Hindawi
Complexity
Volume 2021, Article ID 5579851, 18 pages
https://doi.org/10.1155/2021/5579851

Research Article

Intrusion Detection System to Advance Internet of Things Infrastructure-Based Deep Learning Algorithms

Hasan Alkahtani[1] **and Theyazn H. H. Aldhyani**[2]

[1]*College of Computer Science and Information Technology, King Faisal University, P. O. Box 400, Al-Ahsa, Saudi Arabia*
[2]*Community College of Abqaiq, King Faisal University, P. O. Box 400, Al-Ahsa, Saudi Arabia*

Correspondence should be addressed to Theyazn H. H. Aldhyani; taldhyani@kfu.edu.sa

Received 28 February 2021; Revised 23 March 2021; Accepted 17 April 2021; Published 7 July 2021

Academic Editor: M. Irfan Uddin

Smart grids, advanced information technology, have become the favored intrusion targets due to the Internet of Things (IoT) using sensor devices to collect data from a smart grid environment. These data are sent to the cloud, which is a huge network of super servers that provides different services to different smart infrastructures, such as smart homes and smart buildings. These can provide a large space for attackers to launch destructive cyberattacks. The novelty of this proposed research is the development of a robust framework system for detecting intrusions based on the IoT environment. An IoTID20 dataset attack was employed to develop the proposed system; it is a newly generated dataset from the IoT infrastructure. In this framework, three advanced deep learning algorithms were applied to classify the intrusion: a convolution neural network (CNN), a long short-term memory (LSTM), and a hybrid convolution neural network with the long short-term memory (CNN-LSTM) model. The complexity of the network dataset was dimensionality reduced, and to improve the proposed system, the particle swarm optimization method (PSO) was used to select relevant features from the network dataset. The obtained features were processed using deep learning algorithms. The experimental results showed that the proposed systems achieved accuracy as follows: CNN = 96.60%, LSTM = 99.82%, and CNN-LSTM = 98.80%. The proposed framework attained the desired performance on a new variable dataset, and the system will be implemented in our university IoT environment. The results of comparative predictions between the proposed framework and existing systems showed that the proposed system more efficiently and effectively enhanced the security of the IoT environment from attacks. The experimental results confirmed that the proposed framework based on deep learning algorithms for an intrusion detection system can effectively detect real-world attacks and is capable of enhancing the security of the IoT environment.

1. Introduction

Currently, there are more than 25 billion devices connected to the Internet worldwide, three times as many human beings [1–3]. The Internet of Things (IoT) is based on interconnected smart devices, and different services are used to integrate them into a single network. This allows the smart devices to gather sensitive information and carry out important functions, and these devices connect and communicate with each other at high speeds and make decisions according to indicator information. The IoT environment uses cloud services as a backend for processing information and maintaining remote control. Client users use mobile applications or web services to access data and control the devices. The IoT infrastructure uses large numbers of sensors to extract significant information, and this information is analyzed by artificial intelligence algorithms [4, 5].

Intrusion detection systems (IDSs) are the technical, regulatory, and administrative means used to prevent unauthorized use, abuse, and recovery of electronic information and communication systems and the information they contain, aimed at ensuring the availability and continuity of the work of the information systems and enhancing the protection, confidentiality, and privacy of personal data by taking all measures. Cybersecurity is the practice of defending computers, servers, mobile devices, electronic

systems, networks, and data from malicious attacks. It is also known as information technology security [6–9]. These intrusions incorporate field of research control systems by controlling an alteration of the document system, heightening benefits, making unapproved logins, accessing sensitive records, and using malware (e.g., infections, Trojan horses, and worms), which can change the condition of the network. Network intrusions occur due to approaching packets in the network system to perform behaviors, such as denial of service (DoS) attacks or even attempts to be split into the system. DoS attacks are attempts to make PC assets unapproachable by their planned clients, for example, land attacks, ping of death (POD), and flood attacks. Indications of intrusions incorporating abnormal outcomes while executing different client charges are exemplified by moderate system execution, and sudden system crashes and changes in parts of information structures are, bizarrely, moderate system implementations (e.g., opening records or accessing sites).

Attackers exploit unknown vulnerabilities and bypass known signatures. The IoT environment is based on a smart grid that uses sensor devices, and these devices connect to each other to pass information. Figure 1 displays the world population and the number of sensor devices required for protection from attackers. With the exponential growth of IoT use, the IoT has become a smart object of attackers achieving their targets. Therefore, using artificial intelligence based on deep learning algorithms can detect unknown vulnerabilities using sensors devices [10].

Artificial intelligence is a kind of information-driven approach in which the first step is to understand the data. Various types of data represent specific attack behaviors, including host behaviors and network activities. Server logs reflect host behaviors, and network traffic represents network behaviors. There are several types of attacks, with each having a particular pattern. Therefore, it is important to select suitable data sources to detect various attacks as per the features of the threat. One of the key features of a DoS attack, for example, is to send several packets in a very short time; thus, flow data are ideal for DoS attack detection. A hidden channel includes a data-leaking operation between two different IP addresses and is best suited for session data detection. Therefore, the advance of deep learning algorithms can help detect these network behaviors [11, 12].

Many studies have proposed the development of network security systems, and artificial intelligence plays a primary role in the area of cybersecurity based on IoT for designing an intelligent system for security in the IoT environment. The proposed research aimed to develop an intelligent model that could help secure the IoT structure and devices from threats. Currently, most companies and organizations have undergone digital transformations through IoT devices. However, this has created new complexities and vulnerabilities that, once cybercriminals learn about them, can be quickly exploited. Jokar et al. [13] developed classification algorithms to detect abnormal electricity consumption. Alseiari et al. [14] used soft computing based on clustering technology to monitor network traffic in advanced metering infrastructure (AMI). Vijayanand et al. [15] applied a support vector machine (SVM) based on a multiclass to detect the IDS, where decision tree algorithms gave very powerful results compared with an SVM proposed by Jindal et al. [16]. Boumkheld et al. [17] used a traditional machine learning algorithm over a naive Bayesian network to test the ability of this algorithm to detect IDS. Zigbee-based Q-learning was proposed by Jokar et al. [18] to protect networks from intrusion, who found it the best strategy for monitoring system attacks. Hasan et al. [19] proposed a hybrid convolution neural network (CNN) with long short-term memory (LSTM) to classify the characteristics of electricity information, and the use of a hierarchy to select significant features from intrusion detection networks was proposed by Wang et al. [20]. CNN and LSTM algorithms have been applied to detect attacks [21]. Ullah et al. [22] introduced a hybrid deep neural network to detect intrusion by combining a CNN and a gated recursive unit. A particle swarm optimization (PSO) algorithm has been used to select significant features from data, and a developing system can automatically perform the processes of selecting features and classifications. In Liu et al.'s [23] research, a CNN algorithm was applied to identify attacks, and it was noted that deep learning based on the CNN improved the system. Xiao et al. [24] adopted an autoencoder to reduce the dimension of the intrusion detection data to decrease the interference of redundant features; these features were processed using a CNN to classify the attacks. Yang et al. [25] used a CNN to detect intrusion for improved extraction of features across layers, and feature fusion has been used to obtain comprehensive features. Yang et al. [26] developed a system to secure the IoT in the healthcare environment; it controlled traffic and made the healthcare environment smarter. Furthermore, security methods have been developed for IoT systems, as described in [27–29]. Other algorithms applied as solutions for the security of DNP3 traffic include statistical approaches and machine learning [30, 31]. Keliris et al. [32] used the support vector machine (SVM) algorithm for classification intrusion, and it was noted that the SVM performed well. It has been suggested that a detection system using machine learning techniques in power systems would be feasible for detecting malicious states [33]. Arrignton et al. introduced a machine learning algorithm based on anomaly based intrusion detection for the protection of IoT devices. Liu et al. [34] developed an IDS using suppressed fuzzy clustering and principal component analysis (PCA) algorithms. Kasinathan et al. [35] developed a system signature-based IDS for low-power wireless personal area network (6LoWPAN)-based IoT networks; this system aimed to detect DoS attacks with the highest accuracy. Danda et al. [36] designed a host-based IDS for the security of IoT network devices using rule-based detection.

