phishing, the study used a variety of tactics. The machine learning algorithms were fed standard datasets of phishing assaults from kaggle.com. The suggested model uses feature selection methods like principal component analysis to identify and categorize the datasets' components to study their properties. To categorize the website, a decision tree was employed, and random forest was used for categorization. Finally, a confusion matrix was created to compare the two algorithms' efficiency. The random forest algorithm has a 97 percent accuracy rate. The study team intends to use a convolution neural network to anticipate phishing attempts from a recorded dataset of attacks, which might be included as a tool for intrusion detection systems.

Finally, Singh et al. [12] conducted a survey where they compared 16 distinct study studies. Network-level security, authentication, client-side tools, server-side filters, and user education were the three classes they used to categorize phishing defenses. They came to the conclusion that the research community is still unable to give a "silver bullet" for spoofing attack defense.

As many schools and universities conduct classes online, these organizations must take steps to secure their digital learning environments [13, 14]. The proposed approach of the work aims to detect malicious URLs related to phishing attacks, to predict vulnerabilities, which may come from fraud or cyber-attacks.

3. Proposed Methodology

The primary idea of the proposed methodology is based on an algorithmic approach of the naive Bayes positive classifier [15]. This offers a simple probabilistic approach to part-supervised learning problems. Our goal is to accurately predict the class instance of instantaneous instruction only from the positive class and several unsorted examples. The probabilities that we have to calculate, using only the positive and unclassified examples that we have at our disposal, are the ex-ante probabilities of observing positive and negative examples $p(C = \text{pos})$ and $p(C = \text{neg})$, respectively, as well as the ex-ante probabilities of occurrence of each attribute, for each class (i.e., $p(X_\iota = x_i|C = pos)$ and $p(X_\iota = x_i|C = \text{neg})$. Due to the absence of negative examples, it is impossible to define the $p(C = \text{pos})$, so the user must give an approximation. Let $\widehat{p}(pos)$, so that $p(C = \text{neg})$ is calculated as follows [16]:

$$p(C = \text{neg}) = 1 - \widehat{p}(\text{pos}). \tag{1}$$

In terms of the probabilities of the features given a positive class, $p(X_\iota = x_i|C = \text{pos})$it is estimated strictly for the different types of components [17, 18]:

$$p(X_i = x_i|C = c) = g\left(x_i; \mu_{i,c}, \sigma_{i,c}\right), \tag{2}$$

while for the estimation of $p(X_\iota = x_i|C = \text{neg})$, we use the law of total probability [16, 19]:

$$\begin{gathered} p(X_i = x_i) = p(X_i = x_i|C = \text{pos})p(C = \text{pos}) + p(X_i = x_i|C = \text{neg})p(C = \text{neg}) \Rightarrow \\ p(X_i = x_i|C = \text{neg}) = \frac{p(X_i = x_i) - p(X_i = x_i|C = \text{pos})p(C = \text{pos})}{1 - p(C = \text{pos})}, \end{gathered} \tag{3}$$

where everything is known except the ex-ante probability of occurrence of the characteristic X_ι, $p(X_\iota = x_i)$, which is approximated by assuming that the set UD of the unsorted examples follows the distribution of real-world examples.

The $p(X_\iota = x_i|C = \text{neg})$ approach runs the risk of being negative. Therefore, we need to replace the negative values with 0 and normalize our practices, so that they all have a sum of 1. This is a simple case for the discrete attributes since the domain definition of the attribute takes discrete values, making it possible to calculate them all to normalize them. But, for continuous features, we create a new distribution (normal distribution or sum of Gaussian nuclei). Under the previously mentioned conditions (assumptions), the proposed algorithm that we use in this work is as follows [15, 20–22].

Let us assume a data training body with only positive PD examples and a body of unclassified UD data. Also, let $\widehat{p}(pos)$ estimate the ex-ante probability of the positive class. The naive Bayes positive classifier classifies an unknown x instance as a member of the class [15, 19]:

$$\underset{ce\{\text{pos,neg}\}}{\text{argmax}}\left\{p(C = c|\mathbf{X} = \mathbf{x})\right\} = \underset{c\in\{\text{pas,neg}\}}{\text{argmax}}\left\{p(C = c)\prod p(X_i = x_i|C = c)\right\}. \tag{4}$$

The estimates of the ex-ante probabilities of the classes are calculated from

$$p(C = \text{pos}) = \widehat{p}(\text{pos}), p(C = \text{neg}) = 1 - \widehat{p}(\text{pos}). \tag{5}$$

The estimates of the likelihood of the features are calculated for the discrete elements:

$$\begin{gathered} p(X_i = x_i|C = \text{pos}) = \frac{\#(x_i, \text{PD})}{|P\ D|} \\ p(X_i = x_i) = \frac{\#(x_i, \text{UD})}{|U\ D|}. \end{gathered} \tag{6}$$

For continuous features using Gaussian distribution [23, 24],

$$p(X_i = x|_iC = \text{pos}) = g\left(x_i; \mu_{i,c}, \sigma_{i,c}\right)p(X_i = x_i) = g(x_i; \mu_i, \sigma_i). \tag{7}$$

For continuous features using Gaussian kernels,

$$\begin{gathered} p(X_i = x_i|C = \text{pos}) = \frac{1}{|\text{PD}|}\sum_j g\left(x_i; (x_i)_j, \frac{1}{\sqrt{|P\ D|}}\right), \\ p(X_i = x_i) = \frac{1}{|\text{UD}|}\sum_J g\left(x_i; (x_i)_j, \frac{1}{\sqrt{|U\ D|}}\right). \end{gathered} \tag{8}$$

For all the previously mentioned cases, the following applies:

$$p(X_i = x_i|C = \text{neg}) = \frac{p(X_i = x_i) - p(X_i = x_i|C = \text{pos})p(C = \text{pos})}{1 - p(C = pos)}, \tag{9}$$

which is normalized so that

$$p(X_i = x_i|C = \text{neg}) = \max\{p(X_i = x_i|C = \text{neg}); 0\} \text{ and } \sum_{\forall x} p(X_i = x|C = \text{neg}) = 1, \tag{10}$$

where x takes values from the definition field of X_i.

Given that PD is the set of positively sorted examples and UD is the set of nonsorted, a first not satisfying approach is to assume that all unknown models are negative, so

$$\widehat{p}(\text{pos}) = \frac{|\text{PD}|}{|\text{PD}| + |\text{UD}|}. \tag{11}$$

But since there will also be positive examples in the unclassified UDs, a better approach to $\widehat{p}(pos)$ would be to add the number of these positive examples to the numerator of the above fraction. We construct the first classifier to classify the unknown samples using the simple hypothesis that all unknowns are negative. The number of positive examples to be found is added to the numerator of the above fraction, a new approximation of $\widehat{p}(\text{pos})$ is calculated, and a new classifier is constructed to reclassify the unknown examples [15, 19]:

$$\widehat{p}(\text{pos}) = \frac{|\text{PD}| + |\text{most\_probable\_positive\_from}(\text{UD})|}{|\text{PD}| + |\text{UD}|}. \tag{12}$$

This process is repeated until $\widehat{p}(pos)$ converges, remaining the same in two consecutive steps. However, because not every single classifier can be optimal for all metrics, we will use a voting scheme, that is, a combination of classifiers, to derive the optimal characteristics for all performance metrics as a decision rule based on the predicted class with the most votes.

Specifically, because we have at least two independent, equivalent classifiers which make a single decision on the class of the unlabeled sample, this sample is classified in the class where there is an absolute majority, that is, a decision agreed by at least half of the experts. To make the system more realistic, the decision of each classifier is multiplied by a weight that reflects the individual confidence in its conclusions. The more reliable the classifier is in its choices, the higher the weight value assigned to it. The sum of the weights is equal to one. Therefore, if the decision of the k classifier to classify the unknown sample in the i class is given by d_{ik} with $0 \le i \le m$, where m is the number of classes, then the final combined decision for assignment to class I is as follows [25, 26]:

$$d_i^{\text{com}} = \sum_{i=1,2,\ldots,m} \omega_k * d_{ik}. \tag{13}$$

Therefore, the class y is the one selected if d_y^{com} is the maximum. To find the optimal values of the weights, they must minimize the error function defined as

$$y \neq \text{true\_label for max}\left(d_y^{\text{com}}\right). \tag{14}$$

A decision function is optimal when the previously mentioned formula is minimized in all possible decisions. Assuming independence between classifiers and that if the probability of selecting class i is $p_{i,}$ then the likelihood of choosing any other class is evenly distributed among them, we arrive at a majority weighted vote approach [17, 19, 20].

$$f^{\text{opt}}(x) = \text{sign}\left(\sum_{i=1}^{n} \omega_i * x_i\right). \tag{15}$$

The weights ω_i are given by the relation:

$$\omega_i = \log\left(\frac{p_i}{1 - p_i}\right), \quad i \in [n], \tag{16}$$

where p_i is the probability that the specialist will choose class i.

The calculation of the weights by approaching the joint probability distribution for each class with a set of answers of the classifiers is as follows:

$$\text{P}(c|f_1, \ldots, f_v) = \frac{p(c) * P(f_1, \ldots, f_v|c)}{P(f_1, \ldots, f_v)}, \tag{17}$$

where f_1 is the attribute, and c is the variable for the class. Assuming independence between the features we have from the previous formula

$$P(c|f_1, \ldots, f_v) = \frac{1}{Z} p(c) * \prod_{i=1}^{v} p(f_i|c). \tag{18}$$

We observe that Z is a multiplication factor and is independent of the variable class c. Taking as random variables all the answers of the classifiers instead of the characteristics, we end up with the following:

$$P(c|e_1,\ldots,e_k) = \frac{1}{Z}p(c) * \prod_{i=1}^{k} p(e_i|c). \quad (19)$$

Given the relation,

$$P(c,e_1,\ldots,e_k) = P(c|e_1,\ldots,e_k) * Z, \quad (20)$$

that is, replacing the bound probability with the common ones, we conclude from the previous formula [19, 24]:

$$P(c,e_1,\ldots,e_k) = p(c) * \prod_{i=1}^{k} p(e_i|c). \quad (21)$$

Therefore, the weights are related to the variable of class u with the relation:

$$\omega(e_1,\ldots,e_k) = p(c=u) * \prod_{i=1}^{k} p(e_i|c=u). \quad (22)$$

Thus, the class $\hat{c}$ of the unlabeled sample x is calculated as

$$\hat{c} = \max_{u\in C} \sum_{i} \omega_u * r_{i,u}. \quad (23)$$

Therefore, given each input sample x and set of answers of the classifiers, the weights are calculated, and the final decision is made based on the equation of $\hat{c}$.

A depiction of the proposed methodology is presented in Figure 1.

4. Dataset and Results

In the present study, we used data from the PhishTank database, a complete database for registrations for Phishing URLs. A total of 860,000 URLs were used, of which 500,000 were legit, and 360,000 were phishing. The export of features was based on the idea that URLs are divided into subsections as explicitly shown in domain, directory, file, and parameters. In each section, we measure the number of some special characters (e.g., -, #, @, etc.) and the size of the section and check if certain words appear in specific sections (e.g., "client," "server," "script," etc.) and if there is an IP or e-mail in the domain section, as well as the number of vowels in the domain. In addition, there are features based on external services (WHOIS2, HTTPS3 Protocol, SSL4 certificate, etc.) and components based on the number of occurrences of specific HTTP headers (e.g., cookies; strict-transport-security). The following features were extracted in detail from each URL:

(1) check_ssl: check for valid SSL protocol (0 False - 1 True)
(2) url_redirect: Number of redirects (numeric value)
(3) url_shortened: URL shortcut control (0 False - 1 True)
(4) favicon: check if the favicon is loaded from an external domain (0 False - 1 True)
(5) dns_record: check for DNS domain registration in WHOIS (0 True - 1 False)
(6) iFrame: iFrame existence check (0 False - 1 True)
(7) rightClick: check if right-click is disabled (0 True - 1 False)
(8) onmouseover: check if onmouseover changes the status bar (0 True - 1 False)
(9) check_URL_anchor: check if anchors lead to a new domain (real percentage)
(10) sfh: check if the action of a form tag triggers an action (0 False - 1 True)
(11) double_slash: Existence "//" more than 1 time in the URL (0 False - 1 True)
(12) url_dot_url: Number of "." in full URL (numeric value)
(13) url_hyphen_url: Number of "-" in the whole URL (numeric value)
(14) url_questionmark_url: Number of "?" in full URL (numeric value)
(15) url_at_url: Number of "@" in the whole URL (numeric value)
(16) url_hashtag_url: Number of "#" in the whole URL (numeric value)
(17) url_dollar_url: Number of "$" in the whole URL (numeric value)
(18) url_percent_url: Number of "%" in the whole URL (numeric value)
(19) tld_length: Number of TLD5 (numeric value)
(20) tld_count: Number of sub-TLDs (numeric value)
(21) url_length: Number of characters in the entire URL (numeric value)
(22) e-mail_in_url: Show e-mail inside URL (0 False - 1 True)
(23) word_script_in_url: Display the word "script" inside the URL (0 False - 1 True)
(24) check_https_in_url: Display the word "https" inside the URL (0 False - 1 True)
(25) url_dot_domain: Number of "." in the Domain section (numeric value)
(26) url_hyphen_domain: Number of "-" in the Domain section (numeric value)
(27) count_vowels: Number of vowels in the Domain section (numeric value)
(28) domain_length: Number of characters in the Domain section (numeric value)
(29) ip_in_domain: Display IP in the Domain section (0 False - 1 True)
(30) client_or_server_domain: Display client or server in Domain (0 False - 1 True)
(31) check_age_of_domain: WHOIS Domain Registration Days (numeric value)
(32) days_till_expiration_domain: Days until SSL expires (numeric value)
(33) url_dot_directory: Number of "." in the Directory section (numeric value)

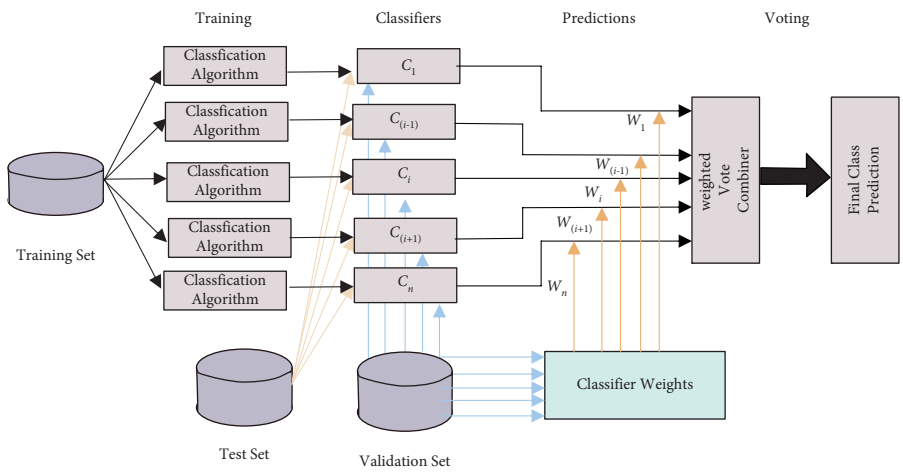


Figure 1: The majority weighted vote methodology.

(34) url_hyphen_directory: Number of "-" in the Directory section (numeric value)

(35) url_at_directory: Number of "@" in the Directory section (numeric value)

(36) url_slash_directory: Number of "/" in the Directory section (numeric value)

(37) url_percent_directory: Number of "%" in the Directory section (numeric value)

(38) directory_length: Length of in the Directory section (numeric value)

(39) url_dot_File: Number of "." in the File section (numeric value)

(40) url_hyphen_File: Number of "-" in File section (numeric value)

(41) url_at_File: Number of "@" in the File section (numeric value)

(42) url_percent_File: Number of "%" in File section (numeric value)

(43) file_length: Number of characters in the File section (numeric value)

(44) url_dot_params: Number of "." in the Params section (numeric value)

(45) url_hyphen_params: Number of "-" in the Params section (numeric value)

(46) url_at_params: Number of "@" in the Params section (numeric value)

(47) url_underline_params: Number of "_" in Params section (numeric value)

(48) url_hashtag_params: Number of "#" in the Params section (numeric value)

(49) url_dollar_params: Number of "$" in the Params section (numeric value)

(50) url_percent_params: Number of "%" in the Params section (numeric value)

(51) params_length: Number of characters in the Params section (numeric value)

(52) tld_params: check if there are any of the TLDs in Params (0 False - 1 True)

(53) count_params: Number of parameters to get a value (numeric value)

(54) cookie: check if the HTTP header adds a cookie (0 False - 1 True)

(55) strict_trans_sec: check for HTTP header to switch to HTTPS (0 False - 1 True)

(56) a_tags_count: Number of tags in the HTML code of the web page (numeric value)

(57) form_tags_count: Number of form tags in HTML code (numeric value)

(58) e-mail_tags_count: Number of "emails" displayed in HTML code (numeric value)

(59) pass_tags_count: Number of "password" occurrences in HTML code (numeric)

(60) hidden_tags_count: Number of hidden tags in HTML code (numeric value)

(61) actions_tags_count: Number of action tags in HTML code (numeric value)

(62) signin_tags_count: Number of "sign in" occurrences in HTML code (numeric)

(63) signup_tags_count: Number of "sign up" occurrences in HTML code (numeric)

(64) label: for the type of URL (0 legitimate - 1 phishing)

To prove the possibility of the proposed scheme, we made a comparison with known machine learning methods. The results of the process are presented in Table 1.

