Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 8030510, 10 pages
https://doi.org/10.1155/2022/8030510

Research Article

A Novel Intelligent-Based Intrusion Detection System Approach Using Deep Multilayer Classification

A. Ugendhar,[1] Babu Illuri,[2] Sridhar Reddy Vulapula,[3] Marepalli Radha,[4] Sukanya K,[5] Fayadh Alenezi,[6] Sara A. Althubiti,[7] and Kemal Polat[8]

[1]*Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus, Ibrahimpatnam, Hyderabad, Telangana-501506, India*
[2]*Department Electronics and Communication Engineering, Vardhaman College of Engineering, Hyderabad, India*
[3]*Department of Information Technology, Vignana Bharathi Institute of Technology, Hyderabad, India*
[4]*Department of Computer Science and Engineering, CVR College of Engineering, Mangalpalli (V), Ibrahimpatnam (M), R R District, Hyderabad, Telangana 501510, India*
[5]*Department of E.C.E, TKR College of Engineering and Technology, Meerpet, Ranga Reddy, Hyderabad, Telangana-500097, India*
[6]*Department of Electrical Engineering, Jouf University, Sakaka 72388, Saudi Arabia*
[7]*Department of Computer Science, College of Computer and Information Sciences, Majmaah University, Al-Majmaah 11952, Saudi Arabia*
[8]*Department of Electrical and Electronics Engineering, Bolu Abant Izzet Baysal University, Bolu, Turkey*

Correspondence should be addressed to Kemal Polat; kpolat@ibu.edu.tr

Received 15 March 2022; Accepted 13 April 2022; Published 6 May 2022

Academic Editor: Musavarah Sarwar

Cybersecurity in information technology (IT) infrastructures is one of the most significant and complex issues of the digital era. Increases in network size and associated data have directly affected technological breakthroughs in the Internet and communication areas. Malware attacks are becoming increasingly sophisticated and hazardous as technology advances, making it difficult to detect an incursion. Detecting and mitigating these threats is a significant issue for standard analytic methods. Furthermore, the attackers use complex processes to remain undetected for an extended period. The changing nature and many cyberattacks require a quick, adaptable, and scalable defense system. For the most part, traditional machine learning-based intrusion detection relies on only one algorithm to identify intrusions, which has a low detection rate and cannot handle large amounts of data. To enhance the performance of intrusion detection systems, a new deep multilayer classification approach is developed. This approach comprises five modules: preprocessing, autoencoding, database, classification, and feedback. The classification module uses an autoencoder to decrease the number of dimensions in a reconstruction feature. Our method was tested against a benchmark dataset, NSL-KDD. Compared to other state-of-the-art intrusion detection systems, our methodology has a 96.7% accuracy.

1. Introduction

Internet-enabled services have grown exponentially in recent years. According to current estimates, more than 60 billion Internet-connected gadgets will be available by 2023 [1]. Despite this, computer networks are continually at risk of attack from threat hackers via the Internet. The concept of intrusion detection system (IDS) was first proposed by [2]. Since then, a number of IDS products have been developed and refined to meet the needs of network security. However, because of the rapid advancement of technology over the previous decade, the size of networks and the number of applications handled by network nodes have been increased significantly. As a result, a massive amount of critical data is being generated and shared across various network nodes. These data and network nodes' security have grown increasingly difficult due to many threats generated either through the mutation of an existing assault or through the development of a special attack. Security concerns can affect almost every node in a network [3]. For example, the data

node may be highly crucial for a company. The company's reputation and financial losses could be severely impacted if the node's information is compromised. Ineffectiveness in detecting various attacks, including zero-day attacks, and minimizing false alarm rates has been demonstrated by existing IDSs (FAR). As a result, there is a growing demand for a network intrusion detection system that is efficient, accurate, and cost-effective to ensure robust network security [4]. Figure 1 shows the cyberattacks on the MacAfee network in 2021.

With the help of firewalls and IDSs, various security threats can be effectively countered in a single system. Misuse and anomaly detection schemes are the two basic types of IDS schemes that can be implemented using various machine learning approaches. Detection systems rely primarily on the signatures of security threats and malicious activity to allow multiclass classification and multilevel detection. The IDS, on the other hand, is unable to identify new assaults in which its signature does not exist. Therefore, these systems benefit from being better able to detect known harmful behavior and its variations. As an alternative, anomaly detection-based IDS techniques rely on the usual behavior of users to detect new threats and only support binary classifications [5]. It is important to keep user profiles up-to-date in dynamic companies where roles occasionally shift [6]. As a result, some anomaly detection techniques may have an issue with false positives. Machine learning techniques are being used in various scenarios, including anomaly detection and misuse detection [7]. Because of the absence of labelled training datasets and the heavy reliance on retrieved features extracted by humans, conventional machine learning approaches cannot be deployed on big platforms [8]. In machine learning, deep learning is a new paradigm that uses artificial neural networks (ANNs) and has a better performance than existing methods.

Researchers have developed several ML and DL-based methods to improve NIDSs' ability to detect malicious assaults over the past decade. Although network traffic has risen, NIDSs' ability to identify malicious intrusions has been restricted by the increased number of security threats that have resulted. To better detect network intrusions, researchers are just beginning to look into the potential of applying deep learning (DL) algorithms in NIDSs. Traditional security methods cannot be directly applied to IoT devices because of their limited computational and basic resources. Rule-based detection approaches, on the other hand, were found to be effective [9] As a result, anomaly-based detection procedures are essential as IoT surroundings and technology keep growing.

Deep neural networks (DNNs), including convolutional neural networks (CNN) [10], deep reinforcement learning (DRL) [11], and hybrid DNN structures (HDNN) [12–19], are being studied for their intrusion detection capabilities. Shallow neural networks (SNNs) are a subset of ANNs and the primary focus of deep learning research. Distinct from the more traditional SNNs with a hierarchy of networks, DNN can simulate more complex models because of its better modeling and abstract representation capabilities.