Cho et al. [37] proposed machine leaning algorithms to detect the botnet attacks at hosts and network levels on the IoT environment. The feature selection method was presented to select the features of malicious attack behaviors. Diro and Chilamkurti [38] introduced the deep learning to classify the intrusion from host level in IoT. Cruz et al. [39] proposed the intelligent mechanism model to detect the intrusion based on the decision making method moreover

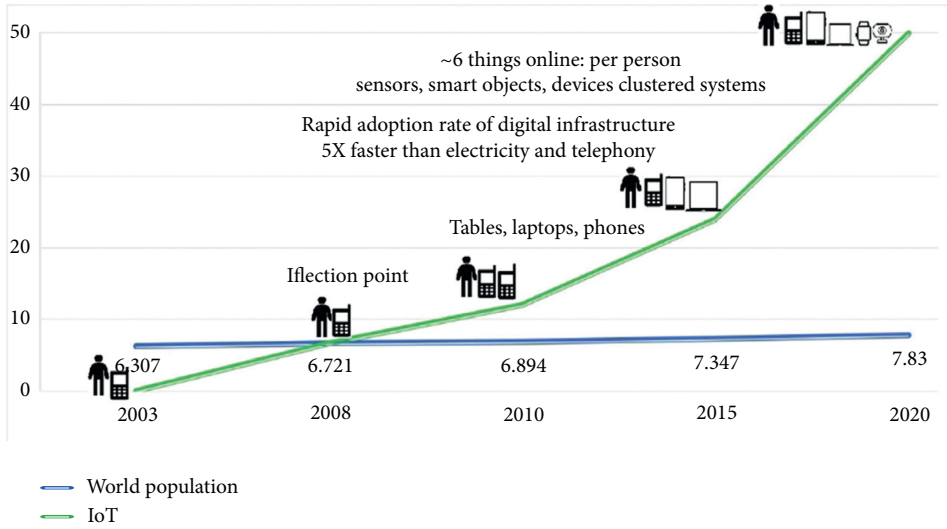

Figure 1: Projecting the "things" behind the internet of things (IoT).

and developed recurrent neural network (RNN) to improve the previous model [40].

Currently, artificial intelligence based on machine learning and deep learning algorithms for data-processing capabilities provide the most effective value to the area of cyber defense by uncovering patterns, shapes, and outliers that indicate potential incidents, even if these solutions do not align with known attack patterns [41]. An IDS is a commonly used security tool for protecting and mitigating the IoT and its infrastructure from unseen and unpredictable intrusions. There are few studies on IDSs in the IoT based on artificial intelligence; therefore, developing a framework and achieving optimal results are the biggest challenges due to the network data having imbalanced data. Our target was to develop a secure, movable framework for securing large IoT networks. Here, we present advanced artificial intelligence, such as deep learning models, namely, CNN, LSTM, and combined CNN-LSTM algorithms. We have significantly expanded the framework to integrate a deep learning algorithm to familiarize it with changing threats to the IoT network for anomaly detection. The main contributions of this study are as follows:

(1) Use of advanced artificial intelligence algorithms such as CNN, LSTM, and a hybrid CNN-LSTM to develop a system to detect intrusions into the IoT environment.

(2) The proposed system was developed using IoT network data that are not commonly used; this dataset was generated in 2020 and was the biggest challenge for developing a robust framework.

(3) The proposed system was compared with a research article that developed these data. It was noted that the results of our system were outperformed.

2. Materials and Methods

Figure 2 displays the framework of the proposed system for detecting IoT environment intrusions. The proposed system is composed by some phases to evaluate for obtaining the best accuracy. The components of the proposed system are described in the following sections.

2.1. IoTID20 Dataset Attack. For this experiment, an IoTID20 dataset attack was conducted to test the proposed framework. The IoTID20 dataset was collected from IoT devices and interconnecting structures; the IoT devices were connected to or installed in a smart home environment, such as SKTNGU and EZVIZ Wi-Fi cameras, to create the IoTID20 dataset. Figure 3 shows the environment of the IoTID20 dataset; the laptops, tablets, and smartphone devices were connected by Wi-Fi to the smart home router. The SKT NGU and EZVIZ Wi-Fi cameras were IoT victim devices, and all other devices in the testbed were the attacking devices.

The newly developed IoTID20 dataset was adopted from Pcap files available online. The dataset contained 80 features and two main label attacks and normal. The IoTID20 dataset attack was generated in 2020. Figure 2 shows the IoT environment of the generated IoTID20 dataset. Table 1 displays all the types of IoTID20 dataset attacks, and the numbers of features for each class label are presented in Figure 4. This dataset was obtained from Kaggle https://sites.google.com/view/iot-network-intrusion-dataset/home.

2.2. Particle Swarm Optimization Method. Preprocessing is a very important stage for improving classification algorithms. IoT data have various types of formats and dimensionality; therefore, dimensionality reduction was necessary to select significant features from the data. The PSO method has been suggested for handling important features from network datasets for detecting malicious attacks. PSO is a population-based computation intelligence method suggested by Eberhat and Kennedy [42], and it is an operative and respected global search system [43]. The PSO algorithm is called a reasonable algorithm because of its simple feature

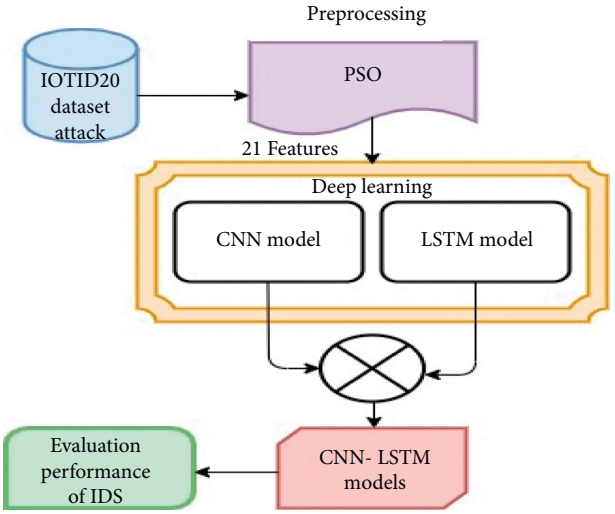

Figure 2: Generic framework of the proposed system.

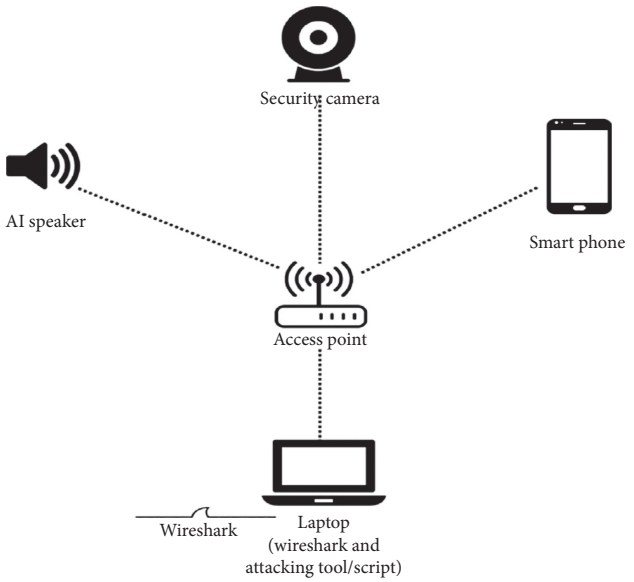

Figure 3: IoTID20 dataset testbed environment.