Although all the models achieve high success rates, the proposed one achieved the highest success rates. With the voting naive Bayes positive technique [15, 19] that we propose, we perform the highest percentages for accuracy, precision, recall, and F1, which indicates the possibility of generalization of the proposed system. Also, the metric MCC, which is used as a measure of the quality of the categorization, and the high results of the proposed method prove that the coefficient considers the TP, FP, TN, and FN,

Table 1: Performance measures.

Model	Accuracy	Auc	Recall	Prec.	F1	Kappa	MCC	TT (sec)
Voting naive bayes positive	0.9314	0.9982	0.9292	0.9320	0.9312	0.8722	0.8871	2.339
Light gradient boosting machine	0.8949	0.9777	0.8770	0.8970	0.8941	0.8197	0.8218	0.244
Extreme gradient boosting	0.8942	0.9759	0.8745	0.8976	0.8935	0.8187	0.8211	15.896
CatBoost classifier	0.8926	0.9763	0.8710	0.8950	0.8921	0.8154	0.8172	4.328
Random forest classifier	0.8918	0.9739	0.8685	0.8961	0.8918	0.8145	0.8169	0.562
Gradient boosting classifier	0.8864	0.9747	0.8635	0.8914	0.8861	0.8053	0.8082	0.665
SVM - radial kernel	0.8726	0.9498	0.8388	0.8765	0.8716	0.7806	0.7832	0.387
k-Neighbors classifier	0.8687	0.9494	0.8336	0.8700	0.8666	0.7727	0.7753	0.128
MLP classifier	0.7988	0.8728	0.8076	0.7877	0.7541	0.7719	0.7056	6.322

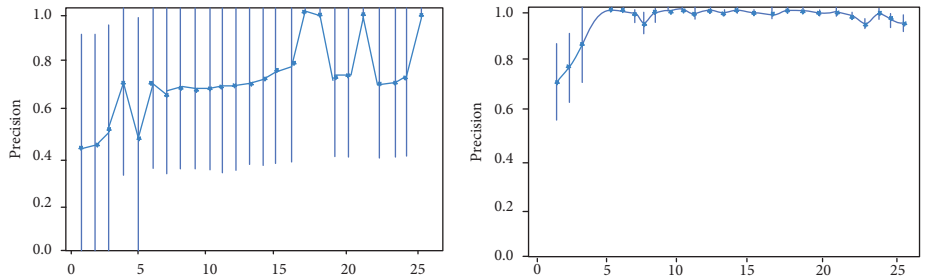


Figure 2: Precision majority vote (left) vs. precision weighted vote (right).

which ensures a very balanced performance in cases where the two classes have different sizes, as in the problem that concerns us. The MCC is essentially a correlation coefficient between the predicted and observed values of the categorization, and it takes values between -1 and +1. A factor of +1 represents a perfect prediction. If its value is 0, the categorizer prediction is no better than a random prediction. When its value is -1, there is a total difference between the forecast price and the real one. While there is no perfect way to describe the results of a single numbered confusion matrix, the metric MCC is considered one of the best. The methodology in question also strengthened the weighting process in the majority weighted vote process and how the model weightings were calculated [27, 28].

Also, the majority weighted vote process leads to better performance of the final model because it reduces model variability without significantly increasing bias. This means that while the predictions of an individual model are pretty sensitive to the noise of the training set, the weighted average of the results of many classifiers is not if they are not correlated with each other. This happens here due to the method followed since different classifiers see different points of the education set. A typical example of proof of this fact is in Figure 2, which clearly shows the performance of the classifiers with the two different procedures and the apparent superiority of the proposed majority weighted vote.

In general, with the majority weighted vote procedure followed, even if the relative majority agrees with the prevalence of a class, the uncertainty about their prediction against the firm opinion of the two models would lead to a wrong result by a majority vote. On the other hand, although theoretically ensuring significant percentages in the evaluation metrics and showing commendably good results, a simple voting process does not consider the general cases of class inhomogeneity, so the forecasts do not guarantee a final result based on generalization.

In conclusion, the operation and the results of the application are considered very satisfactory, which should also be noted that it manages to detect phishing websites from the first minute they are published, in contrast to the browsers and databases of cybersecurity companies, which require some time-space, maybe a lot of reports from users.

5. Conclusions

The consequent increase in the popularity of online educational resources, combined with the lack of preparedness, has made the education sector an ideal target for digital phishing attacks. The identification and timely assessment of these threats to the functioning of educational organizations allow the detection of incidents and the corresponding identification of correlations and causal relationships with security incidents, which can significantly mitigate the effects of organized cyber attacks. In this spirit, a semi-supervised majority-weighted voting system for detecting phishing attacks was proposed in this paper. Specifically, the voting naive Bayes positive algorithm was used, which offers an innovative approach to the probabilistic learning process with partial supervision. Our goal is to accurately predict the class-class of test snapshots using both classified and positive training snapshots, as well as a variety of unclassified examples.

This algorithmic process, which we presented for the first time in the literature, was evaluated in a very complex problem of identifying URLs related to phishing attacks in a timely scenario associated with the educational process. A

very complex but ideal dataset was used, which computes the problem of phishing attacks in the educational sector in a complete way, and the proposed algorithm achieved very high generalization rates.

Future research for the extension of the proposed system is related to implementing the system with more classes to reveal in more detail the system's ability to model more complex problems. It would also be essential to identify ways the system can receive information from a posteriori or a priori probabilities in a complete predictive environment with retrospective relationships. For example, the method by Bayesian inference will be enhanced, which is a method of statistical inference, where Bayes' theorem is used to update the probability for a hypothesis as more evidence or information becomes available.

Data Availability

Data are available on reasonable request.

Conflicts of Interest

The authors declare that are no conflicts of interest.

References

[1] W. Holmes and S. Anastopoulou, "What do students at distance universities think about AI?" in *Proceedings of the Sixth (2019) ACM Conference on Learning @ Scale*, pp. 1–4, Chicago IL USA, June 2019.

[2] A. Basit, M. Zafar, X. Liu, A. R. Javed, Z. Jalil, and K. Kifayat, "A comprehensive survey of AI-enabled phishing attacks detection techniques," *Telecommunication Systems*, vol. 76, no. 1, pp. 139–154, 2021.

[3] D. K. McGrath and M. Gupta, "Behind Phishing: An Examination of Phisher Modi Operandi," 2021, https://www.usenix.org/legacy/event/leet08/tech/full_papers/mcgrath/mcgrath_html/index.html.

[4] A. Cuzzocrea, F. Martinelli, and F. Mercaldo, "Applying machine learning techniques to detect and analyze web phishing attacks," in *Proceedings of the Twentyth International Conference on Information Integration and Web-based Applications & Services*, pp. 355–359, Yogyakarta, Indonesia, November 2018.

[5] I. Saha, D. Sarma, R. J. Chakma, M. N. Alam, A. Sultana, and S. Hossain, "Phishing attacks detection using deep learning approach," in *Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 1180–1185, Tirunelveli, India, August 2020.

[6] I. Ortiz Garces, M. F. Cazares, and R. O. Andrade, "Detection of phishing attacks with machine learning techniques in cognitive security architecture," in *Proceedings of the 2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 366–370, Las Vegas, NV, USA, December 2019.

[7] M. N. Alam, D. Sarma, F. F. Lima, I. Saha, R.-E. Ulfath, and S. Hossain, "Phishing attacks detection using machine learning approach," in *Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 1173–1179, Tirunelveli, India, August 2020.

[8] M. I. A. Ajlouni, W. Hadi, and J. Alwedyan, "Detecting phishing websites using associative classification," *Journal of Information Engineering and Applications*, vol. 3, no. 7, pp. 6–10, 2013.

[9] A. Basit, M. Zafar, A. R. Javed, and Z. Jalil, "A novel ensemble machine learning method to detect phishing attack," in *Proceedings of the 2020 IEEE Twenty Third International Multitopic Conference (INMIC)*, pp. 1–5, Bahawalpur, Pakistan, November 2020.

[10] K. Demertzis and L. Iliadis, "Cognitive web application firewall to critical infrastructures protection from phishing attacks," *Scienpress Ltd*, vol. 9, no. 2, p. 26, 2019.

[11] T. Peng, I. Harris, and Y. Sawa, "Detecting phishing attacks using natural language processing and machine learning," in *Proceedings of the 2018 IEEE Twelveth International Conference on Semantic Computing*, pp. 300-301, ICSC), Laguna Hills, CA, USA, January 2018.

[12] C. Singh and Meenu, "Phishing website detection based on machine learning: a survey," in *Proceedings of the 2020 Sixth International Conference on Advanced Computing and Communication Systems (ICACCS)*, pp. 398–404, Coimbatore, India, March 2020.

[13] B. Chang, "Student privacy issues in online learning environments," *Distance Education*, vol. 42, no. 1, pp. 55–69, 2021.

[14] A. A. Drozdova and A. I. Guseva, "Modern technologies of E-learning and its evaluation of efficiency," *Procedia - Social and Behavioral Sciences*, vol. 237, pp. 1032–1038, 2017.

[15] T. Dong, W. Shang, and H. Zhu, "Naive bayesian classifier based on the improved feature weighting algorithm," in *Advanced Research on Computer Science and Information Engineering*, pp. 142–147, Springer, Berlin, Germany, 2011.

[16] A. J. M. Garrett, "Review: probability theory: the logic of science, by E. T. Jaynes," *Law, Probability and Risk*, vol. 3, no. 3-4, pp. 243–246, 2004.

[17] L. E. B. Salasar, J. G. Leite, and F. Louzada, "Likelihood-based inference for population size in a capture-recapture experiment with varying probabilities from occasion to occasion," *Brazilian Journal of Probability and Statistics*, vol. 30, no. 1, pp. 47–69, 2016.

[18] D. Semenova and N. Lukyanova, "Random set decomposition of discrete-continuous random variables," in *Proceedings of the 2012 IV International Conference "Problems of Cybernetics and Informatics" (PCI)*, pp. 1–4, Baku, Azerbaijan, September 2012.

[19] J. O. Berger, "Bayesian analysis," in *Springer Series in Statistics," in Statistical Decision Theory and Bayesian Analysis*, J. O. Berger, Ed., Springer, New York, NY, USA, pp. 118–307, 1985.

[20] J. O. Berger, "Basic concepts," in *Springer Series in Statistics," in Statistical Decision Theory and Bayesian Analysis*, J. O. Berger, Ed., Springer, New York, NY, USA, pp. 1–45, 1985.

[21] J. L. Myers, A. Well, and R. F. Lorch, *Research Design and Statistical Analysis*, Routledge, Oxfordshire, UK, 2010.

[22] A. E. Barinov and A. A. Zakharov, "Clustering using a random walk on graph for head pose estimation," in *Proceedings of the 2015 International Conference on Mechanical Engineering, Automation and Control Systems (MEACS)*, pp. 1–5, Tomsk, Russia, December 2015.

[23] W. Wu, "The discrete Gaussian expectation maximization (gradient) algorithm for differential privacy," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 7962489, 13 pages, 2021.

[24] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2014, https://arxiv.org/abs/1312.6114.

[25] P. Lin, "Research on optimization of distributed big data real-time management method," in *Proceedings of the 2018 Third International Conference on Smart City and Systems Engineering (ICSCSE)*, pp. 626–630, Xiamen, China, December 2018.

[26] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.

[27] S. Raschka, *An Overview of General Performance Metrics of Binary Classifier Systems*, https://arxiv.org/abs/1410.5330, 2014.

[28] A. Dogan and D. Birant, "A weighted majority voting ensemble approach for classification," in *Proceedings of the 2019 4th International Conference on Computer Science and Engineering*, pp. 1–6, UBMK), Samsun, Turkey, September 2019.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 7384803, 8 pages
https://doi.org/10.1155/2022/7384803

Research Article

A Privacy-Preserved Variational-Autoencoder for DGA Identification in the Education Industry and Distance Learning

Xingxing Zheng and Xiaona Yin

Zhengzhou Preschool Education College, Zhengzhou 450000, China

Correspondence should be addressed to Xingxing Zheng; xingxing_zheng@126.com

Received 16 February 2022; Revised 23 February 2022; Accepted 26 February 2022; Published 24 March 2022

Academic Editor: Konstantinos Demertzis

One of the most insidious methods of bypassing security mechanisms in a modern information system is the domain generation algorithms (DGAs), which are used to disguise the identity of malware by periodically switching the domain name assigned to a command and control (C&C) server. Combating advanced techniques, such as DGAs, is an ongoing challenge that security organizations often need to work with and possibly share private data to train better and more up-to-date machine learning models. This logic raises serious concerns about data integrity, trade-related issues, and strict privacy protocols that must be adhered to. To address the concerns regarding the privacy and security of private data, we propose in this work a privacy-preserved variational-autoencoder to DGA combined with case studies from the education industry and distance learning, specifically because the recent pandemic has brought an explosive increase to remote learning. This is a system that, using the secured multi-party computation (SMPC) methodology, can successfully apply machine learning techniques, specifically the Siamese variational-autoencoder algorithm, on encrypted data and metadata. The method proposed for the first time in the literature facilitates learning specialized extraction functions of useful intermediate representations in complex deep learning architectures, producing improved training stability, high generalization performance, and remarkable categorization accuracy.

1. Introduction

The pandemic has had an impact on how people learn and on all stakeholders involved. The functioning of higher education institutions has been harmed. Many institutions are currently unable to perform examinations because of the pandemic, and the face-to-face teaching and learning process has been harmed. The current scenario is projected to last for some time, so it is critical to improve the learning process by establishing strategies with respect to online learning, preserving social distance, and, of course, the privacy of the data exchanged online by the various stakeholders [1, 2]. Researchers have been focusing on the growth of distance education in recent years, but because of the COVID-19 pandemic, distance learning has become a critical task for the education system and the privacy related issues.

For several years, botnet technology has been the mainstay of orchestrating and supporting a wide variety of cyber-attacks, such as DDoS attacks, and phishing. Malicious botnet masters use DGAs extensively to make it possible for the C&C server to communicate with their bots in such a way as to bypass the known malware detection mechanisms. Typically, a DGA algorithm, using a seed known only to devices participating in the botnet, periodically generates, at random times, a pseudo-random set of domain names, which act as candidates for assignment to the C&C server [3]. As a result, traditional static botnet handling techniques are becoming ineffective as the pace at which C&C changes names is unable to detect and terminate communication in time. Examples of such static techniques include blacklisting the static domain name of a C&C server as soon as it is detected, reverse-engineering the malware of an infected device to reconstruct the DGA used roughly and

studying how the malware generates names. Ideally, however, malicious names should be detected in real-time, with the predictions being performed at the level of individual names, to avoid establishing bots' communication with the C&C server [4].

The field of machine learning has dramatically attracted the interest of cyber security researchers to address this problem [5]. In previous approaches based on machine learning methods, a DGA [6] name detector required the extraction of human-defined statistical attributes to be effective. Instead, machine learning techniques automatically extract the necessary features during the training process while relying solely on the domain name string to make the required predictions, categorizing the names between DGA and legit. This feature is handy as malware is no longer aware of the components used to train model detectors. As a result, they cannot modify DGAs to generate names that are not detected based on these characteristics. Models trained in machine learning techniques are highly accurate and efficiently generalized, as presented in the relevant literature [7, 8].

The rest of the work includes Section 2, which provides an overview of the methods found in the literature and related to similar methods. Section 3 describes in detail the motivation of this work. Section 4 includes the methodology of the proposed system. Section 4 explains the dataset used, and Section 5 analyzes the experiments for implementing the proposed approach. Finally, Section 6 summarizes the research conducted and presents the future objectives that can extend it.

2. Literature Review

With the increasing use of modern technologies in every field globally, the need to enhance the cyber security posture of educational organizations has significantly grown in recent years, especially in 2020 and after the COVID-19 pandemic. Many modern institutions have shifted their services to a remote-based approach. The research community has also focused on the education industry and distance learning to find innovative and privacy-preserving solutions to this new reality [2, 9].

Zhang et al. [10] reviewed current research on privacy-preserving technologies and deep collaborative learning. They concluded that each privacy-preserving technology has its own unique features. They asserted that while safe multiparty computing and homomorphic encryption can provide a high level of privacy and accuracy, the cost to users is a substantial computational and communication burden. Differential privacy, in which users input random noise into their data before sending it to the server, is a more practical and efficient solution. Nonetheless, it diminishes the model's accuracy. When huge businesses with sensitive data operate as users, homomorphic encryption technology is essential to assure the model's security. Differential privacy technology is required when a large number of people with little computer power operate as consumers to ensure the model's efficiency. A mix of safe multiparty computing, homomorphic encryption, and differential privacy is being used in a growing number of studies to provide a suitable trade-off between data privacy and utility.

Ryffel et al. [11] developed and explored a federated learning framework built on PyTorch for privacy-preserving deep learning. Their framework prioritized data ownership and secure processing. It proposed a representation based on command chains and tensors, allowing them to perform complicated privacy-preserving structures like federated learning, SMPC, and differential privacy while presenting a familiar deep learning API to the end-user. The Boston Housing and Pima Indian Diabetes datasets were used to test their implementation. Nonetheless, they discovered early in the development process that the framework added an enormous performance burden.