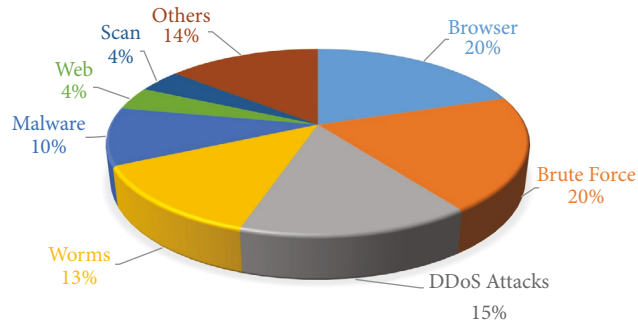

Figure 1: Cyberattacks on the MacAfee network in 2021.

As a result, DNNs have a great deal of potential for creating helpful techniques by making use of excellent data representation.

1.1. Problem Statement. A single algorithm is commonly used in traditional ML-based intrusion detection, with low detection rates, rigid techniques, and high-dimensional data. When designing an intrusion detection framework for the modern Internet, it is important to keep in mind that it must react quickly and easily to the constantly changing environment. A wide-ranging intrusion detection framework is presented in this article, which can enhance the effectiveness of IDSs in many different ways. Traditional supervised machine learning techniques can benefit from DNN's ability to produce more accurate data representations. However, the time complexity of some approaches, which rely on deep learning techniques, limits their effectiveness.

The autoencoder (AE) model has inspired us to perform experiments using the AE model in real-world IDS applications. First, high-dimensional redundant features are converted into a hyperspace representation linked to input data to lessen the training complexity and impact of high-dimensional redundant features. We used AE and a deep multilayer classifier to improve the classifications task.

The following is a list of the important contributions of this work:

(i) Innovation in IDSs based on data analytics and deep multilayer classification techniques is being developed;

(ii) Designing and development of an IDS capable of efficiently distinguishing between distinct cyber-attack classes in the NLS-KDD dataset with high accuracy;

(iii) Development of an IDS with significant industrial application potential.

The rest of the article is structured as follows: Section 2 briefly discusses some of the essential related works. A detailed presentation of the preliminaries is discussed in Section 3. Section 4 presents the proposed deep multilayer-based approach and autoencoders. Section 5 describes the features of the NSL-KDD dataset and algorithm. Results and discussion are presented in Section 5. Finally, Section 6 provides the conclusion and future scope.

2. Literature Survey

The KDD99 and NSL-KDD datasets have been used in the literature to assess various IDSs. Assault classes in the NSL-KDD dataset were discovered using a three-layer MLP created by Yong et al. [20]. The system's accuracy was 79.9% for multilayer classification and 81.2% for binary classification on the test set. Chawla et al. [21] found a binary classification accuracy of 75.49% utilizing self-organizing maps while testing their method on the NSL-KDD dataset (SOMs). Sadiq et al. [22] used MLP and other classical learning methods to get a binary classification accuracy of 95.7%. There was $k = 10$ folds in the dataset, but this was done by the authors. Ishaque et al.'s [23] semisupervised learning approach is based on fuzzy and ensemble learning theories. An accuracy rating of 84% was achieved on the KDD test set using the NSL-KDD dataset. Deep belief networks (DBNs) for multilayer classification were created by Mighan et al. [24] using a restricted Boltzmann machine (RBM) architecture with a Softmax output layer. It was determined that the proposed approach was quite accurate, with only a false alarm rate of 2.47%, even though just 10% of the KDD99 test samples were employed. SDN was used to create a DNN for the purpose of anomaly detection in [25]. Training a neural network with three hidden layers was made possible thanks to the NSL-KDD dataset. Only six criteria and a two-way discriminating procedure have been utilized, as opposed to the usual (normal vs. abnormal). The results of the experiments were correct 75% of the time. Deep neural networks trained on the KDD99 dataset have been proposed by Liu et al. [26]. A gradient-enhanced machine makes it simpler to detect intrusions (GBM). The GBM parameters were fine-tuned using a grid search. For this investigation, the data from UNSW-NB15, NSL-KDD, and GPRS were all used. When it comes to accuracy and specificity testing, GAR forest, tree-based ensembles, and fuzzy classifiers are all outperformed by this approach. A random forest-based IDS's false alarm rate and accuracy were also assessed in [27]. Also considered were data from GPRS, NSL-KDD, and UNSW-NB15. This classifier is put up against others like Multilayer Perceptrons [28], NBTrees [29], a Random Tree ensemble [30], and Nave Bayes [31]. Study indicated that random forest-based IDSs beat other classifiers in terms of performance. Scan attacks, DoS attacks, and MITM subsets of ordinary traffic were analyzed by Farahnakian et al. [31]. The combined DoS, scans, Mirai, and MITM assaults that were included in our analysis were not investigated for intrusion activities. A different study used a multistage classification technique based on clustering and oversampling [13–20] to forecast whether or not the intrusion would occur.

2.1. Deep Learning-Based Intrusion Detection System. Commercial NIDS uses statistical measures or calculated thresholds to represent packet length, interarrival time, flow size, and other network traffic metrics [32]. False positive and false negative alarms are frequent occurrences. False negative notifications suggest that the NIDS is less likely to detect attacks. In contrast, many false positive alerts show that the NIDS is more likely to warn even when no attack has occurred. Commercial solutions are ineffective because of today's threats [33–38].