Table 1: IoTID20 dataset attacks.

Dos	Mirai	Mitm	Scan
Syn flooding	Host brute force HTTP flooding UDP flooding	ARP spoofing services	Host port os

coding, global search, computational reasonability, fewer parameters, and less demanding execution to address and select important feature problems [44]. PSO is used to find important features. Figure 5 shows the particles swarm optimization algorithm steps for selecting significant features from an intrusion network dataset. PSO uses the principal space method for searching space using a subset of primary components that have explored and selected features. For the PSO method, particles are used to represent solutions from the population in the search space particles, which is called a swarm. To generate the particles by distributing 1 and 0 randomly, in the particle, if the principal component is 1, the particle is chosen for another side, and if the particle component is 0, then it is ignored. To make the PSO more powerful, it works randomly and travels in the search space to search for an obtained optimal subset of features by updating their position and velocity. The place of particle i and its rapidity are shown in the following equations:

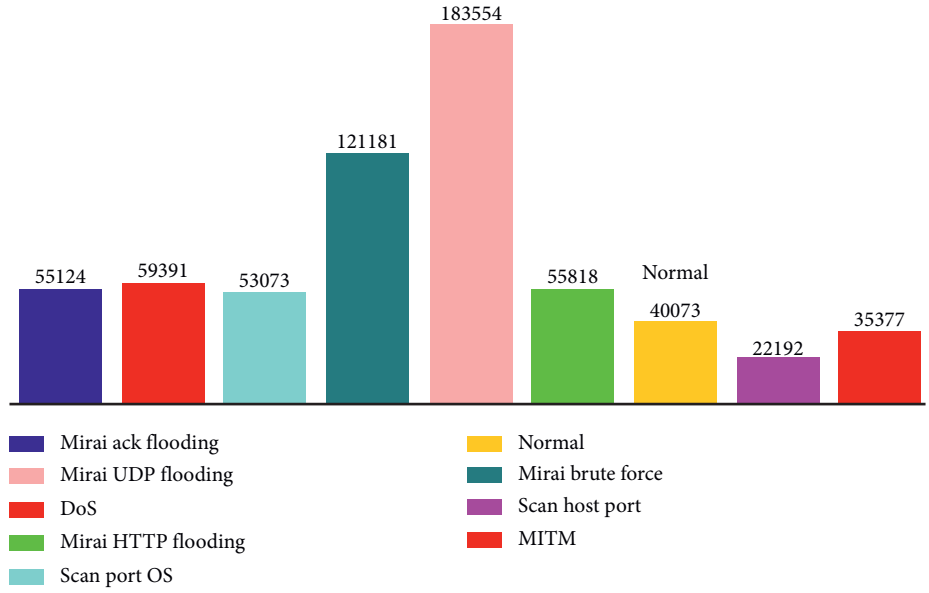

Figure 4: Numbers of instances for each class of IoTID20 dataset.

$$x_i = \{x_{i1}, x_{i2}, \ldots, x_{iD}\}, \tag{1}$$

$$vx_i = \{v_{i1}, v_{i2}, \ldots, v_{iD}\}, \tag{2}$$

where D indicates the search space of the particle. Equation (3) was used to calculate the velocity and position for search space as follows:

$$v_{id}^{t+1} = w * v_{id}^{t} + c_1 * r_{1i} * \left(p_{id} - x_{id}^{t}\right) + c_2 * r_{2i} * \left(p_{gd} - x_{id}^{t}\right), \tag{3}$$

$$v_{id}^{t+1} = v_{id}^{t} + v_{id}^{t+1}, \tag{4}$$

where d is the dimension in the search space, t denotes the iteration in the process for search space, w is the inertia weight, c_1 and c_2 are acceleration constants, r_{1i} and r_{2i} are random values distributed in 0 and 1, and p_{id} and p_{gd} represent the $p$best and $g$best in dimension space in the search space. The values of location and rapidity in each particle are updated until they obtain the best features. Then, the condition is stopped when the iteration reaches the maximum number and obtains satisfactory fitness values.

The IoTID20 dataset was very big, with around 6,332,562 instances for improving the deep learning algorithms. The PSO algorithm was proposed for handling dimensionality reduction. Twenty-one of the most significant features were selected to develop the system. The PSO method used position and velocity for searching the best road to obtain appropriate features from the dataset. We used Iteration 19 $g$best, and the value of fitness was 90.666351, whereas Iteration 20 was used for $g$best and the value of fitness was 90.666351. The significant features obtained using the PSO method are presented in Table 2 (Algorithm 1).

2.3. Correlation Analysis. Pearson's correlation coefficient method was applied to analyze the correlation between the selected features and classes (normal and attacks) for approving the significant subset feature as follows:

$$R = \frac{n\sum(x \times y) - \left(\sum x\right)\left(\sum y\right)}{\left[n\left(\sum x^2\right) - \left(\sum x^2\right)\right] \times \left[n\left(\sum y^2\right) - \left(\sum y^2\right)\right]} \times 100\%, \tag{5}$$

where R is Pearson's correlation coefficient approach, x is training input values of the features, y is input values of classes (normal and attack), and n is total number of input variables.

Table 3 summaries Pearson's correlation coefficient method, and it was employed to evaluate and examine the selected features by using the PSO method. It is noted that all 20 features have optimal correlation with normal class. However, the features, namely, Fwd_Byts/b_Avg and Bwd_Byts/b_Avg have strongest relationship ($R = 100\%$) with normal class. Overall, all the features have good relationship with normal class.

Table 4 shows Pearson's correlation coefficient method for finding the relationship between the most significant features obtained from the PSO method with attack class. It is noted that the Fwd_PSH_Flags, Fwd_Byts/b_Avg, and Bwd_Pkts/b_Avg features obtained $R = 100\%$ whereas FIN_Flag_Cnt, RST_Flag_Cnt, CWE_Flag_Count, and ECE_Flag_Cnt features have obtained $R = 99.0\%$. We have approved that selected features by employing the PSO method were appropriated for enhancing the intrusion detection system.

2.4. Deep Learning Algorithms. In this section, the three advanced deep learning algorithms are presented: CNN, LSTM, and CNN-LSTM.