Bo Chang [12] used vignettes to depict various privacy scenarios in the online learning environment while researching student privacy issues in online learning environments. Because of the sharing of individual grades among group members and providing public input on blogs, his studies revealed direct legal ramifications of concern to students that are not entirely addressed in FERPA policy. Other privacy issues, such as open access to each other's work, transparent reflections, public comments, critical examination of assignments, and collaborative evaluations of students' work, arose in more nuanced ways. He suggested focusing on students' products rather than their names, informing students of the benefits they will receive and the contributions they will make by sharing their work publicly, and providing options for students to keep their identities private if they are uncomfortable about sharing their work publicly. To alleviate students' discomfort, he also emphasized that professors should educate students about the FERPA policy while adapting to partial privacy.

In his research study, Karunakaran [3] employed both public and real-time environmental datasets to detect text features and knowledge-based feature extraction to detect DGAs that randomly produced malicious domains. Because attackers only know how the DGA method works, he surveyed an algorithm to identify the DGA more efficiently. His model produced extremely excellent classification accuracy results. Finally, he suggested that by training and evaluating the dataset, he might improve the proposed technique.

From the above literature, we realize that researchers put their efforts into employing deep learning techniques and finding the best possible trade-off between privacy and utility because of the high processing power that these methods require.

3. Motivation

As it turned out, there are quite a few capable systems for categorizing DGAs, but their ability depends mainly on the dataset used for their training. Most of the published research uses DGA names that are publicly available and have resulted from published related projects and successful reverse-engineering efforts for various DGA families. The problem with these datasets is the limited number of names they consist of and, secondly, the large percentage of them that are obsolete [13]. Admittedly, they lack names from

more recent DGAs, reducing the discernment of trainee models regarding emerging DGAs. On the other hand, organizations specializing in cyber security and ISP providers train their models, using for the training process data that they are in no way willing to share due to competition and financial interests. The problem becomes more realistic if we consider that one of the largest sources of both DGA or legit names, a recursive DNS server, obeys strict privacy protocols, making it impossible to access essential data (e.g., logs with DNS queries) for the best and most up-to-date training of a machine learning model [14, 15]. So, we come up with a scenario in which, while all stakeholders want to enhance their models, using the broadest possible variety and the most recent data available for training. No one wants to contribute to this direction by exposing their data to the public.

The above situation seems to lead to a dead end. To avoid compromising privacy while promoting scientific research into large datasets to improve digital security, it is imperative to simultaneously implement technical solutions to address data protection and usage requirements [16, 17]. A serious answer to this situation is using secured multiparty computation (SMPC) [18] cryptography to train a deep learning model, combining all available data during training without disclosure and meeting the requirements for privacy [7, 13, 19].

SMPC is a cryptographic technique that allows different parties to perform calculations through inputs while keeping these inputs private [18, 20, 21]. Essentially, in this model, a set of parts with private inputs performs distributed functions, ensuring the required privacy and security issues. Conceptually, SMPC replaces a reliable intermediary to implement reliable calculations [21–25].

4. Proposed Methodology

This work proposes a privacy-preserved Siamese variational-autoencoder for DGA network traffic identification. Variational autoencoders (VAEs) are neural networks that try to discover the internal structure of input data to produce similar data [26, 27]. In other words, these are models that try to display the characteristics of the data $\varphi(x)$ and the categories $\pi(y)$ in an embedding space. The main idea is that data from the same type should be displayed in the same area. The category description will be displayed, while data from different categories should be displayed in other areas [28, 29]. This creates a partition of the embedding space in $|Y|$ areas. Then a simple architecture, such as a simple classifier, is used to learn this separation. It is trained to classify the common integration space points in $|Y|$ possible classes. Finally, at the model evaluation time, the data (either belonging to known or unknown classes) is encoded in the embedding space and then classified into one of the available categories. The success of this approach is that the projection in the field of standard integration is made both from the areas of the known and unknown categories available during the training process [30, 31].

The proposed work uses Siamese, i.e., the parallel use of 2 autoencoders that encode and decode DGAs and their descriptions in one embedding space. To synchronize the areas of embedding space, the maximum mean discrepancy (MMD) metric is used in the model error function. Minimizing this amount synchronizes the probability distributions of DGAs and their categories in embedding space.

$$\begin{aligned}\mathscr{L}_{VAE} = \mathscr{L}_{VAE_1} + \mathscr{L}_{VAE_2} = \beta D_{KL}\left(p_{E_1}(z|\phi(x))\| p(z)\right) - \mathbb{E}_{p_{E_1}(z|\phi(x))}\left[\log p_{D_1}(\phi(x)|z)\right] + \\ \beta D_{KL}\left(p_{E_2}(z|\pi(y))\| p(z)\right) - \mathbb{E}_{p_{E_2}(z|\pi(y))}\left[\log p_{D_2}(\pi(y)|z)\right].\end{aligned} \tag{1}$$

where β is the model hyperparameter, E_1 and D_1 are the encoder and decoder of DGA, and E_2 and D_2 are the encoder and decoder of the descriptions.

Siamese VAE encodes the characteristics $\varphi(x)$ in a probabilistic profile, which is modeled as standard, so it depends on an average value and a scatter table. Then a point of the standard integration space $z\ N\ (\mu, \Sigma)$ is sampled and decoded. The error is added to the total error function of the model so that [27, 32]

The MMD ensures that the DGAs and descriptions are adequately decoded and form areas in the embedding space, but their distributions are synchronized by minimizing the Wasserstein distance to which it applies as follows [29, 33, 34]:

$$W_p(\mu, \nu) := \left(\inf_{\gamma \in \Gamma(\mu,\nu)} \int_{M\times M} \mathrm{d}x, y^p \mathrm{d}\gamma(x, y)\right)^{1/p}. \tag{2}$$

Finally, the synchronization of the distributions is calculated as follows:

$$\mathscr{L}_{DA} = \sqrt{\mu_1 - \mu_2^2 + \sum_1^{1/2} - \sum_2^{1/2} 2\,\text{Frobenius}}. \tag{3}$$

Although at this point the model works satisfactorily, the cross-synchronization technique is additionally used where

$$\mathscr{L}_{CA} = \phi(x) - D_1(E_2(\pi(y))) + \pi(y) - D_2(E_1(\phi(x))). \tag{4}$$

The VAE of the descriptions is required to decode DGAs, and the DGA decoder to decode descriptions, so the total error function of the model is

$$\mathscr{L}(x, y; E, D) = \mathscr{L}_{VAE} + \gamma\mathscr{L}_{CA} + \delta\mathscr{L}_{DA}. \tag{5}$$

where γ, δ are hyperparameters of the model.

Finally, a simple Softmax classifier is used to classify embedding space in $|Y|$ categories. An indicative architecture of the proposed system is shown in Figure 1.

To engage stakeholders who wish to enhance their models using the broadest possible variety and the most recent data available for education without the potential exposure of their private data, a machine learning protocol based on the SMPC technique is implemented [20, 22]. The proposed function offers participants the same possibility, as it allows the calculation of its value F only through the exchange of messages between n participants. Such a calculation could theoretically be performed in the presence of an inviolable and trustworthy referee other than n the participants, to whom each would give his value d_x, and he would correctly calculate its value F and announce to everyone only the result, as would be the case with the use of federated learning techniques.

In our case, we are interested in and present the implementation of a cumulative protocol for calculating the function $(d_1, \ldots, d_n) = d_1 + d_2 + \cdots + d_n$ based on the Shamir secret sharing (3S) method for n participants with n threshold. The 3S algorithm is based on the secure splitting and sharing of information between several participants. Each of them receives a value unrelated to the secret (in this case, the training data of the machine learning model is considered a secret), called a share of the secret, which has no utility. The secret can only be recreated if several parts of it are combined. For a total number of n shares to be defined, the minimum number $t \leq n$ is set initially, called the threshold (t, n) required to recover the secret S. $t-1$ random integers $a_1, a_2, \ldots, a_{k-1}$ are selected while $a_0 = S$, to implement the following polynomial [18, 22, 35]:

$$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{t-1} x^{t-1}. \tag{6}$$

Based on this polynomial, we obtain n random points $(i, f(i)): \neq 0$. Each point is communicated to one of the n participants. Having the polynomial (x), for the value $x = 0$, we get the value $(0) = a_{0,}$ which is the secret S. It is noted that to maintain the correct secrecy, all operations are done with elements of a finite field F with size P where P first number, greater than all the coefficient values of the polynomial and the values t and n.

For any subset of t points, the above polynomial can be reconstructed using the Lagrange interpolation. Specifically, let $n+1$ points $(x_0, y_0), \ldots, (x_j, y_j), \ldots, (x_n, y_n)$, where all x_j are different from each other. The Lagrange interpolation polynomial of $P_n(x)$ degree $\leq n$ is given by the type as [22, 36]

$$P_{n(x)} = l_0(x) f(x_0) + l_1(x) f(x_1) + \ldots + l_n(x) f(x_n) = \sum_{i=0}^{n} l_i(x) f(x_i). \tag{7}$$

with:

$$l_i(x) = \prod_{0 \leq j \leq n, j \neq i} \frac{x - x_j}{x_i - x_j}. \tag{8}$$

The remainder can be bound as

$$|Rx| \leq \frac{(x_k - x_0)^{k+1}}{(k+1)!} \max_{x_0 \leq \xi \leq x_k} \left| f^{(k+1)}(\xi) \right|. \tag{9}$$

The proposed protocol has the following steps [36, 37]:

(1) Each participant p with a value d creates a random polynomial of n-th degree with a fixed value its hidden value d_p, as $f(x) = d_p + a_1 x + a_2 x^2 + \cdots + a_n x^n$.

(2) Calculates n values of (x) for n different but predefined values x_p with $x_p \neq 0$, one for each participant, including himself.

(3) Sends to each participant p the corresponding value (x_p).

(4) Steps 1 to 3 are performed by all participants, and each one sends the corresponding values of the random polynomial (x). An essential element of the process is that f_p values are not sent randomly. Assuming that each default value x_p is assigned to a specific participant p, then the corresponding value of (x_p) for the corresponding x must be sent to him by all participants.

(5) Having each participant p receive n values, $f_1(x_p)$, $f_2(x_p), \ldots, f_n(x_p)$ calculates their sum and notifies it to the other participants.

(6) When all the sums have been announced, each participant uses them to perform Lagrange interpolation and reconstruct a new polynomial (x) equal to the sum of the random polynomials of all participants $f_{\text{all}}(x) = f_1(x) + f_2(x) + \ldots + f_n(x)$.

(7) Given the sum of all polynomials, it is expected that the constant of $c_{all} = d_1 + d_2 + \cdots + d_n$ and is calculated for $x = 0$, $(0) = c_{\text{all}} \sum$.

5. Dataset

Two different sets of domain names were used to carry out the experiments. In the first dataset, 400,000 records are used from nonwordlist-based DGAs alone. Half of them came from Alexa's collection of the top 1 million randomly selected from the most popular names, and the rest were created by running specific DGA algorithms. Ten different DGAs were executed in more detail, and 20,000 names were generated for each of the above algorithms. The second dataset uses wordlist-based DGAs and includes 500,000 records, half of which came from the Alexa collection of the top 1 million randomly selected from the most popular names [4, 6]. The rest were created by executing ten different wordlist-based DGAs. It should be noted that we

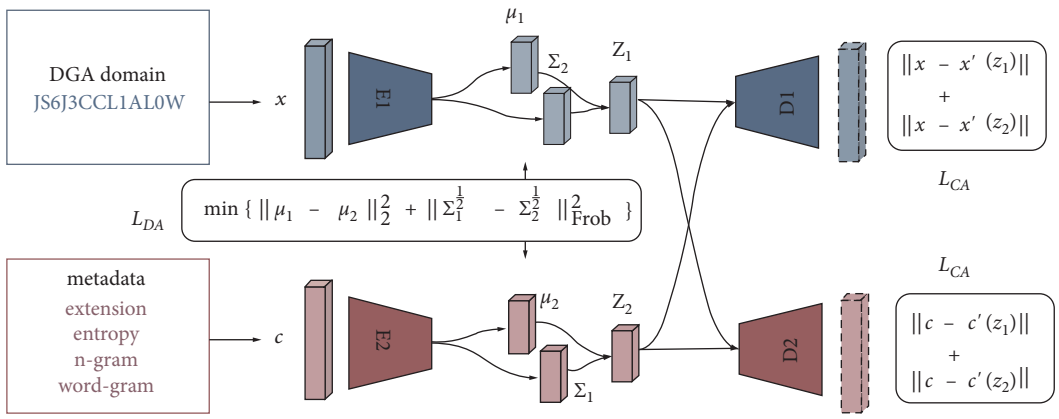


Figure 1: Siamese variational-autoencoder architecture.

used domains that were related to education institutions (.ac,.edu, etc.), and e-learning software (Zoom, Cisco Webex, etc.)

An evaluation dataset consisting of 1,500,000 domain names was used to evaluate the method. Of these 800,000 legit domains, which are different from those of the training dataset, 550,000 come from the Alexa top 1 million collections, while domain name registration machines retrieved the rest. Respectively, of the 700,000 DGAs domains, 300,000 are real DGAs domains registered in corresponding lists of security organizations such as BlackHoleDNS, while the remaining 400,000 have been created after executing 40 different DGAs. The following Figure 2 illustrates the dataset.

The 20 algorithms have not been included in the training dataset to make the dataset more realistic. However, even for those included, different seeds and wordlists were used in their execution, so that the domain names generated were different from those used for training. For the creation of DGA domains, the length of the domains was random, ranging from 6 to 21 alphanumeric characters written in the Latin alphabet. An entropy algorithm was also applied to the creation of the domains as a degree of uncertainty to enhance the realism of the generated domains. Even with the Alexa grams technique, the degree of sequence between the generated DGA domain and the list of domains derived from Alexa was calculated using the technique of the probabilistic model for the prediction of the next n-gram element. Finally, the word-gram process was used to calculate the degree of correlation - sequence between the DGA domain and 500,000 widely used words to predict the next word-gram element.

The metadata used for the second VAE was the end of the domain name, the degree of entropy of each domain (entropy), the degree of the sequence of each domain (n-gram), and the degree of correlation-sequence of each domain (word-gram) [32, 34].

Overall, Table 1 presents the datasets used in this study as follows.

6. Experiments

To evaluate the performance of the proposed system per class as well as the estimation of the actual error during the training, we used the following measures [26, 38]:

Sensitivity = tp/pos, Specificity = tn/negat, Precision = $tp/tp + fp$, recall = $tp/tp + fn$, and f − score = 2x pre x rec/pre + recaccuracy = sensitivity ∗ pos/pos + negat + specifivity ∗ neg/pos + negat = $tp + tn$/pos + negat

Where tn = true negative, tp = true positive, fn = false negative, and fp = false positive.

We conducted three experiments where, for the first time, training was performed with the nonwordlist-based DGA dataset and a test with the MixTest nonwordlist and wordlist DGA. Then training was done with the wordlist-based DGA dataset and testing was done with the MixTest nonwordlist and wordlist DGA. Finally, the two training datasets (nonwordlist-based DGA and wordlist-based DGA dataset) were combined, and the MixTrain nonwordlist was created and the wordlist DGA, which was tested with the MixTest nonwordlist and wordlist DGA. The following Table 2 shows the results of the followed procedure.

As can be seen from the table above, the generalizability of the system is significantly enhanced by the MixTrain (nonwordlist and wordlist-based DGA) dataset, which includes many more and much more complete samples of DGA domains.

In the case of applying the SMPC algorithm, we proved its functionality by proving that ring uniformities "retain" operations. Specifically, we demonstrated that there exists an isomorphism of rings of polynomials (i.e., be 1–1) where at least R is a transposition ring, and I is an ideal of R with [18, 25, 35]

$$\frac{R[t]}{I[t]} \xrightarrow{\cong} \left(\frac{R}{I}\right)[t]. \tag{10}$$

In detail, considering the illustration as follows:

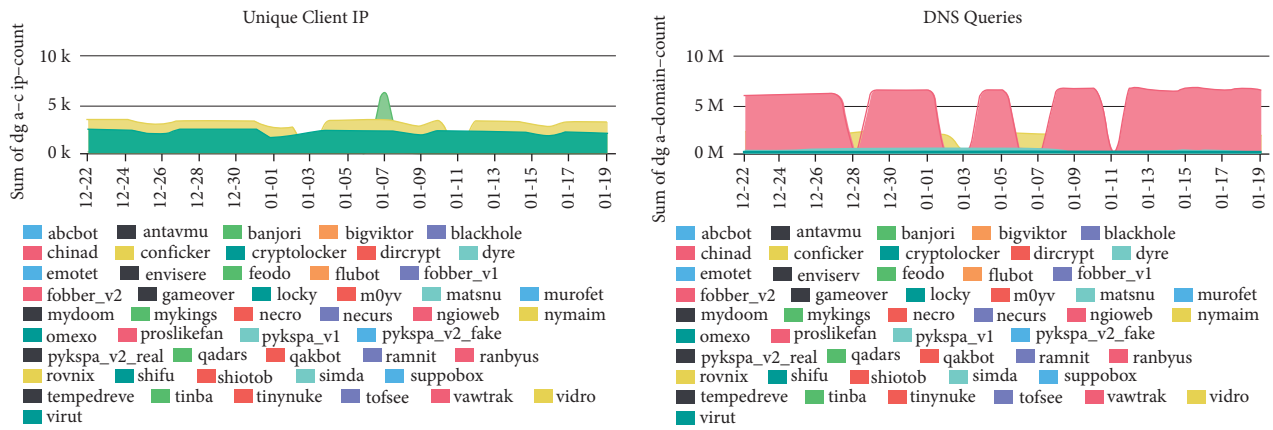


Figure 2: DGA domains unique client Ips and DNS queries by https://data.netlab.360.com/dga/.