A self-learning is a powerful tool for confronting today's threats. Unsupervised and semisupervised machine learning techniques are used to analyze different normal and malicious processes utilizing a vast corpus of regular and attack network and host-level events. Commercial viability for machine learning-based solutions is still in its infancy, but the literature on the topic is beginning to emerge. Current machine learning approaches have a high percentage of false positives and a high computational cost [39]. Machine learning classifiers can learn about basic TCP/IP features because of the localization of these features. TCP/IP information is sent through numerous hidden layers to create hierarchical feature representations and hidden sequential links in deep learning. Deep learning has dramatically improved AI operations such as image processing, audio identification, and natural language processing [40]. As a result of its capability to learn new, previously unknown patterns from raw data, deep learning is often used in cybersecurity. To discover more complex traits, it employs a sequence of adjustments. Classification, picture identification, self-driving cars, and speech recognition are just some of the problems that deep learning and large datasets are being utilized to solve. Unknown layers are used to automatically choose features or mining properties and then execute training and testing on the given dataset to acquire classification results. In contrast to conventional machine learning, deep learning does not initially require the extraction of features, as is the case with regular machine learning. Various methods for deep learning are available, for example autoencoder. A support vector machine is used to learn features based on stack autoencoders rather than a Softmax in the STL-IDS architecture introduced in [41,42]. SVM outperformed Naive Bayes, random forest, and J48 on the NSL-KDD dataset with respect to classification accuracy and training and testing durations. Recurrent neural networks were employed by H. Luo et al. [43] in order to detect intrusions (RNN). 83.28% of the time, they got it right. The active deep learning system proposed by O Ludtke et al. [44] is a self-taught (STL) technique for learning features and dimensions. The sparse autoencoder device can be used to reshape a unique feature illustration in an unsupervised manner. SVM is being used to increase the study's classification accuracy and speed. The two- and five-category classifications are likewise shown to have upright computations. J48, Naive Bayesian RF, and SVM have a lower precision rate in five-category classification than the SVM technique. M. Ahmed et al. [45] created a deep learning conjecture using feature extraction to build an IDS deep learning model. GRUs, MLPs, and Softmax modules were all part of the neural system he demonstrated for detecting intrusions, among other things. The investigation used both KDD and NSL-KDD datasets. According to this study, the KDD 99 and NSL-KDD datasets were better served by utilizing BGRU and MLP together. For example, convolutional neural systems and autoencoders have been

extensively investigated by Bansod et al. [46]. Keras and Theano backends were used to train the model on a GPU-based test platform. Several organizational measures were used in this study, including the recipient working attribute, the area under the arc, the precision-recall curve, the mean average precision, and the classification accuracy.

3. Preliminaries

3.1. Autoencoder. Multilayer neural networks known as "autoencoders" provide the same output as their inputs with minimal reconstruction error since the output is similar to the input and has a small number of minimized variances. Unsupervised learning is used by the autoencoder to decode or reassemble the encoded output. Data may be reduced in dimension, features can be extracted, images can be compressed, and noise can be reduced by using an autoencoder. To keep things simple, we describe the general construction of an autoencoder without diving into specifics. Figure 2 gives the block scheme of the autoencoder.

The four major components of a general autoencoder are the encoder, bottleneck, decoder, and reconstruction loss. Data from the input are further compressed using an encoder, which helps to reduce the number of features the model must deal with. The bottleneck is the layer of input data that has the most compressed data with the lowest features. Using a decoder, a model is able to decode the encoded representation and verify that output and input are exactly alike. Finally, the term "reconstruction loss" refers to the difference between the output of a decoder and the original input while evaluating its performance. In addition, backpropagation is used for training and to further minimize reconstruction losses. The purpose of AE is to achieve this minimum loss. Compression of the input x into $z = E(x)$ is achieved via the encoder function E. The decoder D will attempt to recreate the input as $x' = D(E(x))$. The difference between the encoded and decoded vectors is the reconstruction loss in this case. Reconstruction loss can be measured using the mean squared error (MSE) technique:

$$\text{Loss}(E, D) = \frac{1}{n}\sum_{i=1}^{n}\left(x^i - D\left(E\left(x^i\right)\right)\right)^2. \tag{1}$$

Using Kullback–Leibler (KL) divergence, variational autoencoders (VAEs) may calculate reconstruction loss. Data in the latent space and data projected into the latent space have different probability distributions, which the KL divergence measures. This nonnegative number indicates the degree to which the two distributions differ.

There are a variety of autoencoders, such as denoising, variational, convolution, and sparse autoencoders.

3.2. Deep Neural Network. We proposed an MLP model technique since biological neural network features influence it. An MLP known as a feedback neural network is represented as inputs that can be passed from one node to another using a loop in the system. In mathematical terminology,

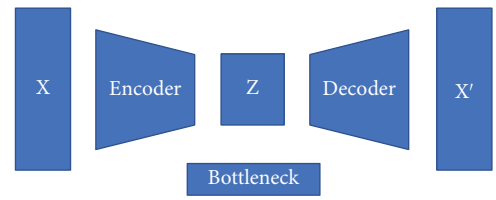

Figure 2: Autoencoder.

each layer of the MLP model contains a significant number of neurons or units. Three or more layers, each with one or more hidden layers, make up this model, including an output layer. The number of hidden layers may be determined using a hyper-parameter selection strategy. Neural connections between layers allow information to move from one layer to the next. In mathematics, the MLP is defined as O: $\mathbb{R}^m \times \mathbb{R}^n$, where m is the size of the input vector $x = x_1, x_2, \ldots, x_{m-1}, x_m$, and N is the size of the output O(x) vector, which is a function of x. Each of the h_i layers can be computed as follows:

$$h_i(x) = f\left(w_i^T x + b_i\right), \tag{2}$$

where h_i: $\mathbb{R}^{di-1} \longrightarrow \mathbb{R}^{di}$, f: $\mathbb{R} \longrightarrow \mathbb{R}$, $w_i \in \mathbb{R}^{ddi-1}$, $b \in \mathbb{R}^{di}$ the size of the input is denoted by the variable d_i, and the nonlinear activation function is denoted by the variable f, which can be either a sigmoid (with values in the range [0, 1]) or a tangent function (values in the range [1, -1]). Figure 3 shows the deep neural network architecture.

4. Proposed Framework

This research proposes a multilayer classification strategy for detecting both the presence of an intrusion and the type of intrusion in the Internet of Things networks under the assumption of an unbalanced type of data. Training and testing datasets are separated, and the proposed method is implemented. The core of the proposed intrusion detection framework consists of preprocessing, autoencoding, databases, classification, and feedback modules. These diverse functional modules are maintained to construct a practical intrusion detection framework with high accuracy and low training complexity. The colored lines in Figure 4 show these functions: the black line is for detection, orange is for retraining, and green is for restoration. Blue two-way lines depict processes that cross with other functions. Figure 4 presents the architecture of proposed framework.