2.4.1. Convolution Neural Network. Deep neural networks are part of artificial neural networks (ANNs) with multilayers. Over the last few decades, ANNs have been

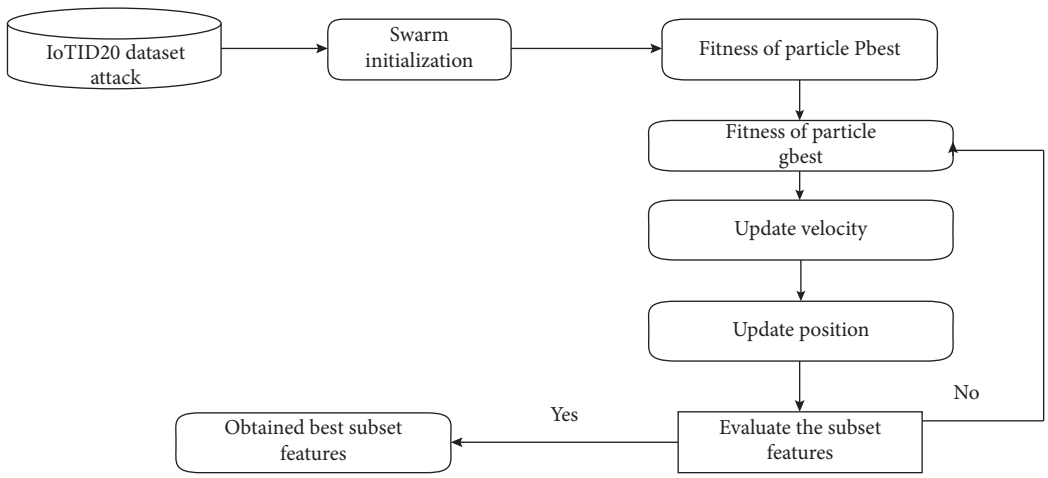

Figure 5: Particle swarm optimization algorithm steps for selecting subsets.

Table 2: 21 significant features obtained by using the PSO method.

Total features	Feature name
21	Src_IP, Fwd_Pkt_Len_Min, Flow_Pkts/s, Flow_IAT_Mean, Flow_IAT_Min, Fwd_IAT_Tot, Fwd_IAT_Mean, Bwd_IAT_Mean. 1, Bwd_IAT_Max, Bwd_IAT_Min, Fwd_PSH_Flags, FIN_Flag_Cnt, RST_Flag_Cnt, CWE_Flag_Count, ECE_Flag_Cnt, fwd_byts/b_avg, bwd_pkts/b_avg, Init_Bwd_Win_Byts, Active_Mean, Idle_Max, class

(1) Initialize parameters X_i^t is fitness, N numbers of particles
(2) Initialize population $P_{\text{i\_best}}$
a while (number of generations or the stopping criterion is not met) {
(3) for ($i = 1$ to N) {
(4) if fitness $X_i^t >$ fitness $P_{\text{i\_best}}$
(5) {
(6) then update $P_{\text{i\_best}} = X_i^t$
(7) if the fitness of $X_i^t > g_{\text{best}}$ then{
(8) then update $g_{\text{best}} = X_i^t$
(9) }
}
(10) Update velocity vector
(11) Update particle position
(12) Next particle
(13) }
(14) Next generation

Algorithm 1: PSO algorithm.

considered to be some of the most powerful algorithms for handling many real-time applications [45]. Deep learning algorithms use many deeper hidden layers to surpass classical ANN methods. [46, 47]. A convolutional neural network is one of the most popular deep neural network algorithms, and it is named convolution by using mathematical linear operation between matrices. Our proposed CNN comprised five main layers: input, convolution, polling, FC, and output. Figure 6 shows the structure of the CNN model used to develop the IoT cybersecurity system.

To extract features from cybersecurity-based IoT data, convolution layers were used. The convolution layers had multiple convolution kernels, composed of the weight of the kernels. The convolution kernel is i, the weight coefficient is indicated by w_i, and the deviation quantity is b_i. The input convolution layer is x_{i-1}, and the convolution layer was processed using equation (5).

$$x_i = f\left(w_i \otimes x_{i-1} + b_i\right), \tag{6}$$

Table 3: Correlation coefficient between features and normal class.

Features	Normal	Normal	Normal	Normal	Normal in	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Src_IP	0.55																			
Fwd_Pkt_Len_Min		0.50																		
Flow_Pkts/s			0.50																	
Flow_IAT_Mean				0.64																
Flow_IAT_Min					0.63															
Fwd_IAT_Tot						0.80														
Fwd_IAT_Mean							0.89													
Bwd_IAT_Mean								0.62												
Bwd_IAT_Max									0.62											
Bwd_IAT_Min										0.62										
Fwd_PSH_Flags											0.1									
FIN_Flag_Cnt												0.99								
RST_Flag_Cnt													0.99							
CWE_Flag_Count														0.99						
ECE_Flag_Cnt															0.99					
Fwd_Byts/b_Avg																0.1				
Bwd_Pkts/b_Avg																	0.1			
Init_Bwd_Win_Byts																		0.58		
Active_Mean																			0.1	
Idle_Max																				0.50

Table 4: Correlation coefficient between features and attack class.

Features	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack	Attack
Src_IP	0.55																			
Fwd_Pkt_Len_Min		0.50																		
Flow_Pkts/s			0.50																	
Flow_IAT_Mean				0.50																
Flow_IAT_Min					0.68															
Fwd_IAT_Tot						0.69														
Fwd_IAT_Mean							0.84													
Bwd_IAT_Mean								0.63												
Bwd_IAT_Max									0.63											
Bwd_IAT_Min										0.63										
Fwd_PSH_Flags											0.1									
FIN_Flag_Cnt												0.99								
RST_Flag_Cnt													0.99							
CWE_Flag_Count														0.99						
ECE_Flag_Cnt															0.99					
Fwd_Byts/b_Avg																0.1				
Bwd_Pkts/b_Avg																	0.1			
Init_Bwd_Win_Byts																		0.53		
Active_Mean																			0.94	
Idle_Max																				0.87

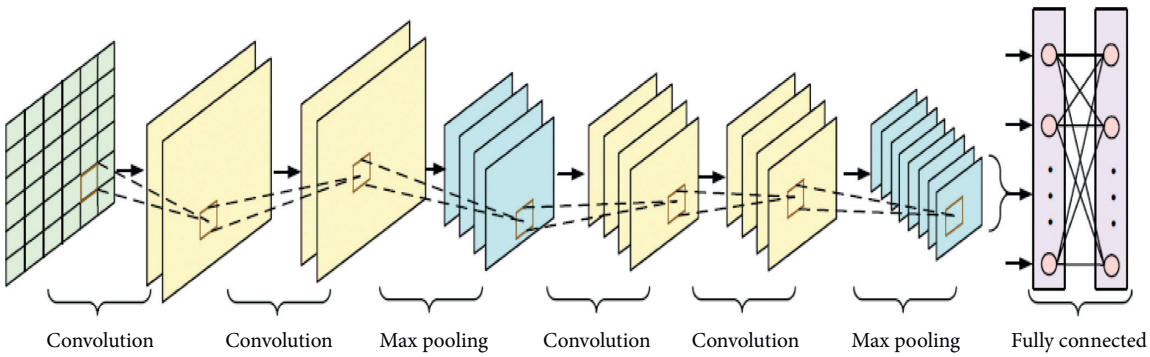

Figure 6: Structure of the convolution neural network (CNN) model for classification of Internet of Things (IoT) intrusions.

where x_i is the output convolution l, i is the convolution kernel, $\otimes$ is the convolution operation, and $f(x)$ is the activation function.

The convolution kernel was used to pass the IoT training data into max pooling for the extraction of the characteristics of the IoT network data. The extracted features were transferred into the output layer using the tanh function. It was noted that the tanh function was an appropriate activation function for designing the system.

$$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1, \tag{7}$$

where tanh is the function and x is the training input data.

$$Q_j = \text{Max}\left(P_j^0, P_j^1, P_j^2, P_j^3, \ldots, P_j^t\right), \tag{8}$$

where Q_j is the output results from the IoT cybersecurity dataset, j is the pooling region, Max is the operation, and P_j^t is the element of the pooling.