Table 1: Training and test datasets.

Training datasets		Test dataset
Nonwordlist-based DGA	Wordlist-based DGA	MixTest nonwordlist and wordlist DGA
200.000 legit	250.000 legit	800.000 legit
200.000 DGA	250.000 DGA	700.000 DGA

Table 2: Results with various training datasets.

Training dataset	Accuracy	Recall	Precision	f-score
Nonwordlist-based DGA	0.8949	0.8883	0.8904	0.8903
Wordlist-based DGA	0.9072	0.9038	0.9056	0.9058
MixTrain (nonwordlist & wordlist-based DGA)	0.9260	0.9263	0.9259	0.9261

$$\Phi\colon \frac{R[t]}{I[t]} \longrightarrow \left(\frac{R}{I}\right)[t]., P(t) = \sum_{k=0}^{n} a_k t^k \longrightarrow \Phi(P(t)) = \sum_{k=0}^{n} (a_k + I)t^k. \tag{11}$$

Then $\Phi(1_{R[t]}) = \Phi(1) = 1 + I = 1_{(R/I)[t]}$, i.e., the illustration Φ sends the unit of R[t] to its unit (R/I)[t].

If $P(t) = \sum_{k=0}^{n} a_k t^k$ and $Q(t) = \sum_{k=0}^{m} b_k t^k$ are two polynomials in the ring $R[t]$, then we can assume without harm to the generality that $n \le m$ and then we can write $P(t) = \sum_{k=0}^{m} a_k t^k$, where we set $a_{n+1} = \cdots = a_m = 0$. So, we will have

$$\Phi(P(t) + Q(t)) = \Phi\left(\sum_{k=0}^{m} a_k t^k + \sum_{k=0}^{m} b_k t^k\right) = \Phi\left(\sum_{k=0}^{m} (a_k + b_k)t^k\right) = \sum_{k=0}^{m} (a_k + b_k + I)t^k =$$

$$= \sum_{k=0}^{m} (a_k + I)t^k + \sum_{k=0}^{n} (b_k + I)t^k = \Phi(P(t)) + \Phi(Q(t)). \tag{12}$$

Similarly, setting $c_k = \sum_{l=0}^{k} a_l b_{k-l}, 0 \le k \le n + m$ we will have

$$\Phi(P(t)\cdot Q(t)) = \Phi\left(\sum_{k=0}^{m} a_k t^k \cdot \sum_{k=0}^{m} b_k t^k\right) = \Phi\left(\sum_{k=0}^{n+m} c_k t^k\right) = \sum_{k=0}^{n+m} (c_k + I)t^k\Bigg) = \sum_{k=0}^{n+m}\left(\sum_{l=0}^{k}(a_l + I)(b_{k-l} + I)\right)t^k$$

$$= \sum_{k=0}^{n} (a_k + I)t^k \cdot \sum_{k=0}^{m} (b_k + I)t^k = \Phi(P(t)) \cdot \Phi(Q(t)). \tag{13}$$

Thus, the illustration Φ is a homomorphism of rings which in addition is a homomorphism because if $A(t) = \sum_{k=0}^{n} (a_k + I)t^k$ is a typical ring element $(R/I)[t]$, then setting $P(t) = \sum_{k=0}^{n} a_k t^k \in R[t]$ we will have $\Phi(\mathrm{P}(t)) = \mathrm{A}(t)$.

Let $P(t) = \sum_{k=0}^{n} a_k t^k \in \mathrm{Ker}(\Phi)$, then $\Phi(P(t)) = \sum_{k=0}^{n} (a_k + I)t^k = 0_{(R/I)[t]} = I$ is the zero polynomial in the ring $(R/I)[t]$, i.e., $a_k + I = I$, and therefore $a_k \in \mathrm{I}$, $0 \le k \le n$. This means that the polynomial $P(t) \in I[t]$. Conversely, if $P(t) \in I[t]$, then $a_k \in \mathrm{I}$, $0 \le k \le n$, and then obviously [21, 23]

$$\Phi(P(t)) = \sum_{k=0}^{n} (a_k + I)t^k = \sum_{k=0}^{n} (0_{R/I})t^k = 0_{(R/I)[t]}. \tag{14}$$

Therefore

$$Ker(\Phi) = I[t]. \tag{15}$$

and so, the subset $I[t]$ is an ideal of $R[t]$ as the nucleus of a ring homomorphism. Finally, because the imaging Φ is training, it follows that Φ induces the ring isomorphism of the original hypothesis as

$$\frac{R[t]}{I[t]} \xrightarrow{\cong} \left(\frac{R}{I}\right)[t]. \tag{16}$$

We can therefore use the methodology of the SMPC algorithm to apply machine learning techniques with very high performance even in cases of encrypted data.

7. Discussion and Conclusions

The detection and timely assessment of DGA domains and DGA network traffic allows for the detection of incidents and the corresponding identification of correlations and relationships with security incidents, significantly mitigating the effects of sophisticated cyber-attacks. Individual efforts by independent actors cannot perform effectively and quickly in the field of knowledge discovery. On the contrary, collaborative efforts, which, as it turns out, can be implemented with remarkable learning models that also work on encrypted data, can lead to a significant increase in the accuracy of results and the generalization of learning models. Also, the increasing nature of the data requires the rise of training datasets, always considering the adaptation of the method to the available memory resources and computing power.

Considering the need for realistic and accurate security incident detection systems, this paper presented an innovative and highly practical privacy-preserved machine learning methodology for the timely detection of DGA domains and the network traffic they generate, with respect to distance education functionality. This methodology combines the Siamese variational-autoencoder in a complete framework. It is a robust system that calculates the number of maximum probable intervals within which an event is likely to occur based on a parametric evaluation that uses realistic datasets.

An essential advantage of the method, which has been proven experimentally, is that VAEs can, by receiving a combination of data and metadata, detect complex and sophisticated DGA domains. The dynamic identification of the proposed system directly integrates all the information in the sequence of the sample set, creating conditions for a realistic approach in recognizing security events.

Significant improvements in the evolution of the proposed system mainly concern the optimization of VAE hyperparameters, which are sensitive to modifications in determining the input data trend. Also, a significant improvement involves how the system is investigated with dynamic variational inference methodologies to provide a detailed approach to the subsequent probability of unobserved variables and apply a statistical conclusion for these variables.[39].

Data Availability

The data used in this study are available from the author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This study was supported by the Project of Exploration on the Training Model of Primary School English Teachers in the Context of Professionalism (No. 2016-JSJYYB-207).

References

[1] B. Paris, R. Reynolds, and C. McGowan, "Sins of omission: Critical informatics perspectives on privacy in e-learning systems in higher education," *Journal of the Association for Information Science and Technology*, 2021.

[2] D. Turnbull, R. Chugh, and J. Luck, "Transitioning to *E*-learning during the COVID-19 pandemic: how have higher education institutions responded to the challenge?" *Education and Information Technologies*, vol. 26, no. 5, pp. 6401–6419, 2021.

[3] P. Karunakaran, "Deep learning approach to DGA classification for effective cyber security," *J. Ubiquitous Comput. Commun. Technol.*vol. 2, no. 4, pp. 203–213, 2021.

[4] K. Demertzis and L. Iliadis, "Evolving Smart URL Filter in a Zone-Based Policy Firewall for Detecting Algorithmically

Generated Malicious Domains," in *Proceedings of the Statistical Learning and Data Sciences*, pp. 223–233, SLDS, London, UK, Aprill 2015.

[5] K. Shaukat, S. Luo, V. Varadharajan et al., "Performance comparison and current challenges of using machine learning techniques in cybersecurity," *Energies*, vol. 13, no. 10, p. 2509, 2020.

[6] T. Chin, K. Xiong, C. Hu, and Y. Li, "A machine learning framework for studying domain generation algorithm (DGA)-Based malware," in *Proceedings of the Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 254, Rochester, USA, December 2018.

[7] J. Gawlikowski, C. R. N. Tassi, M. Ali et al., "A Survey of Uncertainty in Deep Neural Networks," 2021, http://arxiv.org/abs/2107.03342.

[8] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection," *ACM Computing Surveys*, vol. 54, no. 2, pp. 1–38, 2022.

[9] A. Klašnja-Milićević and M. Ivanović, "E-learning personalization systems and sustainable education," *Sustainability*, vol. 13, no. 12, p. 6713, 2021.

[10] D. Zhang, X. Chen, D. Wang, and J. Shi, "A survey on collaborative deep learning and privacy-preserving," in *Proceedings of the 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pp. 652–658, IEEE, Guangzhou, Jun, june-2018.

[11] T. Ryffel, A. Trask, M. Dahl et al., "A generic framework for privacy preserving deep learning," 2018.

[12] B. Chang, "Student privacy issues in online learning environments," *Distance Education*, vol. 42, no. 1, pp. 55–69, 2021.

[13] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.

[14] Z. Yu, A. M. Abdulghani, A. Zahid, H. Heidari, M. A. Imran, and Q. H. Abbasi, "An overview of neuromorphic computing for artificial intelligence enabled hardware-based hopfield neural network," *IEEE Access*, vol. 8, pp. 67085–67099, 2020.

[15] K. Al Jallad, M. Aljnidi, and M. S. Desouki, "Anomaly detection optimization using big data and deep learning to reduce false-positive," *Journal of Big Data*, vol. 7, no. 1, p. 68, 2020.

[16] T. Alshalali, K. M'Bale, and D. Josyula, "Security and privacy of electronic health records sharing using hyperledger fabric," in *Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 760–763, IEEE, Las Vegas, NV, USA, Sep. 2018.

[17] M. Dhingra, M. Jain, and R. S. Jadon, "Role of artificial intelligence in enterprise information security: a review," in *Proceedings of the 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pp. 188–191, IEEE, Waknaghat, Dec. 2016.

[18] C. Zhao, S. Zhao, M. Zhao et al., "Secure multi-party computation: theory, practice and applications," *Information Sciences*, vol. 476, pp. 357–372, 2019.

[19] A. Darwish, "Bio-inspired computing: algorithms review, deep analysis, and the scope of applications," *Future Computing and Informatics Journal*, vol. 3, no. 2, pp. 231–246, 2018.

[20] J. Bringer, H. Chabanne, and A. Patey, "Privacy-preserving biometric identification using secure multiparty computation: an overview and recent trends," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 42–52, 2013.

[21] B. Jiang, "Two-party secure computation for any polynomial function on ciphertexts under different secret keys," *Security and Communication Networks*, vol. 2021, pp. 1–7, 2021.

[22] H. Akbari-Nodehi and M. A. Maddah-Ali, "Secure coded multi-party computation for massive matrix operations," *IEEE Transactions on Information Theory*, vol. 67, no. 4, pp. 2379–2398, 2021.

[23] R. Hou, F. Tang, S. Liang, and G. Ling, "Multi-party verifiable privacy-preserving federated k-means clustering in outsourced environment," *Security and Communication Networks*, vol. 2021, pp. 1–11, Article ID 3630312, 2021.

[24] S. Zavrak and M. Iskefiyeli, "Anomaly-based intrusion detection from network flow features using variational autoencoder," *IEEE Access*, vol. 8, Article ID 108346, 2020.

[25] M. Sepehri, S. Cimato, and E. Damiani, "Privacy-preserving query processing by multi-party computation," *The Computer Journal*, vol. 58, no. 10, pp. 2195–2212, 2015.

[26] T. W. Anderson, "An introduction to multivariate statistical analysis," Wiley, NY, London, Sydney, 2003.

[27] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[28] Y. Bian and X. Tang, "Abnormal detection in big data video with an improved autoencoder," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–6, Article ID e9861533, 2021.

[29] Z. Gu and Y. Yang, "Detecting malicious model updates from federated learning on conditional variational autoencoder," in *Proceedings of the 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 671–680, IEEE, Portland, OR, USA, May 2021.

[30] E. Protopapadakis, A. Voulodimos, A. Doulamis, N. Doulamis, D. Dres, and M. Bimpas, "Stacked autoencoders for outlier detection in over-the-horizon radar signals," *Computational Intelligence and Neuroscience*, vol. 2017, pp. 1–11, Article ID e5891417, 2017.

[31] X. Xu, J. Li, Y. Yang, and F. Shen, "Toward effective intrusion detection using log-cosh conditional variational autoencoder," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6187–6196, 2021.

[32] C. Doersch, "Tutorial on Variational Autoencoders," 2021, http://arxiv.org/abs/1606.05908.

[33] M. M. Abdelsamea, G. Gnecco, M. M. Gaber, and E. Elyan, "On the relationship between variational level set-based and SOM-based active contours," *Computational Intelligence and Neuroscience*, vol. 2015, pp. 1–19, Article ID e109029, 2015.

[34] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2014, http://arxiv.org/abs/1312.6114.

[35] Y. Sun, Q. Wen, Y. Zhang, and W. Li, "Privacy-preserving self-helped medical diagnosis scheme based on secure two-party computation in wireless sensor networks," *Computational and Mathematical Methods in Medicine*, vol. 2014, pp. 1–9, Article ID e214841, 2014.

[36] Z. Wang, S.-C. S. Cheung, and Y. Luo, "Information-theoretic secure multi-party computation with collusion deterrence," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 980–995, 2017.

[37] L. Zhou, L. Wang, Y. Sun, and T. Ai, "AntNest: fully non-interactive secure multi-party computation," *IEEE Access*, vol. 6, Article ID 75639, 2018.

[38] S. Raschka, "An Overview of General Performance Metrics of Binary Classifier Systems," 2014, http://arxiv.org/abs/1410.5330.

[39] R. Beyah, B. Chang, Y. Li, and S. Zhu, *Springer International Publishing*, Springer, Berlin, Germany, pp. 433–448, 2018.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 1945507, 10 pages
https://doi.org/10.1155/2022/1945507

Research Article

A Machine Vision Anomaly Detection System to Industry 4.0 Based on Variational Fuzzy Autoencoder

Wei Jiang

Zhengzhou College of Finance and Economics, Zhengzhou 450000, China

Correspondence should be addressed to Wei Jiang; jiangwei198308@163.com

Received 5 February 2022; Accepted 18 February 2022; Published 16 March 2022

Academic Editor: Konstantinos Demertzis

From a technological point of view, Industry 4.0 evolves and operates in a smart environment in which the real and virtual worlds come together through smart cyber-physical systems. These devices that control each other autonomously activate innovative functions that enhance the production process. However, the industrial environment in which the most modern digital automation and information technologies are integrated is an ideal target for large-scale targeted cyberattacks. Implementing an integrated and effective security strategy in the Industrial 4.0 ecosystem presupposes a vertical inspection process at regular intervals to address any new threats and vulnerabilities throughout the production line. This view should be accompanied by the deep conviction of all stakeholders that all systems of modern industrial infrastructure are a potential target of cyberattacks and that the slightest rearrangement of mechatronic systems can lead to generalized losses. Accordingly, given that there is no panacea in designing a security strategy that fully ensures the infrastructure in question, advanced high-level solutions should be adopted, effectively implementing security perimeters without direct dependence on human resources. One of the most important methods of active cybersecurity in Industry 4.0 is the detection of anomalies, i.e., the identification of objects, observations, events, or behaviors that do not conform to the expected pattern of a process. The theme of this work is the identification of defects in the production line resulting from cyberattacks with advanced machine vision methods. An original variational fuzzy autoencoder (VFA) methodology is proposed. Using fuzzy entropy and Euclidean fuzzy similarity measurement maximizes the possibility of using nonlinear transformation through deterministic functions, thus creating an entirely realistic vision system. The final finding is that the proposed system can evaluate and categorize anomalies in a highly complex environment with significant accuracy.

1. Introduction

The systems that make up the industrial environment in the Industry 4.0 standard and those inherited from the existing infrastructure show enormous heterogeneity resulting in a massive number of different interfaces with different characteristics and security requirements [1]. Unfortunately, modern architectural standards do not determine the organization of heterogeneous systems based on the essential security requirements, which translates into a significant increase of the attack surface for possible cyberattacks [2]. It is important to emphasize that cyberattacks in the industrial environment can be implemented as a rearrangement of the operation of mechatronic equipment, the configuration of different signs or alerts, the omission of steps in the production process, and so on [1, 3].

The current situation focuses more on the human factor, the experience, and the opinion of experts, using assistive technology to assess and mitigate risks and threats. There should be in-depth human supervision and intervention by highly qualified staff for best results with this approach. On the other hand, the rapid development of computer systems has led to artificial intelligence mechanisms to solve complex problems without human intervention [4].

One of the critical areas of application of computational intelligence algorithms is the recognition of anomalies in real-time machine vision systems [5]. The detection of abnormalities is wholly related to recognizing patterns in a dataset that depicts different behaviors than expected [6]. The goal is to detect possible deviations while maintaining low false alarm rates. The activity is monitored in real time

with a regular pattern when applying anomaly detection algorithms [7]. When a deviation is detected, the safety management mechanism is activated to investigate the incident further and take measures to deal with it.