The Softmax function is the nonlinear activation function in our MLP model for the classification problem of multiclass. Each class's probabilities are output of the Softmax function, which selects the biggest value among the probabilities to provide a more accurate result for each class. All three activation functions' mathematical formulas are given below:

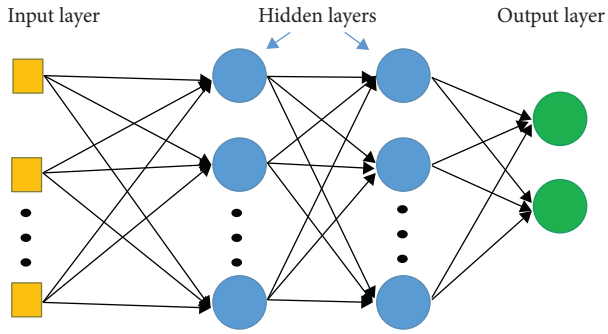

Figure 3: Deep neural network architecture.

$$\text{Sigmoid} = \frac{1}{1 + e^{-x}}, \tag{3}$$

$$\text{Softmax}(x_i) = \frac{e^{xi}}{\sum_{j=1}^{n} e^{xj}}, \tag{4}$$

where input is defined as x.

Multiclass logistic regression is the same as a three-layer MLP with a Softmax function in the output layer. In broad terms, MLP for a large number of hidden layers is formulated as follows:

$$H(x) = H_1(H_{l-1}(H_{l-2}(\ldots(H_1(x))))). \tag{5}$$

In order to enhance deep learning efficiency, our method is distinguished by its modeling of loss functions and ReLU, which are discussed in detail below.

4.1. Preprocessing. Due to the fact that the training and testing datasets contain both numerical and nominal values, they are normalized. Every feature should be scaled the same while normalizing values. Our method takes into account all of the dataset's characteristics. As a result, each feature is essential.

4.2. Loss Functions. In order to get the most performance out of an MLP model, it is critical to choose an ideal parameter. As a first stage, this incorporates the loss function. The difference between the expected and actual values is calculated using a loss function, which is expressed as follows:

$$d(t, p) = \|t - p\|_2^2, \tag{6}$$

where t stands for the desired value and p stands for the predicted value. Using $p(pd)$ as the distribution of probabilities, multiclass classification uses the negative log probability with t as the target class:

$$d(t, p(pd) = -\log\, p(pd)_t. \tag{7}$$

To speed up the learning process, researchers have found that a technique known as the "rectified linear unit" (or "ReLU") has a high level of proficiency. As a result of ReLU, the vanishing and exploding gradient problem is significantly reduced in the history of neural networks. Compared to the standard nonlinear activation functions like sigmoid and tangent [47], it is proven to be the most efficient way to train large datasets in terms of time and cost. As a result of this nonlinearity, neurons are referred to as [34]. ReLU is expressed as follows:

$$f(x) = \max(0, x), \tag{8}$$

where input is defined as x.

4.3. Autoencoder Training. The autoencoder is trained only on standard data packets (Figure 5). This method has various advantages. NSL-class KDD's imbalance can be overcome by training the AE exclusively on typical traffic. It enables the model to distinguish between legitimate and malicious data transmission as a secondary benefit. Thus, real-time applications like fog devices can be better served because we can immediately decide whether or not data transmission is normal or under attack. Figure 5 shows the normal data are used for training the autoencoder.

Dataset for developing an autoencoder; based on the label or class of each data packet sample, D was separated into normal and attack datasets, respectively.

$$\begin{aligned} D_0, D_1 &\leftarrow \text{split}(D), \\ \text{where,} \quad D_0 &\leftarrow (x_i, y_0), \\ i &= 1, 2, \ldots, k, \\ D_1 &\leftarrow (x_i, y_i), \\ i &= 1, 2, \ldots, N - k, \end{aligned} \tag{9}$$

where D_0 is the "normal" dataset and D_1 is the "attack" dataset. On D_0, we train the AE. The number of outputs generated by the AE is the same as the number of inputs; however, there is a loss in reconstruction for each x_i. Attack data have a substantially larger reconstruction loss because the AE is only trained on "normal" data. An experiment led us to a point at which the value of reconstruction loss exceeded a certain threshold. An "attack" data point is defined as the one that has a reconstruction loss greater than the threshold value; otherwise, the data point is considered "normal."

5. Results and Discussion

Experiments were carried out on NSL-KDD incursion data, a condensed form of KDDCup 99 data. It is possible to delete redundant connection records from the test data in KDDCup 99 by applying filters. The outcomes were obtained after implementing the multilayer technique. The studies were carried out on a personal computer with an Intel core i7-1065G7 processor and 1.30 GHz/16 GB of RAM, imbalanced-learn, Scikit Learn [48], and Keras [49]. To test the suggested concept, Python libraries were employed. The NSL-KDD dataset consists of 41 distinct features. Nominal, binary, and numeric features are subclasses. Nominal data cannot be used directly by an autoencoder.