The softmax function was used to calculate the probability distribution of an N-dimensional vector. The main purpose of using softmax at the output layer was for the multiclass classification method used in machine learning algorithms, deep learning, and data science. The correct calculation of the output probability helps determine the proper target class for the input dataset, and the probabilities of the maximum values are increased using an exponential element. The softmax equation is shown in the following equation:

$$O_i = \frac{e^{z_i}}{\sum_{i=1}^{M} e^{z_i}}, \tag{9}$$

where i and z_i are the output from pervious layers, O_i indicates the output of softmax function, and M is the total number of output nodes.

2.4.2. Long Short-Term Memory Recurrent Neural Network. The recurrent neural network (RNN) is an advanced artificial intelligence algorithm used in many real-life applications. A traditional RNN was applied to predict the temporal training data, but it faced difficulties when handling gradient explosion data. To solve this issue, the LSTM model was proposed. The LSTM model used a memory function to replace the hidden RNN unit. Figure 7 displays the structure of the LSTM model for detecting intrusions from the IoT network dataset. The LSTM model consisted of three important gates: the forget, input, and output gates [48].

The forget gate was used to find forgotten information, where h_t is the input data, and the interval number of the output gate is [0, 1], where 0 indicates "completely discarded" and 1 indicates "completely retained." The current state is represented by c_t as follows:

$$\begin{aligned} h_t &= \text{sigma}\left(W_{xt} + Uh_{t-1} + b^{(h)}\right), \\ f_t &= \text{sigma}\left(W^{(f)} + X_t + U^{(f)}\, h_{t-1} + b^{(f)}\right), \end{aligned} \tag{10}$$

where h_t is input training data, and input to the previous cell is presented by h_{t-1}. The forget gate is indicated by f_t, the significant parameters of the LSTM are weight $W^{(f)}$, and $b^{(f)}$ is bias. The input gate was used to update the information using two functions, namely, sigma and tanh. The sigma function was employed to determine what information needed updating, whereas the tanh function generated information for updating.

$$\begin{aligned} i_t &= \text{sigma}\left(W^{(i)} + X_t + U^{(i)} h_{t-1} + b^{(i)}\right), \\ m_t &= \tanh\left(W^{(m)} + X_t + U^{(m)}\, h_{t-1} + b^{(m)}\right), \\ c_t &= i_t \cdot m_t + f_t \cdot c_{t-1}. \end{aligned} \tag{11}$$

When the cell state c_{t-1} is the cell state from the previous cell, which was used to update by using cell state c_t, the new information must be discarded, and f_t. c_{t-1} and i_t. m_t are combined to obtain the next cell state as follows:

$$\begin{aligned} o_t &= \text{sigma}\left(W^{(o)} + X_t + h_{t-1} + b^{(o)}\right), \\ h_t &= o_t \cdot \tanh(c_t), \end{aligned} \tag{12}$$

where o_t is the output gate and the weight vector of the neural network is represented by W, U, and V. The sigma function was used to find which information would be the output, and tanh was employed to propose the cell state and declare the final output.

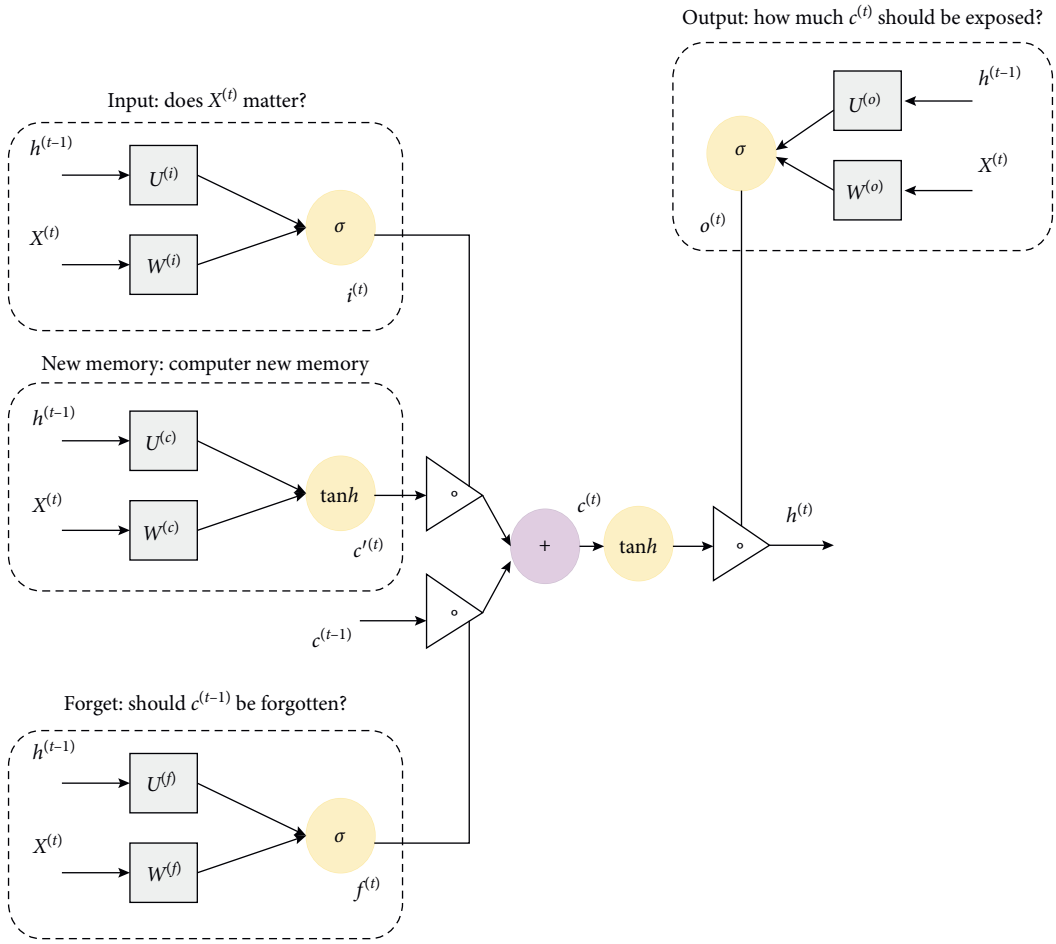

Figure 7: Generic structure of the long short-term memory (LSTM) model for the classification of Internet of Things (IoT) intrusions.

2.4.3. Combined CNN-LSTM Network. We proposed combining two advanced deep learning algorithms to detect intrusion from an IoT network dataset. A hybrid model was designed to automatically detect the attacks, and the structure of the proposed model is presented in Figure 8. The architecture was developed by combining two deep learning models, namely, the CNN and LSTM networks, whereas the CNN algorithm was used to process the significant features obtained from the PSO method with the size of $20 \times 625{,}783$ to extract new complex features. A convolutional layer size of three kernels was used to extract the complex features, and tanh activation was proposed to transfer the data. A two-kernel max pool was used for dimension reduction, and we mapped the features to the LSTM model for the extraction of new time information. After the LSTM time information was extracted, the fusion features were fully connected for use in the classification process. The softmax was proposed to detect attacks from the IoT network data.

3. Results

In this section, results of the proposed formwork for detection intrusion are presented.

3.1. Experiment Environment Setup. The proposed research was completed using different software and hardware environments. Table 5 shows the requirements used to develop the proposed system. It was noted that these requirements were suitable for training the big data.

Significant parameters used for the development of the deep learning algorithm are presented in Table 6. The kernel convolution was three, and the dropout was 50%. Moreover, the experiment epochs were 10 due to the big data. We used the tanh function for the activation function for both models.