In an industrial environment, anomaly detection is used to intelligently identify divergent behavior that could lead to mechanical failure or other adverse conditions. This process provides a robust security mechanism for industry hubs and business network systems within Industry 4.0 [2, 8].

An additional objective of anomaly detection is the immediate identification of irregular use, misuse, and abuse of industrial systems by external factors and equipment failure cases [9]. The industrial environment consists of scattered heterogeneous nodes that exchange information through a common automated communication infrastructure. In this context, the complexity increases exponentially as the number of interconnected systems expands [10].

The main problem in the environment mentioned above concerns the heterogeneity in detecting anomalies, which imposes the integration of intelligent machine vision systems in many industrial systems. The purpose of these applications is to alert cyber-physical systems when items outside of predetermined specifications appear on the production line so that corrective decisions can be made promptly to ensure product quality and productivity [3]. These systems are adaptive and can deal with the uncertainty of the environment in which they are applied. Similarly, with the integration of vision in the production process, it is possible to detect abnormalities through visual inspection in time, offering significant benefits, especially in construction problems or material failures [11].

This work aims to create a machine vision device to ensure the quality of metal components in the automotive sector, where Industry 4.0 standards are applied. In particular, the VFA algorithm is proposed, which can detect poor assembly alignment in gearboxes that may be due to cyberattacks [12]. The process of detecting these anomalies is achieved by using blurred entropy and Euclidean measurement of blurred similarity between samples, thus creating an entirely realistic and highly reliable machine vision system.

The ability of automated visual inspection on the production line to detect anomalies, given that it targets many industrial products, has been a constant research demand, thoroughly investigated by the research community, with significant developments depicted in the relevant literature [7, 13, 14].

The rest of the work includes Section 2, which provides an overview of the methods found in the literature and related to similar technical standardization. Section 3 describes in detail the methodology of the proposed system. In contrast, Section 4 explains the scenarios for implementing the proposed approach. Finally, Section 5 summarizes the research conducted and presents the future objectives that can extend it.

2. Literature Review

The concept of anomaly detection using artificial intelligence has been approached with various methods from the research community because of the numerous challenges involved, such as the vast amount and diversity of data to be analyzed. In recent years, we observed that the community has been researching various autoencoder combinations to solve complex problems effectively [15, 16]. Because of the depth and richness of information, the universality of applications, and the difficulty of monitoring processes, this research is becoming increasingly important.

Zimmerer et al. [17] demonstrated an anomaly detection method for identifying and determining aberrant spots in medical imaging. They used a mix of density and reconfiguration-based anomaly detection algorithms, which did not require labeled data and allowed for sample-by-sample anomaly scoring and determination. They showed how to boost anomaly scores by using a context encoder and a variational autoencoder. In a variational autoencoder for pixel-wise anomaly localization, they added the posterior deviations (KL divergence) from the prior latent variable distributions. They also employed a variational autoencoder to combine the previous variations with the reconversion error to improve localization, achieving encouraging results with the potential to improve and speed up future medical picture review and assessment.

Lee et al. [2], in 2018, introduced a sparse representation framework for large-scale and high-dimensional data that builds dictionaries depending on the subspace of variational autoencoder (VAE). This autonomous framework injects minimal reconstruction into VAE, which is divided into two parts: secret data mapping and concise dictionary info generation. It can be used to uncover secret data and extract more high-level features than hand-crafted features in large-scale datasets by acting as a dimensionality reducer.

Carletti et al. [18] proposed a method for determining "feature importance" in anomaly detection, with the goal of addressing the barriers to ML adoption in Industry 4.0 scenarios. The absence of supervised datasets makes intelligent monitoring systems difficult. The feature point importance evaluation method is intended for isolation forest, among the most widely used anomaly detection methods.

Banifakhr and Sadeghi [19] demonstrated a method for detecting anomalies in trajectories using CCTV records of vehicle traffic. The method makes use of machine learning and deep learning techniques to overcome the problems of not having enough data to build an effective model and not having enough anomaly data to cover all conceivable aberrant trajectories. They solved the challenge by combining optimal convolutional neural network and adaptive neuro-fuzzy inference system network classifiers with an autoencoding network to create an optimized structure for anomaly detection at the decision level. The classifier first categorizes the input trajectory into one of the specified groups. The result is then evaluated by the trained autoencoder networks to determine whether the route is regular or aberrant.

Tsai and Jen [14] sought to detect surface defect irregularities using an autoencoder. On the one hand, they did not use pixel-wise flaw separation, but instead used photo detectability. A normalization was included in the suggested convolutional autoencoder, which enhances the characteristic dispersion of fault examples within a small spectrum. This method brings all training samples' representative

feature vectors as nearly as possible to the average feature vector. In the evaluation stage, a defect sample can produce a different range from the learned center of fault samples. They also added two normalization penalties that could limit the spread of retrieved properties from a group of fault samples to a small area. For less-regular texture backdrops, the first regularization is learned for uniformly surface areas, while the second could further differentiate the faulty features.

For evaluating ultrasonic testing (UT) data, Milkovic et al. [15] suggested a variational autoencoder (VAE). In standard UT data, the VAE was applied to characterize the distributions. Their strategy was to train on normal data only, which resulted in variations in VAE output and latent values in cases with abnormal data, which served as a foundation for anomaly identification. The problem of detecting anomalies in ultrasound pictures necessitated the use of numerous criteria. First, they rebuilt the error and variances of the mean and standard deviation of latent variable parameters. Then, on top of the decoder, they added a second encoder, allowing the use of two new parameters, which merged reconstructions and hidden descriptions as potential anomaly signs.

Finally, in 2022, Lu et al. [7] introduced a deep learning-based anomaly detection method for identifying lace faults in industrial settings. Lace is unique in that it is one of the only industrial items that is completely dependent on manual fault control. Video preprocessing, pixel rebuilding, and pixel categorization were the three stages of their system. Only defect-free lace films are required during the offline phase to train the pixel reconstruction model and determine the detection threshold using the adaptive thresholding method. The proposed framework reconstructs lace videos and conducts defect inspection utilizing reconstruction error and a predetermined threshold in the online stage. On holes and damaged yarn, their model worked perfectly. To overcome the dataset deficit, they aimed to gather more faulty samples, which is a time-consuming technique, and analyze the lace pattern layout. They also aimed to explore the pixel reconstruction model to obtain more precise rebuilding findings, which can help distinguish small problems and noises.

From the above literature, we can conclude that the research community is primarily trying to find a practical machine learning approach to solve complex problems with the most effective methods.

3. Proposed Machine Vision-Based Anomaly Detection System

In the present work, a holistic approach to anomaly recognition in machine vision systems is implemented and proposed, based on an original VFA methodology where the possibility of using nonlinear transformation through deterministic functions is maximized. This is an innovative model of artificial vision, for optimal decision making, regarding the recognition of anomalies in the industrial environment. Specifically, we present a novel methodology using fuzzy entropy and Euclidean fuzzy similarity measurement for the first time in the literature, in order to maximize the possibility of using nonlinear transformation through deterministic functions, thus creating an entirely realistic and highly reliable machine vision detection system for Industry 4.0 based on variational fuzzy autoencoder [15, 20].

The evaluation of the methods was carried out in a highly complex cybersecurity scenario, where cybercriminals could modify the assembly parameters of the production mechanisms. This fact is not perceived by the other sensors connected to the production system. Utilizing the most advanced machine vision techniques and fuzzy logic methodologies, the proposed method has achieved very high success rates, creating serious expectations for additional cybersecurity applications [20, 21]. A depiction of the autoencoder architecture is presented in Figure 1.

The VFA architecture layout contains a hidden layer consisting of D neurons. The encoder encodes the input vector x into the vector h. Each h_i coordinate corresponds to the output of a hidden layer neuron so that

$$h_i = f_i\left(w_{ei}^T x + b_{ei}\right), \tag{1}$$

where f_i is the activation function and w_{ei} and b_{ei} are the parameters of the ith neuron of the encoder. The decoder then decodes the representation by producing

$$\tilde{x}_i = g_i\left(w_{di}^T h + b_{di}\right), \tag{2}$$

at output i, where g_i is the activation function and w_{di} and b_{di} are the parameters of the ith neuron of the decoder. The training is done by minimizing the loss function:

$$J(x, g(f(x))). \tag{3}$$

An easy way for the encoder to learn valuable features is through the $D < N$ constraint, i.e., the dimensionality of the hidden representation is less than the dimensionality of the data. In this case, the encoder encrypts any incoming information, and then the decoder tries to reconstruct the input. Because $D < N$ is valid, some of the information contained in the attribute space is lost. The decoder attempts to recover the lost data through h. The network, therefore, tries to trap as much information as possible in vector h, neglecting potentially useless information contained within the attribute space. If each x_i comes from an independent and identically distributed (iid) distribution independent of the others, then h rarely contains any helpful information. However, if there is any structure between the data, the autoencoder can detect it.

Another way to export useful features is by applying sparse restrictions to the network. For this purpose, additional constraints are introduced in the loss function that forces the network's neurons to be activated less frequently so that the h_i are as detachable as possible. Typical limitations concern the matrix of network weights, such as the norm L_1 or L_2. The parameter λ corresponds to a hyperparameter of the network, which is determined during its training. High values of the parameter give further power to the constraint by reducing the values of the network weights:

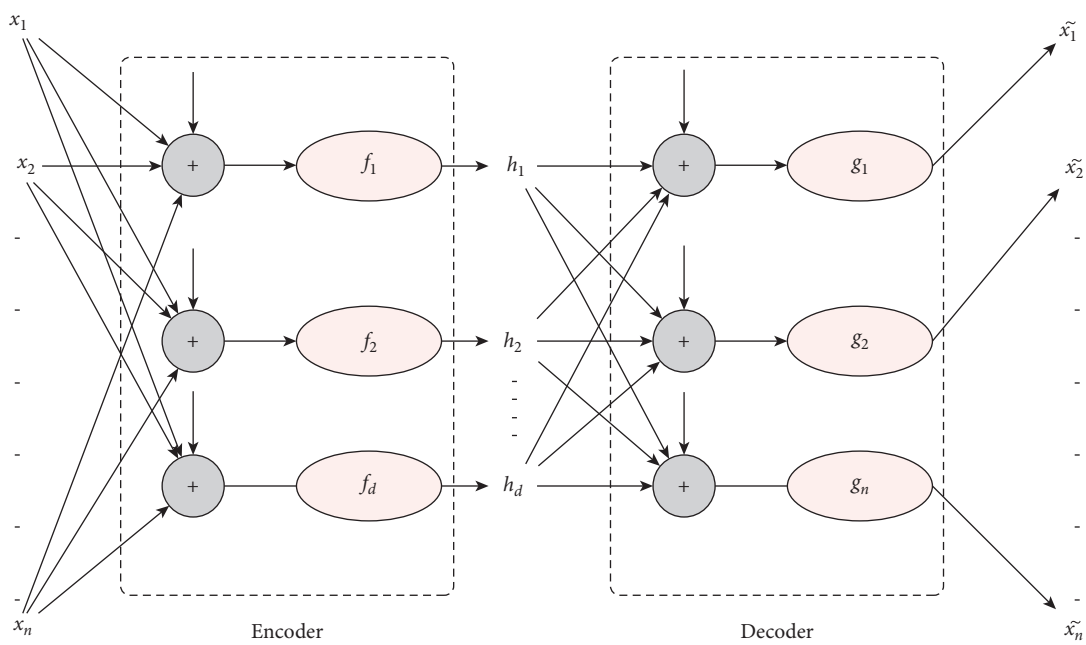


Figure 1: Autoencoder.

$$J(x, g(f(x))) + \lambda\Omega(h). \tag{4}$$

The proposed VFA methodology assumes some unknown distribution on the data to determine the distribution parameters. More specifically, let the dataset $X = \{x^{(i)}\}_{i=1}^{N}$, consisting of N iid samples. Each sample $x^{(i)}$ comes from a random process of an unobservable random variable h which comes from some prior distribution $p_\theta^*(h)$ so that from this distribution, a sample $h^{(i)}$ is obtained, respectively, and a sample $x^{(i)}$ is obtained from the bounded distribution $p_\theta^*(x|h)$.

The process of giving birth to the samples comes from its separate latent variable, which it does not share with any other sample, i.e., there are no global latent variables. Based on the above hypothesis, the goal of the proposed system is to determine $p_\theta^*(x|h)$. Because the random variables and the distribution parameters are unknown, according to Bayes theorem, the requested probability is the following:

$$p_{\theta^*}(h|x) = \frac{p_{\theta^*}(x|h)p(h)}{p_{\theta^*}(x)},$$
$$p_{\theta^*}(x) = \int p_{\theta^*}(x|h)p_{\theta^*}(h)\mathrm{d}h. \tag{5}$$

According to the above, the requested posterior is approached through a family of distributions. It is calculated based on the Kullback–Leibler divergence metric estimate, which quantifies the similarity between different distributions. Using the product of the logarithm to the common probability $p(h, x)$ of the above equation, a parametric solution can take the following form:

$$J(\lambda) = \sum_{i=1}^{N} J_i(\lambda) = \sum_{i=1}^{N} \mathbb{E}_{q_\lambda(h \mid x_i)}\{\log(p(h_i|x_i))\} - KL\{q_\lambda(h|x_i)p(h)\}. \tag{6}$$

Expressing the maximization problem as a minimization problem, the loss function of the proposed system can be described:

$$J(X,\theta,\phi)=\sum_{i=1}^{N}J_i(x_i,\theta,\phi),\ J_i(x_i,\theta,\phi)=-\mathbb{E}_{h_i\sim q_\theta(h_i\mid x_i)}\left\{\log\left(p_\phi(x_i|h_i)\right)\right\}+KL\{q_\theta(h_i|x_i)p(h_i)\}. \tag{7}$$

An issue that arises is the sampling process so that this selected version of the sample is as close as possible to the original. On the other hand, it is also required that the sampling requires computational time to be used in real applications. Our proposal for smoothing purposes is for selection to take place at a particular time for each new scale parameter and the required sample number. With this sampling policy, we avoid the repetition of a computationally demanding operation several times in each step since the samples are reused for different values of x. On the other hand, we guarantee the consistency of the property of triangular inequality that satisfies every norm, which can be applied as follows:

$$\begin{aligned}\widehat{g}(x)&=\frac{1}{n}\sum_{i=1}^{n}J(x-v_i),\quad v_i\sim \mathrm{N}\left(0,\sigma_k^2 I\right),\\ \widehat{g}(y)&=\frac{1}{n}\sum_{i=1}^{n}J(x-v_i),\quad v_i\sim \mathrm{N}\left(0,\sigma_k^2 I\right).\end{aligned} \tag{8}$$

The difference between the two equations is

$$\begin{aligned}\widehat{g}(y)-\widehat{g}(x)&=\frac{1}{n}\left(\sum_{i=1}^{n}J(y-v_i)-\sum_{i=1}^{n}J\left(x-v_j\right)\right)\\ &=\frac{1}{n}\left(\sum_{i=1}^{n}J(y-v_i)-J(x-v_i)\right),\quad \left(i=j\Rightarrow v_i=v_j\right),\end{aligned} \tag{9}$$

so it applies to the norm:

$$\begin{aligned}\|\widehat{g}(y)-\widehat{g}(x)\|&=\frac{1}{n}\sum_{i=1}^{n}J(y-v_i)-J(x-v_i)\\ &\le\frac{1}{n}\sum_{i=1}^{n}J(y-v_i)-J(x-v_i)\\ &\le\frac{1}{n}\sum_{i=1}^{n}L\|y-x\|\\ \frac{\|\widehat{g}(y)-\widehat{g}(x)\|}{\|y-x\|}&\le L.\end{aligned} \tag{10}$$

It is evident that if we selected new samples for each point, the algorithm might have failed to find the total minimum, primarily if it used first-order methods like slope descent, as the function is not smooth. This case substantially limits the acceptable cost functions that we can consider for optimization, as it requires the values of the function to be constrained. So, essentially, for a given required approach accuracy, the probability of adhering to it improves exponentially by increasing the number of samples.

Accordingly, given that uncertainty is directly related to the number of data samples, the amount of data about a state expresses the complete possible information. So, reducing the uncertainty since we have similarities between different distributions indicates an equal gain in the amount of data. The degree of similarity, in this case, expresses the degree of proximity of an element of $p_\theta^*(h)$ compared to the original elements of the bounded distribution $p_\theta^*(x|h)$. This interpretation is used to extract an abstract representation from a dataset, taking advantage of the proximity between different amounts of data. Furthermore, the above interpretation is used in the vague control. The degrees of similarity between the current and the reference situations in the rules form the basis for the interpolation mechanism between the conclusions.

Fuzzy entropy was used as a measure of the ambiguity of the whole, which results from the inherent ambiguity and vagueness of the whole itself due to the inability to separate the whole from its complement, that is, the abnormal elements from the normal. In this sense, the measures that assess uncertainty in the context of fuzzy set theory, namely, the entropy measures and the ambiguity indices, were adopted to implement a fully functional and realistic system for detecting anomalies in machine vision systems. The fuzzy entropy equation used is shown below:

$$E_{LT}^{FS}(\overline{A})=-\frac{1}{n}\sum_{i=1}^{n}\left[\mu_A(x)\log_2\mu_{\overline{A}}(x)+\left(1-\mu_{\overline{A}}(x)\right)\log_2\left(1-\mu_A(x)\right)\right]. \tag{11}$$

To measure the distance between normal and abnormal cases, the ambiguity index was used using the Euclidean distance:

$$f(\overline{\mathring{A}})=\left(\sum_{i=1}^{n}\left(\mu_{\overline{A}}(x)-\mu_{\widetilde{Ac}}(x)\right)^2\right)^{1/2}. \tag{12}$$

Thus, the final shape of the proposed architecture takes an intermediate level at which fuzzy logic is applied to the investigation of anomalies. The proposed architecture is schematically presented in Figure 2.