All the input data must be in the form of a number. We used the deep multilayer classification approach to

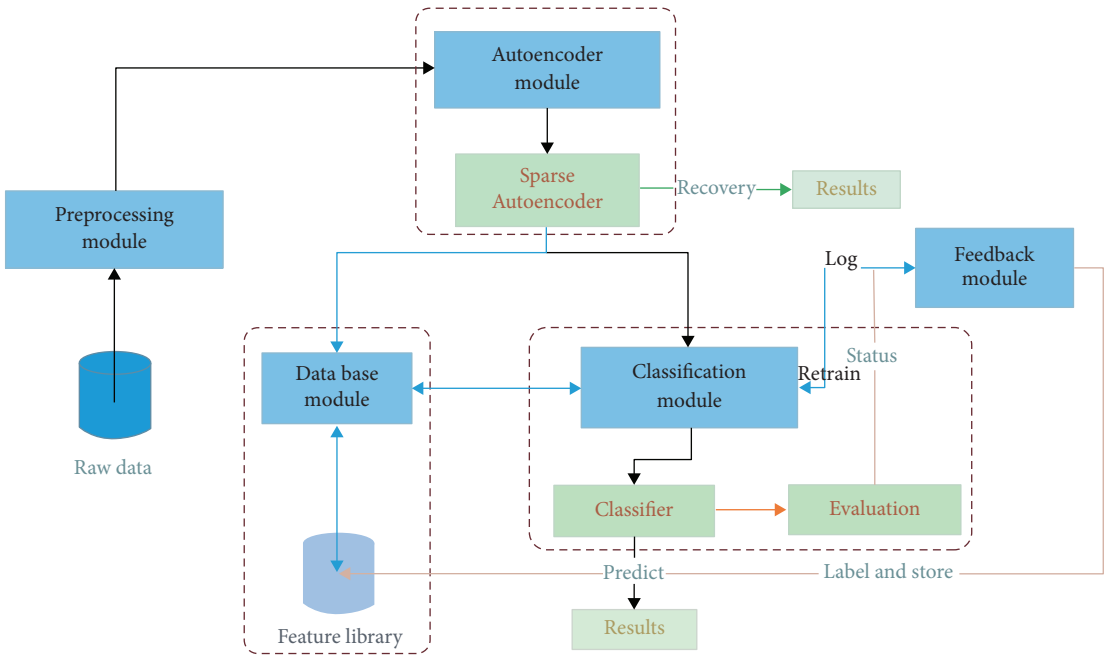

Figure 4: Architecture of proposed framework.

preprocess the nominal or category information. Using the MinMax Scaler functions, the remaining characteristics are preprocessed. As a result of this operation, the 41 characteristics were multiplied by 2. The autoencoder is then fed these features. The parameters of the autoencoder were kept to a minimum. For the first detection step, we use an autoencoder. A "dropout layer" was added to the autoencoder's input to prevent overfitting. This layer serves as a restriction on regularization. Autoencoding is prevented from replicating the input to create output using this input validation method. The dropout layer removes a random number of neurons from the input when training. Autoencoders have a single unnoticed hidden level. We found that the number of neurons in this hidden layer had a significant impact. Low precision is caused by a reduction in reconstruction error due to more neurons. The model's accuracy is also affected by the number of neurons in the system. According to our findings, neurons in the range of 4 to 10 in the hidden layer produce the best results. An "attack" is defined using a threshold value. There is a difference between an attack and a typical instance based on reconstruction error. We used model loss across training data instead of validation data to arrive at this result. Figure 6 shows that reconstruction error and neuron count are correlated. Figure 7 denotes the loss vs epoch during training and testing process using AE. Figure 8 presents the overall performance accuracy evaluation of the system using AE. Figure 9 gives the graphical representation of loss vs epoch during training and testing process using Deep MLP. Figure 10 shows the overall performance accuracy evaluation of the system using deep multilayer network.

5.1. Comparison with Recent State-of-the-Art Techniques. An extensive amount of study has been done on intrusion detection due to its importance in today's cyber environment.

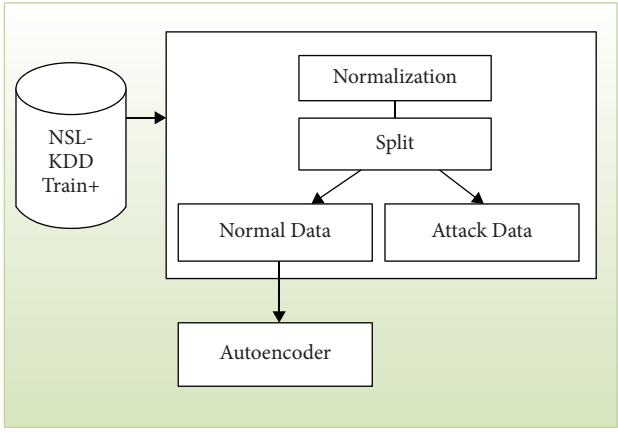

Figure 5: Train the autoencoder by normal data.

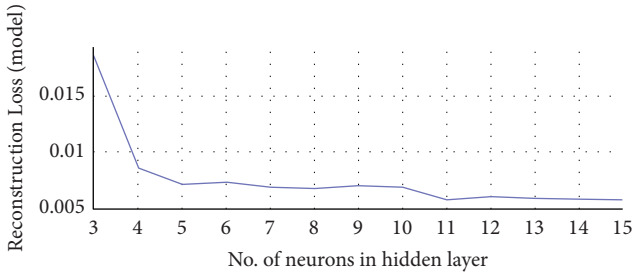

Figure 6: Correlation of reconstruction error and neuron count.

Detecting incursions using machine learning has been done in several methods. Over NSL-KDD, our method scores among the top in terms of accuracy when identifying intrusions using standard machine learning and deep learning techniques. Table 1 reveals that autoencoder-based

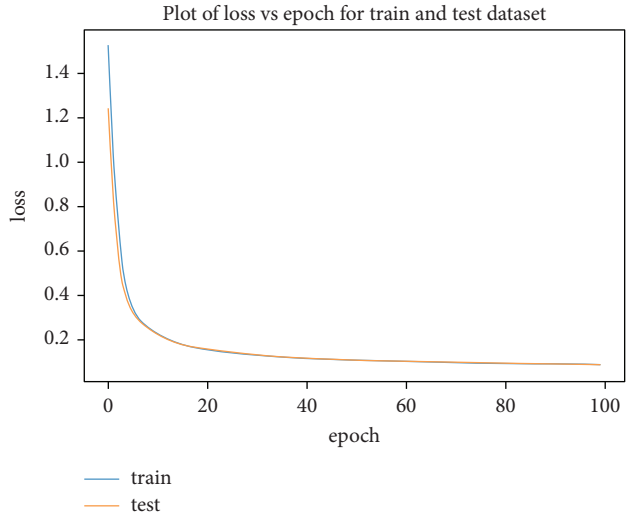

Figure 7: Loss vs epoch during training and testing process using AE.