3.2. Evaluation Metrics. Sensitivity, specificity, precision, recall, and F1-score evaluation metrics were proposed to test and evaluate the framework. The equations are defined as follows:

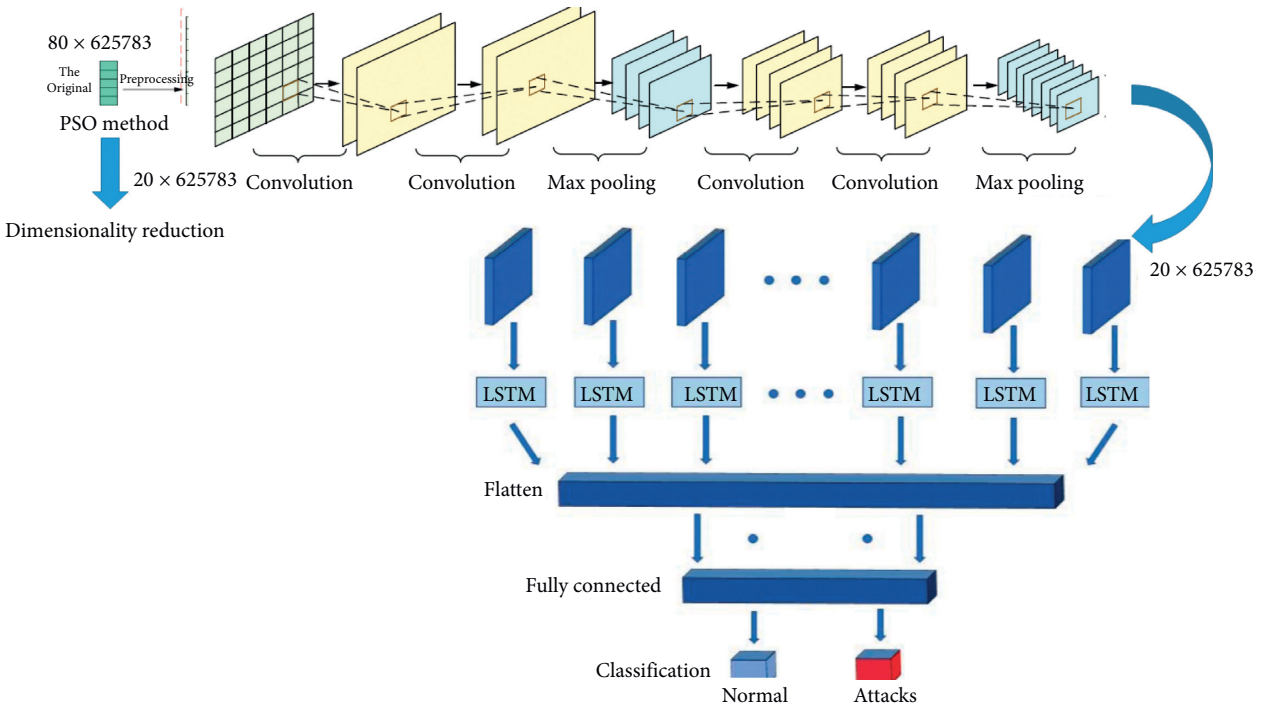

Figure 8: Architecture of the combined convolution neural network long short-term memory (CNN-LSTM) model.

$$\begin{aligned}
\text{accuracy} &= \frac{\text{TP} + \text{TN}}{\text{FP} + \text{FN} + \text{TP} + \text{TN}},\\
\text{specificity} &= \frac{\text{TN}}{\text{TN} + \text{FP}} \times 100\%,\\
\text{sensitivity} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%,\\
\text{recall} &= \text{TPTP} + \text{FN} \times 100\%,\\
\text{F1} - \text{score} &= 2 * \frac{\text{precision} * \text{Recall}}{\text{precision} * \text{Recall}} \times 100\% \text{ QUOTE Sensivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%,
\end{aligned} \tag{13}$$

where TP is true positive, FP is false positive, TN is true negative, and FN is false negative.

3.3. Results and Discussion. The experiments were conducted using a real IoT based on cybersecurity network data, and three advanced artificial intelligence models, namely, CNN, LSTM, and CNN-LSTM, were proposed to classify the attacks from the IoT network dataset. Experiments for developing a robust IoT cybersecurity system for detecting intrusions have been presented. The PSO method was applied to deal with dimensionality reduction and improve the classification process. Among the 81 features, we selected 21 as the most significant features for processing the data to detect the intrusions. It was noted that the proposed method was very robust when using the PSO method.

The numbers of false positives, false negatives, true positives, and true negatives were reported using a confusion matrix. In this research, we had to deal with big data (the total data were 625,783 instances, and the training data were 438,048 instances, whereas the total testing was 187,735 instances). Figure 9 shows the size of sample for training and testing. Table 7 shows the results of the confusion matrix obtained from the proposed system. Figure 10 shows the confusion matrix of the proposed system, and the confusion matrix of the combined CNN-LSTM model is presented in Figure 11.

To validate the proposed system, we divided the dataset into 70% training and 30% testing. Three experiments were conducted using different algorithms, namely, CNN, LSTM, and CNN-LSTM, to detect the intrusions. Table 8 demonstrates the results of the proposed model, and it was noted that the LSTM algorithm obtained a slightly higher accuracy compared with the CNN and CNN-LSTM models.

From the evaluation of the deep learning models of the two classes of normal and attacks obtained from the

Table 5: Experiment environment setup.

Hardware	Environment
Operation system	Windows 10
CPU	I7
Memory	8
Development environment	Jupyter Python 3.6

Table 6: Parameters of the proposed model.

Parameters	Value
Parameter name	Value
Convolutions filters	100
Kernel size of filter	3
Max pooling size	2
Drop out	0.50
Fully connected layer	256
Activation function	Tanh
Classification function	Softmax
Optimizer	RSMprop
Epochs	10
Batch size	5,000

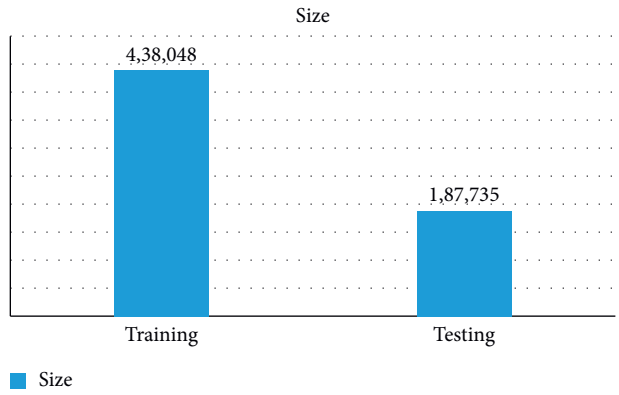

Figure 9: Size of sample for training and testing.

Table 7: Confusion matrices for the proposed framework in testing phase.

Models	TP	TN	FP	FN
CNN	171895	9512	2592	3736
LSTM	174918	9101	3003	713
CNN-LSTM	175059	9346	2758	572

confusion metrics, the empirical results for the LSTM model showed a slightly better performance: the LSTM model results were 98.84%, 99.60%, 77.72%, 99.00%, and 98.82% with respect to precision, sensitivity, specificity, F1-score, and accuracy, respectively. Overall, the deep learning algorithms achieved optimal results for detecting intrusions from the IoT network data. Figure 12 displays the training loss of the deep learning algorithms; it shows the relationship between training loss and the number of epochs in the proposed framework. It was noted that training loss gradually decreased when the training loss increased, and the proposed system of 10 epochs was suitable. The training loss and number of epochs for the combined model are presented in Figure 13.