4. Scenarios

The reliability of steel industry applications is an important growth factor in the shipping industry, aviation, defense systems development, renewable energy sources, etc. A typical example is the durability and accuracy of the operation of gearboxes, where the reliability of the assembly ensures their smooth and long-lasting operation. For a gearbox to work correctly, it must be precisely aligned with the power units, axles (e.g., driveshaft), and other components (e.g., differential) involved. When the other component and the gearbox are not connected properly, the gearbox is not aligned. Poor alignment puts a lot of pressure on the gearbox parts, such as the axles and the coupling, and can deplete the device enough to cause severe wear and even drive failure. When misaligned, one end of a gear can crack or open earlier than it should, and similar damage can occur to bearings.

Poor alignment can occur due to static factors such as manufacturing defects or user error. Dynamic causes include heavy loads stretching the gearbox components and thermal expansion. Also, other factors can cause poor alignments, such as tilt error or oscillation and centrifugal forces. A particular cause that can create misalignment is the improper configuration of SCADA control systems that control the production process through cyberattack.

During regular operation, the control unit operator monitors the standard operation variables of the assembly system on the production line provided by the corresponding sensor. Abnormal behavior occurs when some assembly parameters are not within the normal range. During assembly, a laser shaft alignment system can measure the misalignment of the gears and rotate them to take the correct position. This process is activated after a specific notification of the control system. A cyberattack could deactivate if the cybercriminals could modify the assembly parameters outside of the normal operating range. The proposed mechanical vision system augmented by the VFA algorithm is used to deal with this type of cyberattack, not connected to the production line system operating autonomously as an independent security mechanism.

For the implementation of the experiments, a dataset including snapshots with the operating condition of a component was used, where the primary anomaly is related to a specific type of error related to the alignment of the gearbox gears.

To address the problem of image matching, a heuristic algorithm was used to calculate the digital variation and then apply methods to repeat and optimize these values while calculating the fuzzy sample entropy and Euclidean fuzzy similarity. This method essentially captures pixel layout and smoothness, which minimizes pixel mismatches. Specifically, the calculation of the digital variation is done by minimizing an energy function $E(d)$ as follows:

$$D = \arg\min\left(E_{d}\left(d_{p}\right)\right). \tag{13}$$

The energy function consists of two terms. The first refers to the data and measures how well the variation function d matches the pair of images. The second term refers to the assumptions made by the algorithm:

$$E(d) = E_{\text{data}}(d) + \lambda E_{\text{smooth}}(d). \tag{14}$$

The term E_{data} is equal to the sum of each pixel of the matching costs C of the disparity space image table:

$$E_{\text{data}}(d) = \sum_{(x,y)} C(x, y, d(x, y)). \tag{15}$$

The term E_{smooth} is equal to the sum of the depreciation of the variation differences between adjacent pixels:

$$E_{\text{smooth}}(d) = \sum_{(x,y)} p(d(x, y) - d(x+1, y)) + p(d(x, y) - d(x, y+1)), \tag{16}$$

where x is the scan column and y is the scan bar. The variable p is a function of the difference of the digital variant, genuinely increasing. The term $E_{\text{smooth}}(d)$ can be transformed to accommodate volume differences. This has the effect of reducing the smoothness of the image when the intensity gradient is high. The term λ is the relative weight of the normality term, and its value depends on the calculation method of the correlation cost.

Also, for the calculation of the correlation cost between two pixels using linear interpolations in the neighboring pixels, we used a parametric method which is more efficient and less sensitive to the effect of image signal sampling in case the brightness of the pixels changes abruptly, for example, in-depth discontinuities and repetitive patterns. The calculation was based on the following function:

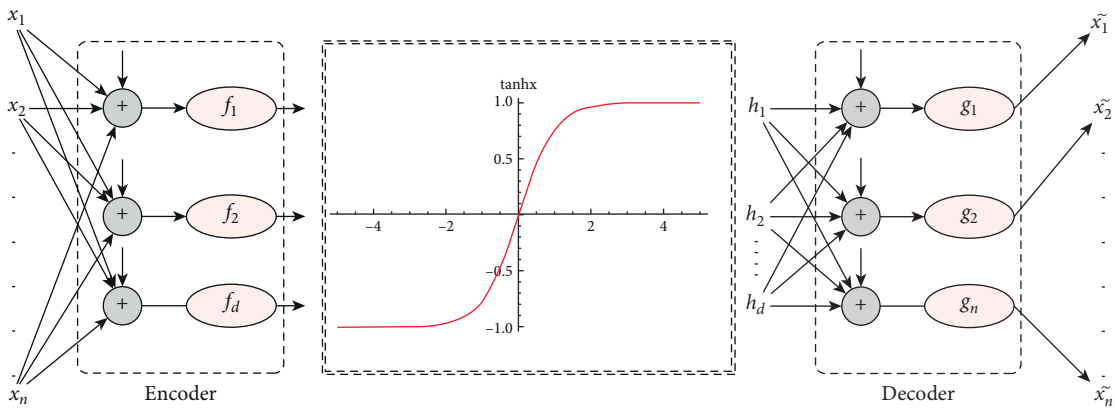


Figure 2: Variational fuzzy autoencoder.

$$
\begin{aligned}
C_{BT}(p,d) &= \min(A,B),\\
A &= \max\begin{pmatrix} 0, I_L(p) - I_R^{\max}(p-d), \\ I_R^{\min}(p-d) - I_L(p) \end{pmatrix},\\
B &= \max\begin{pmatrix} 0, I_R(p-d) - I_L^{\max}(p), \\ I_L^{\min}(p) - I_R(p-d) \end{pmatrix},\\
I^{\min}(p) &= \min\left(I^{-}(p), I(p), I^{+}(p)\right),\\
I^{\max}(p) &= \max\left(I^{-}(p), I(p), I^{+}(p)\right),\\
I^{-}(p) &= \frac{\left(I\left(p-[10]^T\right)+I(p)\right)}{2},\\
I^{+}(p) &= \frac{\left(I\left(p+[10]^T\right)+I(p)\right)}{2}.
\end{aligned}
\tag{17}
$$

Finally, we used the normalized correlation coefficient, which is the normalized expression of the variability of the reference windows and the search of the contrasted images. This coefficient is calculated using the average and the standard deviation of the values of the intensity of the brightness in the window. In this work, the correlation coefficient remains unchanged in uniform and linear changes of the brightness and contrast of the window:

$$
C_{NCC}(p,d) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\left(I_L(p)-\overline{I}_L\right)\left(I_R(p-d)-\overline{I}_R\right)\right)}{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(I_L(p)-\overline{I}_L\right)^2\sum_{i=1}^{n}\sum_{j=1}^{m}\left(I_R(p-d)-\overline{I}_R\right)^2}}.
\tag{18}
$$

Figure 3 shows a schematic representation of the relative entropy differentiation in the applied sample.

The concept of entropy is mainly based on the difficulty of distinguishing between a set and its complement, so the less the set differs from its complement, the vaguer it is. Therefore, there is a specific reason for the percentage of anomaly that characterizes it. In this sense, each probabilistic set is generated by randomizing the degree of participation of each element of its definition field separately. For this purpose, a probabilistic space is introduced and a random variable is assigned to each element with values between the space of the measure of similarity of the factors under consideration [7, 22, 23].

Figure 4 presents the methodology of accurate geometric determination of the fuzzy anomaly of the samples, utilizing the measurement of their Euclidean vague similarity [24]. The distance between two fuzzy sets is defined to be the regular Euclidean distance between the two corresponding vectors.

The general differences that can identify local or global anomalies were also investigated based on the proposed architecture and the characteristics of the data under consideration, as shown in Figure 5.

Finally, Table 1 presents typical snapshots of the process of using the VFA algorithm and the success rates we achieved.

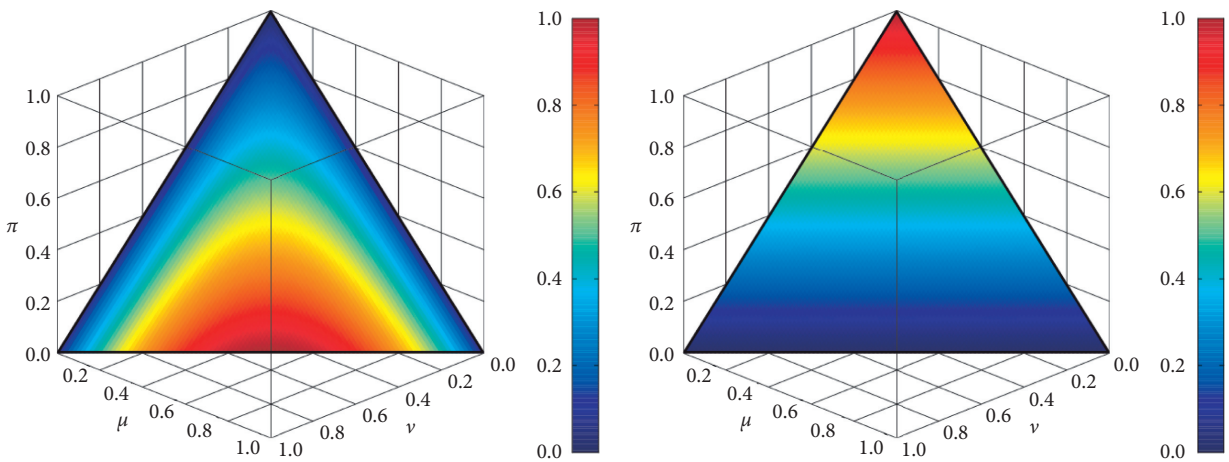


Figure 3: Fuzzy entropy comparison.

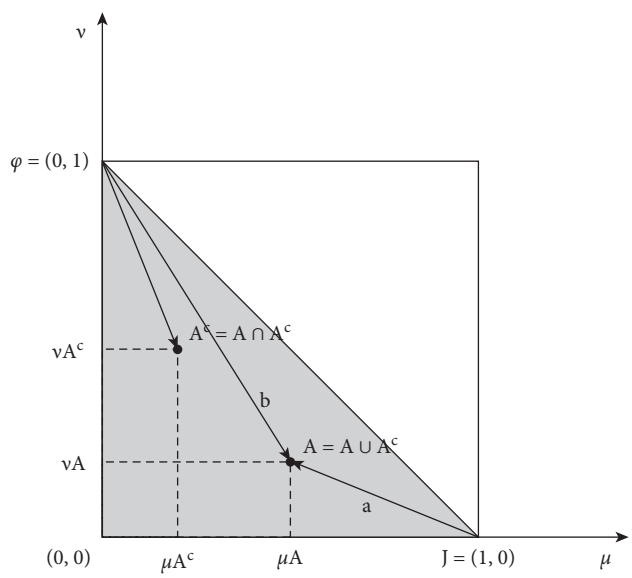


Figure 4: Geometric similarity using fuzzy Euclidean distance.

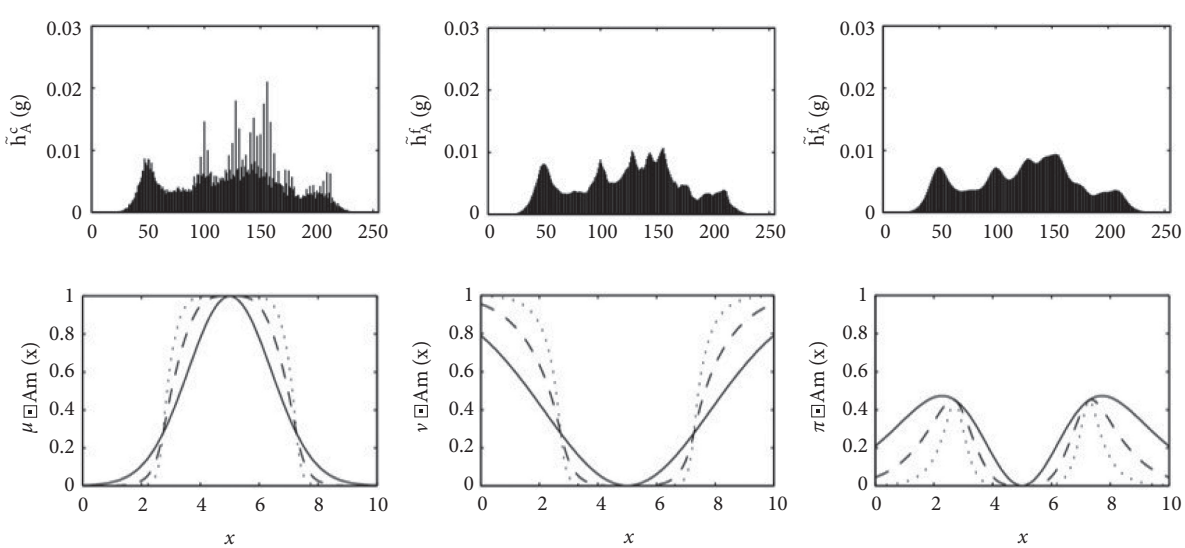


Figure 5: Similarity measure using image features.

Table 1: Performance evaluation of the proposed algorithm.

	Precision	Recall	*F*-measure	Anomaly score
VFA_Instance_1	88.9	89.0	89.0	0.12
VFA_Instance_2	86.7	87.1	86.8	0.18
VFA_Instance_3	92.1	92.0	92.1	0.08
VFA_Instance_4	89.9	89.9	89.9	0.11
VFA_Instance_5	90.3	90.2	90.2	0.10
VFA_Instance_6	84.5	84.7	84.8	0.22
VFA_Instance_7	88.3	88.3	88.2	0.13
VFA_Instance_8	94.2	94.2	94.0	0.06
VFA_Instance_9	95.1	95.0	95.1	0.05
VFA_Instance_10	93.4	93.4	93.4	0.06
Average score	90.34	90.38	90.35	0.11

When the increase or decrease of the anomaly is not abrupt, the extra time points included in the control intervals have a relatively high probability of redefinition with higher accuracy. However, if we have a sharp increase or decrease in probability, the intervals may include times when the activity may show less categorization accuracy. This observation is related to the inherent noise in the dataset, resulting in the fluctuation of the algorithm tendency. In general, however, the finding is that the proposed system can evaluate and categorize with significant accuracy anomalies in a highly complex environment.

The computational complexity is linearly dependent on the sequence length, which means inference is fast and scalable to very large files. All experiments were performed in the Google Colab environment using a GPU processor. To avoid high overhead and achieve timely model convergence, it was necessary to train the proposed system using a relatively small but at the same time satisfactory batch size. Due to the overuse of memory, the heuristic algorithm was used to calculate the digital variation and optimize these values while calculating the fuzzy sample entropy and Euclidean fuzzy similarity. It turns out that these methods are suited to perform the computation of extremely complex processes.

5. Conclusions

The detection and timely evaluation of abnormalities in machine vision systems allow the industrial sector to make innovative leaps. This logic is in line with Industry 4.0 and the vision for innovative approaches in modern industry. In this work, we presented a machine vision system that contributes to the efficiency of the new ecosystem of Industry 4.0. It is an intelligent system for identifying anomalies in advanced gearbox assembly systems. Specifically, we presented the VFA methodology whereby using fuzzy entropy and Euclidean fuzzy similarity measurement, we maximized the possibility of using nonlinear transformation through deterministic functions, thus creating an entirely realistic and highly reliable machine vision detection system.

The evaluation of the methods was carried out in a highly complex cybersecurity scenario, where cybercriminals could modify the assembly parameters of the production mechanisms. This fact is not perceived by the other sensors connected to the production system. Utilizing the most advanced machine vision techniques and fuzzy logic methodologies, the proposed method has achieved very high success rates, creating serious expectations for additional cybersecurity applications.

Significant progress could be made in hardening the system with methods of intuitive fuzzy logic. In addition to similarity measures between samples, dissimilarity measures could also be measured, thus making the system even more sensitive and realistic. Also, an extension of the proposed method could study the system's operation in an inversely proportional manner, where two VFAs would operate as opposed to the parallel detection of anomalies.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] N. Jazdi, "Cyber physical systems in the context of Industry 4.0," in *Proceeding of the 2014 IEEE International Conference on Automation, Quality and Testing, Robotics*, pp. 1–4, Cluj-Napoca, Romania, May 2014.

[2] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for Industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, pp. 18–23, 2015.

[3] J. Goh, S. Adepu, M. Tan, and Z. S. Lee, "Anomaly detection in cyber physical systems using recurrent neural networks," in *Proceedings of the 2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*, pp. 140–145, Singapore, January 2017.

[4] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.

[5] H. Golnabi and A. Asadpour, "Design and application of industrial machine vision systems," *Robotics and Computer-Integrated Manufacturing*, vol. 23, no. 6, pp. 630–637, 2007.

[6] Y. Hou, M. Zhang, and L. Yang, "Fault detection of actuators via extended state observer," in *Proceding of the 2019 CAA Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS)*, pp. 651–655, Xiamen, China, July 2019.