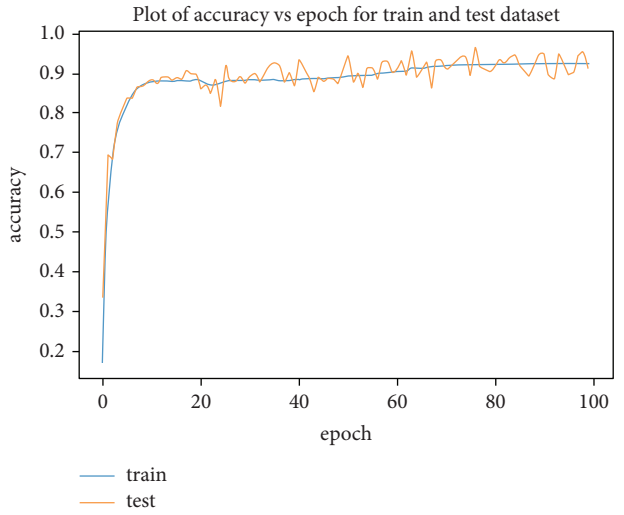

Figure 8: Overall performance accuracy evaluation of the system using AE.

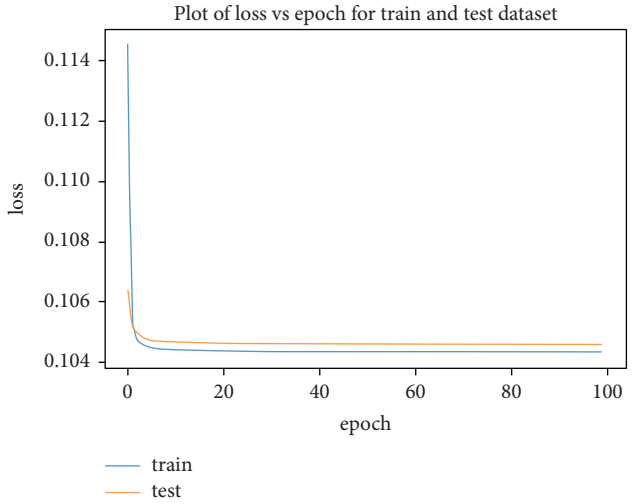

Figure 9: Graphical representation of loss vs epoch during training and testing process using deep MLP.

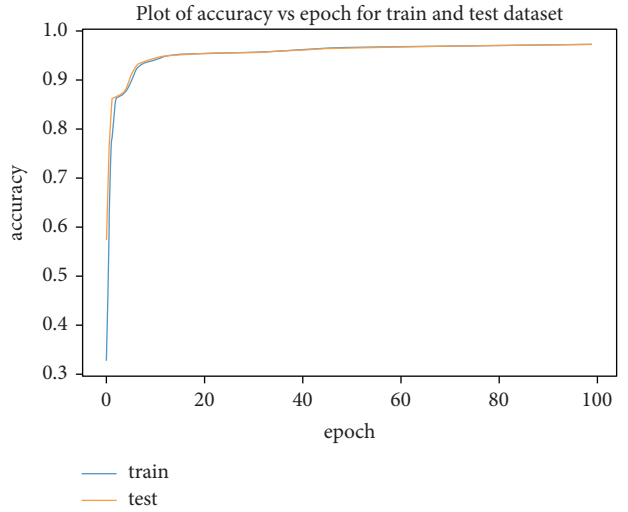

Figure 10: Overall performance accuracy evaluation of the system using deep multilayer network.

Inputs: X - input dataset,
Subsampling size
Output: Reconstruction loss for anomaly test data
Step 1: Initialize data = { };
Step 2:# Initializing a MinMax Scaler
scaler = MinMaxScaler()
Step 3:# Instantiating the Autoencoder
model = Autoencoder()
# creating an early_stopping
early_stopping = EarlyStopping(monitor = 'val_loss',
patience = 2,
mode = 'min')
# Compiling the model
model.compile(optimizer = 'Adam',
loss = 'mae')
Step 4: # mlp = Sequential() # initializing model
input layer and first layer with 50 neurons
mlp.add(Dense(units = 50, input_dim = X_train.shape
[1], activation = 'relu'))
# output layer with softmax activation
mlp.add(Dense(units = 5,activation = 'softmax'))

Algorithm 1: Deep multilayer classification.

Table 1: Performance of the proposed IDS with the recent state-of-the-art techniques.

S.No	Method	Accuracy
1	RNN [2]	78.32
2	AE [39]	89.34
3	DNN + KNN [4]	92.14
4	ND-tree [33]	82.90
5	Isolation forest [46]	92.50
6	Proposed method	96.70

Table 2: Comparison with other similar research work.

Reference	Systematic study	Focused on NIDS	AI-based techniques		Future scope
			ML	DL	
Yong et al. [20]	×	×	✓	×	✓
Sadiq et al. [22]	×	×	✓	✓	✓
Marta et al. [2]	×	✓	✓	✓	✓
Zhang et al. [4]	×	×	✓	×	✓
Thamilarasu et al. [8]	×	✓	✓	✓	✓
Farahnakian et al. [31]	×	×	✓	×	✓
Proposed approach	✓	✓	✓	✓	✓

approaches outperformed the competition. NSL-KDDTrainC and NSL-KDDTestC datasets were used to test the procedures in Table 1.

6. Conclusions

Deep multilayer classification autoencoder-driven intelligent intrusion detection was proposed in this article. The NSL-KDD dataset was used as a baseline for the proposed IDS. The AE architecture was fed with the most important properties discovered by data-driven deep learning, which comprises a single hidden layer with 50 units (AE50). According to Table 1 and recent state-of-the-art, the suggested AE50 classifier was compared with deep and classical methods (Table 2). According to comparative results, the deep multilayer classifier outperformed all other approaches, with an accuracy of 96.70%.