The proposed system was validated by dividing the dataset into 30% testing, and the accuracy performances of the CNN and LSTM algorithms are presented in Figure 14. The performance of the combined CNN-LSTM model is presented in Figure 15. The three deep learning algorithms performed differently when detecting intrusions based on the IoT dataset. The CNN algorithm achieved 96% accuracy and the LSTM achieved 98% accuracy, whereas the combined CNN-LSTM model attained 98% accuracy. It was observed that the LSTM

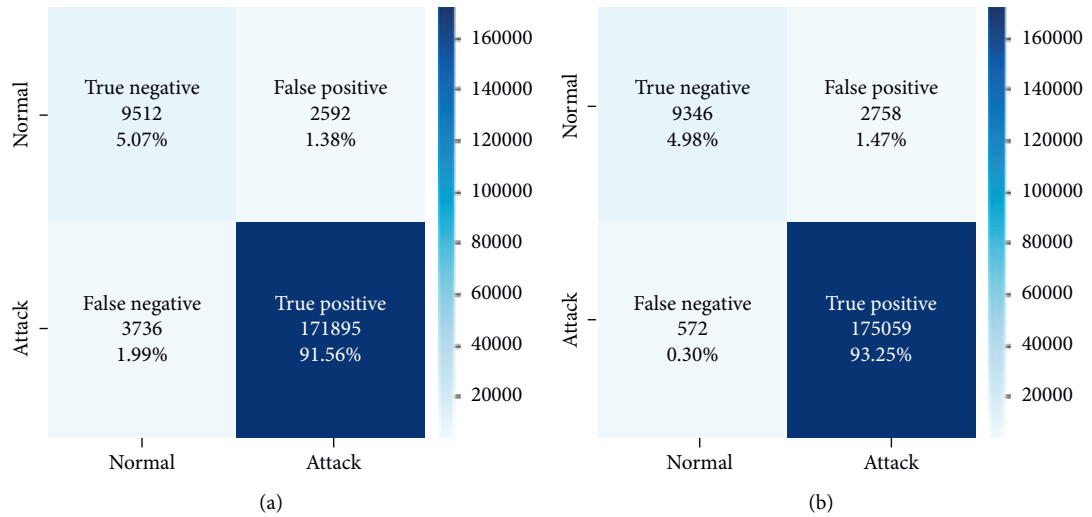

Figure 10: Confusion matrix of (a) the convolution neural network (CNN) model and (b) the long short-term memory (LSTM) model.

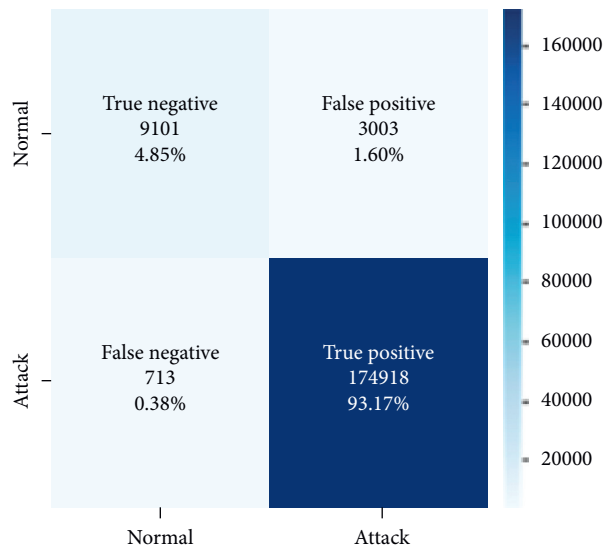

Figure 11: Confusion matrix of the convolution neural network long short-term memory (CNN-LSTM) model.

Table 8: Results of the proposed system for the validation phase.

	Precision (%)	Sensitivity (%)	Specificity (%)	F1-score (%)	Accuracy (%)	Time (second)
CNN	98.40	99.0	77.20	98.70	96.60	80
LSTM	98.0	99.70	71.60	98.90	98.20	160
CNN-LSTM	98.40	99.20	77.40	98.80	98.0	80

model was slightly better than the CNN and the combined CNN-LSTM models. Overall, it was noted that both classifications achieved better results due to the dataset having the highest dimensionality, and we found that the system was able to handle this and improve the performance of systems.

The proposed methodology was compared with research work that generated these data by Ullah et al. [49], who proposed a machine learning algorithm, namely, SVM and Gaussian Naïve bays (NB), linear discriminant analysis (LDA), and decision and random forest to detect intrusion from the IoT environment. The Shapiro–Wilk algorithm was used to select the significant features from the entire dataset, the LDA, the decision tree, the random forest, and the ensemble. It was noted that 10 features were the most significant features that enhanced the classification algorithm to attain good results. They used cross-validations 3, 5, and 10 to validate their results. Thus, we developed a system based on deep learning algorithms to improve the accuracy of detecting attacks. The PSO method was

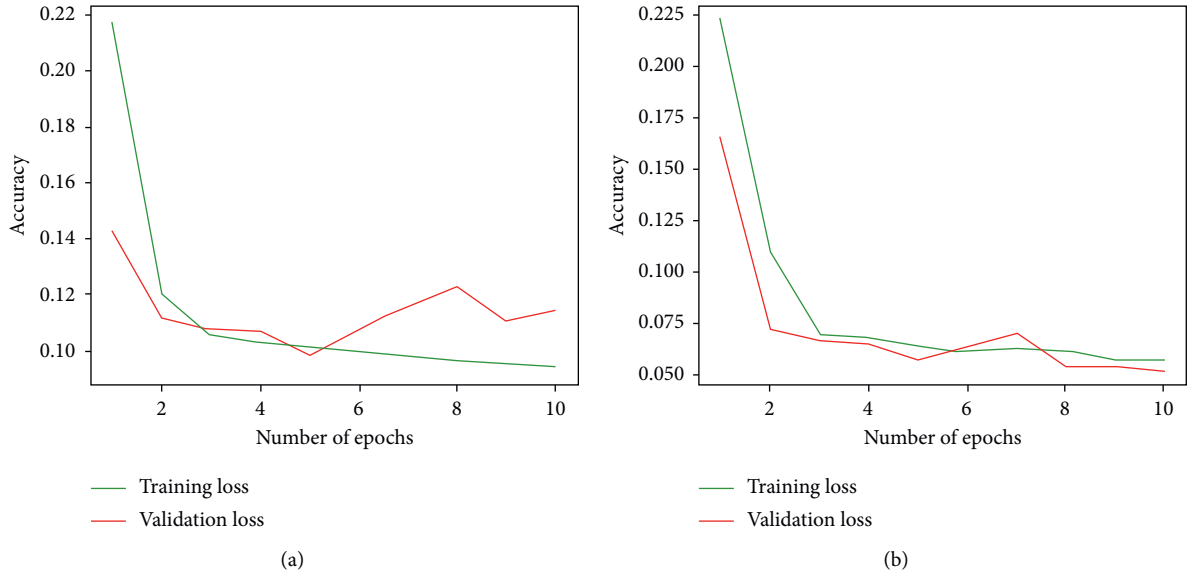

Figure 12: Training loss and epochs of (a) the convolution neural network (CNN) model and (b) the long short-term memory (LSTM) model.