[7] B. Lu, D. Xu, and B. Huang, "Deep-learning-based anomaly detection for lace defect inspection employing videos in production line," *Advanced Engineering Informatics*, vol. 51, Article ID 101471, 2022.

[8] R. Bogue, "Cloud robotics: a review of technologies, developments and applications," *Industrial Robot: International Journal*, vol. 44, no. 1, pp. 1–5, 2017.

[9] K. Stoddart, "UK cyber security and critical national infrastructure protection," *International Affairs*, vol. 92, no. 5, pp. 1079–1105, 2016.

[10] C. Davies, C. Holcombe, J. Skillman et al., "Protocol for a mixed-method study to inform the feasibility of undertaking a large-scale multicentre study comparing the clinical and patient-reported outcomes of oncoplastic breast conservation

as an alternative to mastectomy with or without immediate breast reconstruction in women unsuitable for standard breast-conserving surgery (the ANTHEM Feasibility Study)," *BMJ Open*, vol. 11, no. 4, Article ID e046622, 2021.

[11] N. Elmrabit, F. Zhou, F. Li, and H. Zhou, "Evaluation of machine learning algorithms for anomaly detection," in *In Proceedings of the 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1–8, Dublin, Ireland, June. 2020.

[12] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, "Cyber threats to industrial IoT: a survey on attacks and countermeasures," *IoT*, vol. 2, no. 1, pp. 163–186, 2021.

[13] A. Cuzzocrea, "Big data lakes: models, frameworks, and techniques," in *In Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1–4, Jeju Island, Korea (South), January. 2021.

[14] D.-M. Tsai and P.-H. Jen, "Autoencoder-based anomaly detection for surface defect inspection," *Advanced Engineering Informatics*, vol. 48, Article ID 101272, 2021.

[15] F. Milkovic, B. Filipovic, M. Subasic, T. Petkovic, S. Loncaric, and M. Budimir, "Ultrasound anomaly detection based on variational autoencoders," in *Proceedings of the 2021 12th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 225–229, Zagreb, Croatia, September. 2021.

[16] F. Wang, H. Wang, and L. Xue, "Research on data security in big data cloud computing environment," in *Proceedings of the 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1446–1450, Chongqing, China, March. 2021.

[17] D. Zimmerer, S. A. A. Kohl, J. Petersen, F. Isensee, and K. H. Maier-Hein, "Context-encoding variational autoencoder for unsupervised anomaly detection," 2018, http://arxiv.org/abs/1812.05941.

[18] M. Carletti, C. Masiero, A. Beghi, and G. A. Susto, "Explainable machine learning in industry 4.0: evaluating feature importance in anomaly detection to enable Root cause Analysis," in *Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 21–26, Bari, Italy, October. 2019.

[19] M. Banifakhr and M. T. Sadeghi, "Anomaly detection in traffic trajectories using a combination of fuzzy deep convolutional and autoencoder networks," *Comput. Knowl. Eng.*, vol. 31, December 2020.

[20] Z. Gu and Y. Yang, "Detecting malicious model updates from federated learning on conditional variational autoencoder," in *Proceedings of the 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 671–680, Portland, OR, USA, May 2021.

[21] X. Xu, J. Li, Y. Yang, and F. Shen, "Toward effective intrusion detection using log-cosh conditional variational autoencoder," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6187–6196, 2021.

[22] M. A. Zapletina and D. V. Zhukov, "The review of cellular automata algorithms for placement and routing problems," in *In Proceedings of the 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, pp. 2771–2776, St. Petersburg, Russia, January. 2021.

[23] R. L. Ulloa-Cazarez, N. Garcia-Diaz, and L. Soriano-Equigua, "Multi-layer adaptive fuzzy inference system for predicting student performance in online higher education," *IEEE Latin America Transactions*, vol. 19, no. 1, pp. 98–106, 2021.

[24] J. Gawlikowski, C. R. N. Tassi, M. Ali et al., "A survey of uncertainty in deep neural networks," 2021, http://arxiv.org/abs/2107.03342.

Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3687332, 7 pages
https://doi.org/10.1155/2022/3687332

Research Article

Certificateless Hybrid Signcryption by a Novel Protocol Applied to Internet of Things

Wenzhan Zhang,[1] Yanhui Zhang,[2] Chong Guo,[3] Qi An,[2] Yuming Guo,[2] Ximing Liu,[2] Shijun Zhang,[2] and Junjia Huang[2]

[1]*University of Science and Technology of China, Hefei 230026, China*
[2]*Beijing Chuangan BDsecurity Technology Co., Ltd., Beijing 100160, China*
[3]*Internat of Things Security and Trusted Technology Co., Ltd., Xiamen 361106, China*

Correspondence should be addressed to Chong Guo; ccieguo@126.com and Yuming Guo; gym860118@hotmail.com

Received 10 January 2022; Revised 19 January 2022; Accepted 20 January 2022; Published 26 February 2022

Academic Editor: Konstantinos Demertzis

The rapid development of the Internet of Things (IoT) has accelerated the integration of science and technology with life, enabling the public to start enjoying the convenience brought by intelligent living. However, there are multiple resource-constrained sensing devices in IoT, which are always facing various external or internal attacks, making it difficult to ensure the secure transmission of sensitive data in IoT. Therefore, to address the problem of data transmission in resource-constrained devices in IoT, we propose a new certificateless hybrid signcryption scheme for IoT. It is a novel scheme that satisfies confidentiality and unforgeability, showing higher computational efficiency and lower overhead of transmission. To prove that it satisfies the efficent transmission of IoT, we conduct simulation experiments, and the experimental results show that our proposed scheme has higher efficiency than the existing schemes.

1. Introduction

The rapid development of IoT has accelerated the integration of science and technology with life, enabling the public to start enjoying the convenience brought by intelligent life. For example, digitalization brought by the smart city has solved the problem of "people" having difficulty in doing their work, and the automation brought by the smart home has reduced the public's household work. The convenience brought by IoT is far more than that. Based on the convenience brought by the Internet of Things, the scale of IoT devices is also a gradually expanding trend. It is applied in all walks of life, and the security of IoT devices is gradually coming to the fore.

A large number of legacy devices are undergoing digital transformation; however, few of them are equipped with the appropriate protection capabilities, making the overall security of the IoT less reliable. As a result, cloud-only protection for the IoT is far from adequate for its security. As the variety of IoT devices grows, providing attackers with a wider range of attack entry points, IoT will face even greater risk challenges, and the importance of its security protection cannot be ignored even more. IoT terminal, because of performance and cost limitations, hardware, and software security protection, cannot be integrated so that it is fully exposed to the network. Active protection is difficult to achieve because of energy-saving and other requirements of the limitations, and it cannot be reported immediately to complete the operational status of end-to-end protection and monitoring, making it vulnerable to attacks. There is a phenomenon that a large number of IoT terminals are "working with illness." In addressing IoT security, security protection can be provided from the perception layer, transmission layer, and application layer.

The perception layer has various types of devices, which are secured mainly by encryption and authentication to prevent attackers from illegally accessing tags and nodes. Transport layer protection uses strict authentication

mechanisms between nodes and security protocols that are closely related to keys. Application layer security protection focuses on securing database access control techniques. Among the many security risks of IoT, the great security risk is the leakage of users' privacy. Hence, when considering protection, the user's privacy is first secured. The current solution is mainly through encryption, signature, and authorization authentication.

Providing information security services is achieved through cryptosystems in cryptography, where cryptosystems ensure the secure transmission of messages between the communicating parties in an untrustworthy environment. Confidentiality and authentication are important parts of cryptosystems to provide information security services. Confidentiality refers to the mapping of readable plaintext transformations to unreadable ciphertext using encryption. Authentication prevents the communicator from denying previous actions by signing and verifying the identity information of the signer. With the rapid development of network information, the previous encryption technology cannot meet the security needs of IoT, for example, when the ciphertext is tampered with during transmission, the receiver still cannot receive the correct message even after decrypting it using the correct key. Of course, the authentication of the sender is also important during the transmission of the message. Hence, the use of encryption or signature alone is not enough to meet the current needs of IoT security, and a combination of signature and encryption is needed.

The traditional method of providing encryption and signature is "sign first, encrypt later," however, the computation and communication costs are the sum of the two, which is inefficient. The signcryption scheme simplifies the encryption and signature scheme, reducing the cost of computation and communication while improving the efficiency of signature and encryption. IoT devices usually have limited computing power and cannot afford complex calculations. Hence, signcryption technology can effectively ensure the secure transmission of data while not requiring high computing power.

The traditional encryption technology is usually based on public key infrastructure (PKI) to realize the encrypted transmission of data, and the public key is stored in the public key's directory by the certificate authority. Because of the huge number of IoT devices, using PKI to manage the public key management and authentication of IoT devices needs to assume huge computing and storage capacity, however, the hardware and software resources of IoT devices are not enough to support the resource consumption of PKI encryption system. To ensure the secure transmission of IoT data, a certificateless hybrid signcryption mechanism is proposed to reduce the storage, issuance, and verification costs of public key certificates. It improves the previous key escrow problem and the management problem of certificates in traditional public key infrastructure. The main idea of certificateless hybrid signcryption is that the device itself calculates its own public key, and the private key is jointly generated by the key generation center and the device itself, without binding the identity of the device to the public key, which changes the previous problem of public key escrow.

However, the certificateless signcryption scheme also brings some new problems. The frequent operation of bilinear pairing will consume a lot of hardware and software resources, and the devices with limited IoT resources are not enough to support the above operation. Also, the current schemes are not sufficient to meet the security requirements of IoT device data transmission. Therefore, this paper proposes a new certificateless hybrid signcryption scheme for IoT, and the contributions of this paper are as follows:

(1) We propose a new certificateless hybrid signcryption scheme

(2) We prove that our scheme meets confidentiality and unforgeability

(3) We have compared the efficiency with other schemes and found that our scheme has higher efficiency

The paper is organized as follows: Section 2 focuses on the current state of research on IoT and the development of a certificateless hybrid signcryption scheme. Section 3 focuses on the preparatory knowledge, including the basics of cryptographic theory, such as random oracle machine provable security theory, discrete logarithm, bilinear mapping, etc. Section 4 describes the details of our proposed scheme. Section 5 describes the security analysis of the certificateless hybrid signatures and proves it. Section 6 compares other schemes with the scheme proposed in this paper for efficiency analysis, and finally, Section 7 concludes the above certificateless hybrid signcryption scheme.

2. Related Work

In 1997, Zheng introduced the concept of the signcryption mechanism. It breaks the traditional way of encryption followed by signature, and it adopts the way of simultaneous encryption and signature. It reduces a large number of calculations, and thus, it greatly improves the efficiency of communication, enabling the secure transmission of data [1]. In 2003, AL-Riyami and Paterson proposed certificateless cryptography, which was proposed to solve the problem of key escrow in ID-PKC. The private key in certificateless cryptography is a combination of the user's own private key and part of the private key generated by KGC. It no longer uses a certificate to bind the identity, thus solving the problem of key escrow. However, the ensuing public key replacement attacks still threaten information security [2].

In the early days, there was no formal security definition for hybrid encryption, which was only on the application requirements. The formal security definition was not formally proposed until 2004 when the formal security definition of KEM-DEM structure based on hybrid encryption was formally proposed by Cramer et al. It uses a combination of secret key encapsulation mechanism and data encapsulation techniques, thus allowing hybrid ciphers to

solve the IND-CCA security problem, and hybrid ciphers are also an actual public key cryptosystem [3].

In 2005, Dent proposed the concept of hybrid signcryption cipher, which is a combination of the advantages of symmetric and public key ciphers, i.e., the hybrid signcryption uses the symmetric key to encrypt the plaintext and public key to encrypt the key needed to be used in the management of the information symmetric cipher because the two encryptions are done separately. Hence, they do not interfere with each other and are independent of each other, thus improving the reliability and security of the encryption [4].

The concept of certificateless signcryption was first introduced by Barbosa et al. in 2008. It is a cryptographic technique that provides certificateless encryption and signature, thus triggering a frenzied pursuit of certificateless signcryption in the cryptographic community to the extent that certificateless signcryption became one of the popular research projects in cryptography. However, they gave schemes whose process was too complicated, causing problems, such as it being too complex, inefficient, and difficult to handle security issues. Subsequently, Aranha et al. [5], Wu et al. [6], and Selvi et al. [7] also improved the scheme one after another, however, all of them had more or fewer problems. Aranha et al. did not have a security-proof process, and Wu et al. did not implement the unforgeable nature.

In 2010, Xie et al. [8] proposed signcryption schemes with identity-based and certificateless public key encryption, which requires only two bilinear pairwise operations for its signcryption process. It greatly reduces the computation time. However, its verified dissatisfaction meets the unforgeability. In the same year, Li et al. [9] also proposed a certificateless signcryption scheme, which claimed to be a provably secure scheme requiring only two bilinear pairs of operations, and it was later verified to be insecure. Liu et al. [10] also proposed a certificateless signcryption scheme, which was based on the standard model and required five bilinear pairs of operations, and it was later noted to be insecure.

In 2011, Sun et al. [11] proposed a certificateless signcryption scheme that uses only one bilinear pair operation, which was later also pointed out to have shortcomings. In the same year, Wenhao Liu et al. [12] also proposed a very efficient certificateless signcryption scheme. It was also found to have some insecurity problems. Also, in 2012, Singh [13] proposed a certificateless hybrid signcryption scheme based on identity security authentication.

In 2013, Swapna et al. [14] proposed an elliptic curve-based authentication sign-off scheme in a way that it is a multiagent that can perform multiple sign-off processes simultaneously. In the same year, Li et al. [15] also proposed a certificateless hybrid signicryption scheme, which proved the unforgeability and confidentiality of their scheme. In 2014, Lai [16] proposed a multiparty hybrid signing scheme suitable for use in firewalls and with multiple participants, which is implemented by signcryption and multiparty encryption techniques, and using this scheme can significantly improve computation and transmission efficiency while ensuring confidentiality and nonrepudiation. In 2015, Zhang et al. [17] proposed a certificateless aggregated signcryption scheme, which can guarantee confidentiality and reduce the complexity and overhead of transmission at the same time.

In 2016, Zhou et al. [18] proposed a publicly verifiable certificateless hybrid signcryption scheme that can guarantee the security of transmission despite certain information leakage, in line with the properties of public verifiability, confidentiality, unforgeability, and resistance to information leakage. In 2017, Xu et al. [19] proposed a bilinear pair-based certificateless hybrid signcryption scheme that combines certificateless and hybrid signcryption mechanisms with adaptability, unforgeability, confidentiality, and high-security performance and computational efficiency, and it is more suitable for use when bandwidth receives limitations. In 2019, Yu et al. [20] proposed an improved certificateless hybrid signcryption scheme with an efficient cipher scheme for cover Sun, which eliminates the dross, absorbs the essence, and achieves nonrepudiation, as well as public verification based on the efficiency of the original scheme, which can maintain efficient operation when resisting attacks.

From the analysis of the above research, the research on certificateless hybrid signcryption has never stopped, and the research on certificateless hybrid sigcryption has been gradually improved and perfected. This paper is a novel certificateless hybrid signcryption scheme based on the previous ones, which satisfies confidentiality and unforgeability, showing high computational efficiency and low overhead of transmission.

3. Preliminary

3.1. Basic Mathematical Concepts.

(1) Euler function: for the positive integer n, Euler function $\phi(n)$ is the number of positive integers less than or equal to n that are mutually prime with n

(2) Euler's theorem: if n, a is a positive integer and n, a are mutually prime, then $a^{\phi(n)} \equiv 1 \pmod{n}$

(3) The original root: if n, a are positive integers and n, a are mutually prime, such that $a^d \equiv 1 \pmod{n}$, if $\delta(n, a)$ denotes the smallest positive integer d that makes the equation hold, at which point if $\delta(n, a) = \phi(n)$, then we call a as the original root of mod n

4. The Discrete Logarithm Puzzles

If, for an integer b and a prime number p of an original root, a unique index i can be found such that $b = a^i \, (mod\ p)$, where $0 \le i \le p - 1$ holds, then the exponent i is called b of a as the base of the modulus p of the discrete logarithm.

4.1. Bilinear Pairs. Let a large prime $q < 2^k$, where k denotes a security parameter. Let G_1 be an additive cyclic group of order q, G_2 be a multiplicative cyclic group of order q, P be

the generator of G_1, and $\hat{e} = G_1 \times G_1 \longrightarrow G_2$ be a bilinear map with the following three properties:

(1) Bilinear: $\forall a, b \in Z_q^*$ and $\hat{e}(aP, bP) = \hat{e}(P, P)^{ab}$

(2) Nondegradability: $\hat{e}(P, P) \neq 1$

(3) Computability: $\forall P, Q \in G_1$, there exists an efficient algorithm to compute $\hat{e}(P, Q)$

4.2. Mathematical Difficulties.

(1) DLP (discrete logarithm problem) problem: given any $Q \in G_1$, compute $a \in Z_q^*$ such that $Q = aP$.

(2) CDH (computational Diffie–Hellman) problem: suppose $a, b \in_R Z_q{}^*$, known as P, P^a, P^b. Compute P^{ab}.

(3) BDH (bilinear Diffie–Hellman) problem: given any (aP, bP, cP), where $a, b, c \in Z_q^*$, calculate $e(P, P)^{abc}$.

(4) DBDH (decisional bilinear Diffie–Hellman) problem: For any unknown $a, b, c \in Z_q^*$, known $(P, aP, bP, cP) \in G_1$, and $z \in G_2$, whether $e(P, P)^{abc} = z$is decided. If so, O_{DBDH} returns 1, otherwise, O_{DBDH} returns 0.