A more accurate deep architecture, similar to NSL-KDD instances, will be built in the future to detect malicious assaults as they occur. For real-time analysis of big data, we want to look at how methodologies from [15,16] can be combined with the work we did here. This way, long-term learning, faster decision criteria, and less computational complexity can be used [50].

Data Availability

The datasets used to support the findings of this study are available from the authors upon reasonable request.

Ethical Approval

This article does not contain any studies with human participants. No animal studies were involved in this review.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

All authors contributed equally to this work. In addition, all authors have read and approved the final manuscript and gave their consent to publish the article.

References

[1] D. L. Aguilar, M. A. M. Perez, O. Loyola-Gonzalez, K.-K. R. Choo, and E. Bucheli-Susarrey, "Towards an interpretable autoencoder: a decision tree-based autoencoder and its application in anomaly detection," *IEEE Transactions on Dependable and Secure Computing*, p. 1, 2022.

[2] M. Catillo, A. Pecchia, and U. Villano, "AutoLog: anomaly detection by deep autoencoding of system logs," *Expert Systems with Applications*, vol. 191, Article ID 116263, 2022.

[3] E. Cruz-Esquivel and Z. J. Guzman-Zavaleta, "An examination on autoencoder designs for anomaly detection in video surveillance," *IEEE Access*, vol. 10, pp. 6208–6217, 2022.

[4] H. Zhang, W. Guo, S. Zhang, H Lu, and X Zhao, "Unsupervised deep anomaly detection for medical images using an improved adversarial autoencoder," *Journal of Digital Imaging*, vol. 35, no. 2, pp. 153–161, 2022.

[5] G. Baig Mohammad, S. Shitharth, and P. Revanth Kumar, "Integrated machine learning model for an URL phishing detection," *International Journal of Grid and Distributed Computing*, vol. 14, no. 1, pp. 513–529, 2021.

[6] C. Savaglio, M. Ganzha, M. Paprzycki, C. Bădică, M. Ivanović, and G. Fortino, "Agent-based internet of things: state-of-the-art and research challenges," *Future Generation Computer Systems*, vol. 102, pp. 1038–1053, 2020.

[7] N. Angelova, G. Kiryakova, and L. Yordanova, "The great impact of internet of things on business," *Trakia Journal of Science*, vol. 15, no. 1, pp. 406–412, 2017.

[8] G. Thamilarasu and S. Chawla, "Towards deep-learning-driven intrusion detection for the internet of things," *Sensors*, vol. 19, no. 9, p. 1977, 2019.

[9] R. Williams, E. McMahon, S. Samtani, M. Patton, and H. Chen, "Identifying vulnerabilities of consumer Internet of Things (IoT) devices: a scalable approach," in *Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 179–181, Beijing, China, July 2017.

[10] R. Damasevicius, A. Venckauskas, S. Grigaliunas et al., "LITNET-2020: an annotated real-world network flow dataset for network intrusion detection," *Electronics*, vol. 9, no. 5, p. 800, 2020.

[11] A. Nauman, Y. A. Qadri, M. Amjad, Y. B. Zikria, M. K. Afzal, and S. W. Kim, "Multimedia internet of things: a comprehensive survey," *IEEE Access*, vol. 8, pp. 8202–8250, 2020.

[12] I. Ullah and Q. H. Mahmoud, "A scheme for generating a dataset for anomalous activity detection in IoT networks," in *Proceedings of the Canadian Conference on Artificial Intelligence*, pp. 508–520, Springer, Cham, Switzerland, May 2020.

[13] F. Alenezi, "Image dehazing based on pixel guided CNN with PAM via graph cut," *Computers, Materials & Continua*, vol. 71, no. 2, pp. 3425–3443, 2022.

[14] F. Alenezi, A. Armghan, S. N. Mohanty, R. H. Jhaveri, and P. Tiwari, "Block-greedy and CNN based underwater image dehazing for novel depth estimation and optimal ambient light," *Water*, vol. 13, no. 23, p. 3470, 2021.

[15] G. P. Joshi, F. Alenezi, G. Thirumoorthy, A. K. Dutta, and J. You, "Ensemble of deep learning-based multimodal remote sensing image classification model on unmanned aerial vehicle networks," *Mathematics*, vol. 9, no. 22, p. 2984, 2021.

[16] F. Alenezi and K. C. Santosh, "Geometric regularized Hopfield neural network for medical image enhancement," *International Journal of Biomedical Imaging*, vol. 2021, Article ID 6664569, 12 pages, 2021.

[17] F. Alenezi and E. Salari, "A fuzzy-based medical image fusion using a combination of maximum selection and Gabor filters," *International Journal of Engineering Sciences*, vol. 9, pp. 118–129, 2018.

[18] F. S. Alenezi and S. Ganesan, "Geometric-pixel guided single-pass convolution neural network with graph cut for image dehazing," *IEEE Access*, vol. 9, Article ID 29391, 2021.

[19] S. Majid, F. Alenezi, S. Masood, M. Ahmad, E. S. Gündüz, and K. Polat, "Attention based CNN model for fire detection and localization in real-world images," *Expert Systems with Applications*, vol. 189, Article ID 116114, 2022.

[20] B. Yong, W. Wei, K. C. Li et al., "Ensemble machine learning approaches for webshell detection in Internet of things environments," *Trans. Emerg. Telecommun. Technol.*, p. e4085, 2020.

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[22] A. S. Sadiq, H. Faris, A. M. Al-Zoubi, S. Mirjalili, and K. Z. Ghafoor, "Fraud detection model based on multi-verse features extraction approach for smart city applications," in *Smart Cities Cybersecurity and Privacy*, pp. 241–251, Elsevier, Amsterdam, The Netherlands, 2019.

[23] M. Ishaque and L. Hudec, "Feature extraction using deep learning for intrusion detection system," in *Proceedings of the 2nd Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, pp. 1–5, Riyadh, Saudi Arabia, May 2019.

[24] S. N. Mighan and M. Kahani, "A novel scalable intrusion detection system based on deep learning," *International Journal of Information Security*, vol. 20, pp. 1–17, 2020.