5. The Proposed Scheme

This chapter gives a new certificateless hybrid signcryption scheme for IoT, and below are the 6 main modules of the scheme.

5.1. System Initialization. Select the additive cyclic group G_1 and the multiplicative cyclic group G_2 , where $|G_1| = |G_2| = q$, P is the generator of G_1. Meanwhile, KGC selects a bilinear pair e: $G_1 \times G_1 \longrightarrow G_2$, randomly choosing x_0 as the master key and computes $P_{pub} = x_0$P as the system public key. Three hash functions are selected, $H_1 = \{0, 1\}^* \longrightarrow G_1$,
$H_2 = \{0, 1\}^* \times \{0, 1\}^* \times G_1 \times G_1 \times G_1 \times G_1 \longrightarrow Z_q^*$, and $H_3 = G_2 \times G_1 \times G_1 \longrightarrow \{0, 1\}^*$.

5.2. User Key Generation. The user randomly selects x_i as the secret value and calculates $P_i = x_i P$ as the user's public key.

5.3. Generation of Partial Private Keys. The user sends itself ID to KGC, which calculates $Q_i = h_1(ID_i)$ and $d_i' = x_0 Q_i$. The private secure channel is then used to send d_i' sent to the user.

5.4. Generate All User Private Keys. The user receives the KGC sent d_i' after calculating a partial private key $D_i = x_i^{-1} d_i' = x_i^{-1} x_0 Q_i$. After that, the user combines the secret value generated by itself and the partial private key generated by KGC to generate the complete full private key $SK_i = (x_i, D_i)$.

5.5. Signcryption. During the signcryption and signcryption process, it is assumed that the sender's user ID is ID_A, the recipient's user ID is ID_B, and the message to be sent is m.

The known sender ID_A and recipient ID_B have completed the initialization of the key, and they know the system parameters, such as the public key and the system public key of both parties. The specific process is as follows:

(1) Randomly choose random values $r \in Z_q^*$, and calculate $R = rP$.

(2) Calculate $x = e(x_A Q_B, D_A)$, and $y = rP_B$.

(3) Calculate the session key $k = H_3(x, y, R)$, and simultaneously encrypt the data m. Perform symmetric encryption, and compute the ciphertext $c = Enc(k, m)$.

(4) Calculate $h = H_2(ID_A, ID_B, P_A, P_B, P_{pub}, R)$.

(5) Calculate $s = x_A h + r$. Generate a signed cipher C = (c, R, s) sent to the recipient ID_B.

5.6. Unsigncryption. The system public key is the P_{pub}. The ID_A is the identity of the sender, and the P_A is the public key of the sender. Also, the ID_B is the identity of the receiver and SK_B the private key of the receiver. C = (c, R, s) is the ciphertext. The unsigncryption process is as follows:

(1) Calculate $x = e(x_B Q_A, D_B)$ and $y = x_B R$

(2) Calculate the session key $K = H_2(x, y, R)$, simultaneously decrypt the ciphertext c, and calculate the plaintext $m = Dec(k, c)$

(3) Calculate $h = H_2(ID_A, ID_B, P_A, P_B, P_{pub}, R)$. Also, verify if the equation $sP = hP_A + R$ holds

If the validation equation holds, then receive the message m. If it does not hold, the message is dropped directly.

6. Correctness Analysis

In the scheme of this paper, the correctness analysis is in two parts as follows:

6.1. Symmetric Encryption. The first part proves that the session key for symmetric encryption between user A and user B is correct. The parameter x calculated by user A is as follows:

$$x = e(x_A Q_B, D_A) = e\left(x_A Q_B, x_A^{-1} d_A'\right) = e(Q_B, d_A') = e(Q_B, x_0 Q_A) = e(Q_A, d_B') = e\left(x_B Q_A, x_B^{-1} d_B'\right) = e(x_B Q_A, D_B). \quad (1)$$

After extrapolation, it can be found to be equal to parameter x calculated by user B.

The parameter y calculated by user B is as follows:

$$y = rP_B = rx_BP = x_BrP = x_BR. \quad (2)$$

After extrapolation, it can be found to be equal to parameter y calculated by user B, and R is a common parameter known to both user A and user B. Therefore, the session key computed by user A and user $B, K = H_3(x, y, R)$, is the same.

6.2. Authentication Process. The second part proves that the authentication process of user B to user A's signed secret message is correct. User A and user B calculate the message hash h for both $h = H_2(ID_A, ID_B, P_A, P_B, P_{pub}, R)$, where the identity of user A and user B and public keys are known to both parties. The system public key P_{pub} is public, and the parameter R is generated by user A, however, they are also attached to the ciphertext c and passed to user B. Therefore, the message hash values computed by user A and user B h are equal.

User B decides whether to accept the signed message by calculating whether equation $sP = hP_A + R$ or not. If the identity of user A is true, then user A computes the parameter s as $s = x_Ah + r$, and the authentication equation for user B equals

$$sP = (x_Ah + r)P = x_AhP + rP = hx_AP + rP = hP_A + R. \quad (3)$$

Verify that the equation holds. Since x_A is the private key of user A, only user A has it. Then, user A alone can compute its public key P_A corresponding to its public key with the correct parameter s. Hence, the correctness of user B's verification equation for user A is proven.

7. Security Analysis

7.1. Confidentiality Analysis. The scheme in this paper establishes a session using a public-private key encryption scheme with KGC, negotiating the session key and transmitting the signature during the session establishment process. The session key is computationally obtained by $K = H_3(x, y, R)$, where $x = e(x_AQ_B, D_A)$, $y = rP_B$, and $R = rP$. The attacker wants to compute to get the session key between user A and user B. He needs to compute to get x, y, R, where R is contained in the ciphertext, which is easily intercepted by the attacker. While the attacker does not know the private keys of the two users x_A and x_B, the computation of x is a BDH puzzle. Hence, it is not feasible for the attacker to compute the value of x. The attacker needs to know the random number r chosen by user A in the process of establishing the session, or the private key of user B to compute yx_B, and either computing r by R or P_B computing x_B, which are discrete logarithmic puzzles and computationally infeasible.

Hence, the attacker is computationally unable to learn the session key between user A and user B, and the communication between them is confidential.

KGC picks the system private key x_0 which is stored only in the KGC and is not transmitted over any channel, and the attacker is able to use it via the system public key P_{pub} to compute x_0 for the discrete logarithm puzzle, which is computationally infeasible. Correspondingly, the attacker passes the user's public key P_i to compute the user's private key x_i for the discrete logarithm puzzle, which is also unavailable, and hence, the private keys of the user and KGC are confidential.

7.2. Unforgestibility. Unforgestibility means that it is computationally infeasible for other noncluster nodes, masquerading as in-cluster nodes, to generate signature messages that pass verification.

If an attacker wants to forge a valid signcryption ciphertext by masquerading, the secret value x_i and the random value r have to be chosen, and the forged s is generated. However, because of the CDH problem, s cannot pass the verifying equation $sP = hP_A + R$, and KGC will not recognize this malicious node. Hence, the attacker cannot send the ciphertext by masquerading as a legitimate node.

If an attacker wants to replace the private key generated by the node, the user key generates the full private key SK_i. The data sources in the process of generating the complete private key are x_i and d_i. d_i is transmitted to the user by KGC under a secure channel, and x_i is stored within the user's own node and is not available to the forger. If nongroup members want to forge the identity of user A, they can only do so by capturing the public key P_i, which is transmitted to the user by P_i computing x_i. It is the discrete logarithm puzzle, and there is no effective algorithm for the discrete logarithm puzzle so far. Hence, the scheme in this paper is unforgeable.

7.3. Nonrepudiation. Nonrepudiation means that parties in message communication must add information containing their own unique and distinctive information at the time of message transmission data to prevent the denial of the act after the message transmission is completed.

A complete denial resistance mechanism usually consists of two parts: one for the signature part and one for the verification part. The secret key of the signature part is usually the secret key of the sender, which is the sender's own unique and unique information that only the sender of the message holds. It is also the premise and assumption of the denial resistance. The secret key of the verification part is usually the public key of the sender of the message so that the receiver of the message can verify the message.

Encrypting a message with the sender's private key has a four-part effect, which is as follows:

(1) Authentication is performed. If receiver B receives a message encrypted with sender A's private key, it can decrypt it with sender A's public key, and if the decryption is successful, receiver B can be sure that the received message is from sender A. It is because if

Table 1: Comparison of time complexity of various operations.

Computational	Time complexity
Scalar multiplication operation S	$1S \approx 29M$
The addition of points A	$1A \approx 0.11$ M
The bilinear pair operation P	$1P \approx 87M$
Exponential operation E	$1E \approx 21M$
Ordinary hashing operation h	Neglect

Table 2: Comparison of efficiency.

Options	Signcryption			Unsigncryption		
	Dot product operation	Exponential operation	Bilinear operation	Dot product operation	Exponential operation	Bilinear operation
Yu [21]	3	2	2	0	1	6
Jin[22]	3	3	1	3	3	3
Our scheme	3	0	1	3	0	1

Table 3: Comparison of time consumption.

Options	Computational complexity	Total time spent (M)	Times (s)
Yu [21]	3M + 3E + 8P	762	0.0512
Jin[22]	6M + 6 E + 4P	480	0.0322
Our solution	6M + 2P	180	0.0121

receiver B can decrypt the message with A's public key, it proves that the original message is encrypted with A's private key and only A knows his private key. Thus, sender A encrypts the message with his private key to make his own digital signature.

(2) Putting in a fake. An attacker cannot impersonate sender A. If attacker C impersonates A and sends a message, attacker C cannot encrypt the message with A's private key because attacker C does not have sender A's private key, and receiver B cannot decrypt it with sender A's public key. Hence, attacker C cannot impersonate sender A.

(3) Denial-proofness so that if a dispute arises between two parties, receiver B can produce the encrypted message and decrypt it with the public key of sender A, thus proving that the message came from A, since it was encrypted with A's secret key, which only A has.

(4) Prevent the message from being tampered with. If attacker C intercepts the confidential message during the message transmission, he can decrypt it using A's public key and change the message, however, he cannot achieve his goal. As attacker C does not know A's private key and cannot encrypt the message using A's private key, after attacker C sends the altered message to receiver B, B cannot decrypt the message using A's public key either, and B will not think that the message came from A.

In this design, when user B receives the signed ciphertext $C = (c, R, s)$ from user A, he will verify it, where c is the ciphertext R is the temporary parameter generated during communication, and s is the "digital signature" generated by user A.

The process is as follows:

Firstly, when user A sends a ciphertext by computing $s = x_A h + r$, where x_A is the private key of user A and r is a random value generated at each communication. When user B receives the message sent by user A, verify whether the equation $sP = hP_A + R$ holds, where P_A is the public key of user A held by user B.

It is known that $s = x_A h + r$. Bringing it into the verification equation yields the following:

$$sP = (x_A h + r)P = x_A hP + rP = hx_A P + rP = hP_A + R. \tag{4}$$

According to the formula of the user public key $P_i = x_i P$, it is known that only x_A can make the verification equation hold. As x_A is the private key of user A, only A knows it, and if a dispute arises between the two parties, user B can take out the encrypted message and decrypt it by user A's public key, thus proving that the message came from A, and user A cannot deny that it sent the signed message, thus achieving the nonrepudiation of the message.

8. Efficiency Analysis

Table 1 shows the time complexity of each operation, where P denotes the bilinear pair operation, S denotes the scalar multiplication operation on an elliptic curve, A denotes the addition operation on two elliptic curve points, E denotes the exponential operation, and all the above are being calculated and compared as a multiplication product of M.

From Table 2, it can be seen that relative to the existing literature, the scheme in this paper does not add a large computational burden to individual signature nodes based on the implementation of group signatures. Thus, the scheme in this paper has good applicability.

The number multiplication operation time on the elliptic curve on a 900KHZ sensor is approximately 2.6s, and considering the latest CortexA9 1.2 GHz microprocessor for smart terminals, the number multiplication operation time on the elliptic curve is approximately 0.00195s. From Table 3, we can see that the calculation time of our scheme is the shortest.

9. Conclusion

Numerous IoT devices form a huge network to form the Internet of Things. However, these IoT devices have limited resources and are highly vulnerable to various network attacks. To ensure the secure transmission of sensitive IoT data among IoT devices, we propose a new certificateless hybrid signcryption scheme. From the comparison results, we conclude that the proposed approach offloads the optimized computational structure of the original signature scheme and greatly improves the computational performance. Also, the scheme has high computational efficiency. However, this proposed scheme also uses too much bilinear computation, and the reduced computational stress is not significant enough. This scheme can be investigated again in future work targeting the reduction of the number of bilinear computations.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) cost (signature) + cost (encryption)[C]," in *Advances in 17th Annual International Cryptology Conference (Cryptology CRYPTO'97)*, pp. 165–179, Springer-Verlag, Berlin, 1997.

[2] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography[M]," in *Advances in Cryptology-ASIACRYPT 2003*, pp. 452–473, Springer, Berlin Heidelberg, 2003.

[3] R. Cramer and V. Shoup, "Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack," *SIAM Journal on Computing*, vol. 33, no. 1, pp. 167–226, 2004.

[4] A. W. Dent, "Hybrid signcryption schemes with insider security," in *Proceedings of the 10th Australasian Conference on Information Security and Privacy. Lecture Notes in Computer Science Volume 3574*, pp. 253–266, Brisbane, Australia, 2005.

[5] D. Aranha, R. Castro, and J. Lopez, "a1. Efficient certificateless signcryption [EB/OL]," in *Proceedings of 8th Brazilian Symposium on Information and Computer Systems Security*, Gramando, Brazil, March 2008, http://labcom.inf.ufrgs.br/labcom/ceseg/anais/2008/data/pdf/st03_01_resumo.pdf.

[6] C.-h Wu and Z.-x Cheng, "A new efficient certificateless signcryption scheme[C]," in *Proceedings of ISISE 2008.IEEE Computer Society*, pp. 661–664, NW Washington, DC, 2008.

[7] S. S. D. Selvi, S. S. Vivek, and D. Shukla, "Efficient and provably secure certificateless multi-receiver signeryption [C]," in *Proceedings of ProvSec 2008, LNCS 5324*, pp. 52–67, Springer-Verlag, Berlin, 2008.

[8] W.-j. Xie and Z. Zhang, "Efficient and provably secure certificateless signeryption from bilinear maps[C]," in *Proceedings of WCNIS 2010*, pp. 558–562, IEEE Press, 2010.

[9] L. Peng-cheng, M.-x. He, and L. Xiao, "Efficient and prably secure certificateless signcryption from bilinear pairings," *Journal of Computational Information Systems*, vol. 6, no. 11, pp. 3643–3650, 2010.

[10] Z. Liu, Y. Hu, X. Zhang, and H. Ma, "Certificateless signcryption scheme in the standard model," *Information Sciences*, vol. 180, no. 3, pp. 452–464, 2010.

[11] Y. X. Sun and H. Li, "ID-based signcryption KEM to multiple recipients," *Chinese Journal of Electronics*, vol. 20, no. 2, pp. 317–322, 2011.

[12] W.-H. Liu and C.-X. Xu, "Certificateless signcryption scheme without bilinear pairing," *Journal of Software*, vol. 22, no. 8, pp. 1918–1926, 2011.

[13] K. Singh, "Identity based hybrid signcrption revisited," in *International Conference on Information Technology and E-Sercices*, pp. 1–7, 2012.

[14] G. Swapna, P. V. Reddy, and T. Gowri, "Efficient identity based multi-proxy multi-signcryption scheme using bilinear pairings over elliptic curves," in *International Conference on Advances in Computing, Communications and Informatics and Informatics*, pp. 418–423, 2013.

[15] F. G. Li, M. Shirase, and T. Takagi, "Certificateless hybrid signcryption," *Mathematical and Computer Modelling*, vol. 57, no. 3-4, pp. 324–343, 2013.

[16] C. W. Lai, R. F. Lai, D. Zhong, and F. G. Li, "An efficient multi-party hybrid sign-off scheme," *Science Technology and Engineering*, vol. 14, no. 17, pp. 83–87, 2014.

[17] Y. Zhang, D. Zhou, and C. Li, "Funny certificateless generalized designated verifier aggregation signature scheme," *Journal of Communication*, vol. 36, no. 2, pp. 48–55, 2015.

[18] Y. W. Zhou, B. Yang, and Q. L. Wang, "Certificate-resistant leakage-free hybrid signoff mechanism for provable security," *Journal of Software*, vol. 27, no. 11, pp. 2898–2911, 2016.

[19] P. Xu and W. Xue, "Publicly verifiable certificateless hybrid signature encryption scheme," *Computer Applications in Software*, vol. 34, no. 11, 2017.

[20] Y. Zhang and J. Hou, "An efficient certificateless hybrid signature encryption scheme for electronic authentication," *Information Security Research*, vol. 5, no. 10, pp. 879–886, 2019.

[21] H. F. Yu and B. Yang, "Provably secure certificateless hybrid signcryption," *Chinese Journal of Computers*, vol. 38, no. 4, pp. 804–813, 2015.

[22] C. Jin, X. Li, P. Wei, and L. Wang, "A new certificateless hybrid signing secret," *Computer Application Research*, vol. 28, pp. 3527–3531, 2011.