[25] C. Zhang, F. Ruan, L. Yin, X. Chen, L. Zhai, and F. Liu, "A deep learning approach for network intrusion detection based on NSL-KDD dataset," in *Proceedings of the IEEE 13th Int. Conf. Anti-Counterfeiting, Secur., Identification. (ASID)*, pp. 41–45, Xiamen, China, October 2019.

[26] Y. Liu, Q. Liao, J. Zhao, and Z. Han, "Deep learning-based encryption policy intrusion detection using commodityWiFi," in *Proceedings of the IEEE 5th Int. Conf. Comput. Commun. (ICCC)*, pp. 2129–2135, Chengdu, China, December 2019.

[27] R. Zhao, J. Yin, Z. Xue et al., "An efficient intrusion detection method based on dynamic autoencoder," *IEEE Wireless Communications Letters*, vol. 10, no. 8, pp. 1707–1711, 2021.

[28] A. Basati and M. M. Faghih, "APAE: an IoT intrusion detection system using asymmetric parallel auto-encoder," *Neural Computing & Applications*, pp. 1–21, 2021.

[29] W. Xu, Y. Fan, and C. Li, "I2DS: interpretable intrusion detection system using autoencoder and additive tree," *Security and Communication Networks*, vol. 2021, Article ID 5564354, 9 pages, 2021.

[30] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection," *IEEE Access*, vol. 6, Article ID 52856, 2018.

[31] F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," in *Proceedings of the 20th Int. Conf. Adv. Commun. Technol. (ICACT)*, pp. 178–183, Chuncheon, Korea (South), February. 2018.

[32] Y. Yu, J. Long, and Z. Cai, "Session-based network intrusion detection using a deep learning architecture," in *Modeling Decisions for Artificial Intelligence*, pp. 144–155, Springer, Cham, Switzerland, 2017.

[33] D. Ratasich, F. Khalid, F. Geissler, R. Grosu, M. Shafique, and E. Bartocci, "A roadmap toward the resilient internet of things for cyber-physical systems," *IEEE Access*, vol. 7, Article ID 13283, 2019.

[34] N. Daldal, M. Nour, and K. Polat, "A novel demodulation structure for quadrate modulation signals using the segmentary neural network modelling," *Applied Acoustics*, vol. 164, Article ID 107251, 2020.

[35] N. Daldal, A. Sengur, K. Polat, and Z. Cömert, "A novel demodulation system for base band digital modulation signals based on the deep long short-term memory model," *Applied Acoustics*, vol. 166, Article ID 107346, 2020.

[36] N. Daldal, Z. Cömert, and K. Polat, "Automatic determination of digital modulation types with different noises using Convolutional Neural Network based on time-frequency information," *Applied Soft Computing*, vol. 86, 2020 ISSN 1568-4946, Article ID 105834.

[37] M. Nour, N. Daldal, M. F. Kahraman, H. Sindi, A. Alhudhaif, and K. Polat, "A novel tilt and acceleration measurement system based on Hall-effect sensors using neural networks," *Mathematical Problems in Engineering*, vol. 2022, Article ID 7000486, 13 pages, 2022.

[38] M. F. Kahraman and S. Öztürk, "Experimental study of newly structural design grinding wheel considering response surface optimization and Monte Carlo simulation," *Measurement*, vol. 147, Article ID 106825, 2019.

[39] C. Liu, J. Liu, J. Wang, S. Xu, H. Han, and Y. Chen, "An attention-based spatiotemporal gated recurrent unit network for point-of-interest recommendation," *ISPRS International Journal of Geo-Information*, vol. 8, no. 8, p. 355, 2019.

[40] A. Boukerche, L. Zheng, and O. Alfandi, "Outlier detection: methods, models, and classification," *ACM Computing Surveys*, vol. 53, no. 3, pp. 1–37, 2020.

[41] V. Cerqueira, L. Torgo, and C. Soares, "Layered learning for early anomaly detection: predicting critical health episodes," in *International Conference on Discovery Science*, pp. 445–459, Springer, 2019.

[42] V. Garcia-Font, C. Garrigues, and H. Rifà-Pous, "A comparative study of anomaly detection techniques for smart city wireless sensor networks," *Sensors*, vol. 16, no. 6, p. 868, 2016.

[43] H. Luo and S. Zhong, "Gas turbine engine gas path anomaly detection using deep learning with Gaussian distribution," in *Proceedings of the 2017 Prognostics and System Health Management Conference (PHM-Harbin)*, pp. 1–6, IEEE, Harbin, China, July 2017.

[44] O. Lüdtke, A. Robitzsch, and S. G. West, "Regression models involving nonlinear effects with missing data: a sequential modeling approach using Bayesian estimation," *Psychological Methods*, vol. 25, no. 2, pp. 157–181, 2019.

[45] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[46] S. D. Bansod and A. V. Nandedkar, "Crowd anomaly detection and localization using histogram of magnitude and momentum," *The Visual Computer*, vol. 36, no. 3, pp. 609–620, 2020.

[47] J. V. S. d. Chagas, R. F. Ivo, M. T. Guimarães, D. A. Rodrigues, E. D. S. Rebouças, and P. P. F. Rebouças, "Fast fully automatic skin lesions segmentation probabilistic with Parzen window," *Computerized Medical Imaging and Graphics*, vol. 85, no. 12, Article ID 101774, 2020.

[48] A. Gharaibeh, M. A. Salahuddin, S. J. Hussini et al., "Smart cities: a survey on data management, security, and enabling technologies," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2456–2501, 2017.

[49] D. Abadi, "Consistency tradeoffs in modern distributed database system design: CAP is only part of the story," *Computer*, vol. 45, no. 2, pp. 37–42, 2012.

[50] S. Thakur, A. Chakraborty, R. De, N. Kumar, and R. Sarkar, "Intrusion detection in cyber-physical systems using a generic and domain specific deep autoencoder model," *Computers & Electrical Engineering*, vol. 91, Article ID 107044, 2021.