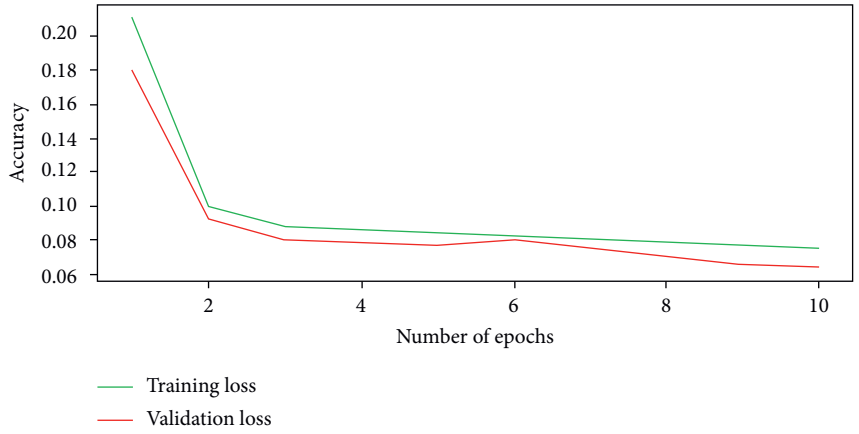

Figure 13: Training loss and number of epochs of the convolution neural network long short-term memory (CNN-LSTM) model.

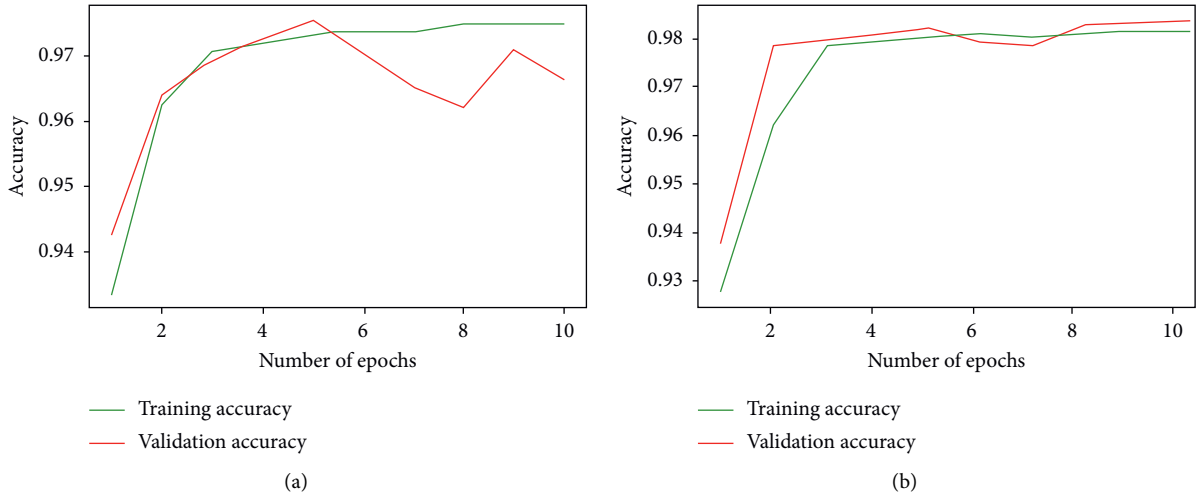

Figure 14: Performance of the proposed models: (a) convolution neural network (CNN) model and (b) long short-term memory (LSTM) model.

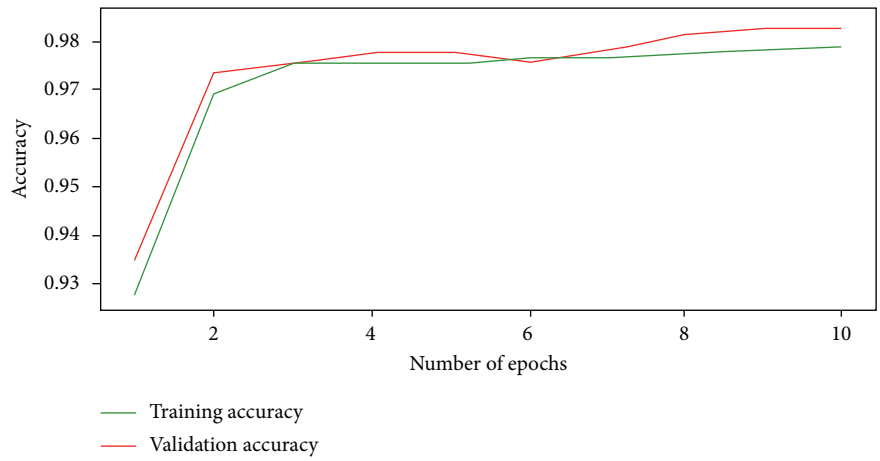

Figure 15: Performance of the proposed models: (a) convolution neural network (CNN) model and (b) long short-term memory (LSTM) model.

Table 9: Comparison of the proposed and existing model results.

Algorithms	Precision	Sensitivity	Specificity	F1-score	Accuracy	Time (second)
SVM	55	-	-	37	40	
Gaussian NB (Naïve bays)	55	-	-	62	73	
LDA	71			62	70	
Decision tree	85			88	88	
Random forest	85			84	84	
Ensemble	87			87	87	
CNN	98.40	0.990	0.772	98.70	0.966	80
LSTM	98.0	0.997	0.716	98.90	0.982	160
CNN-LSTM	98.40	0.992	0.774	98.80	0.980	80

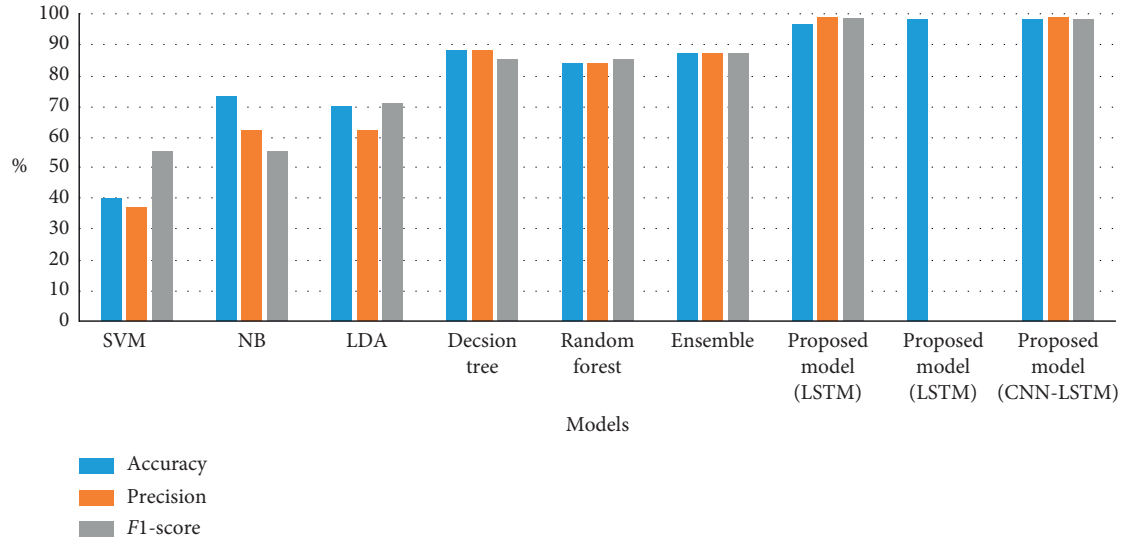

Figure 16: Comparison of the proposed system against the existing system in terms of accuracy metric.

considered to handle imbalanced data for obtaining significant subset features. We found that our system improved the effectiveness of detecting cyberattacks based on the IoT environment. Table 9 compares the performances of our proposed systems with data from previous studies. The proposed framework yielded superior detection accuracy compared with other machine algorithms (see Figure 16).

4. Conclusion

We presented the implementation and evaluation of a proposed framework to detect intrusions based on IoT infrastructure. We developed a robust system using advanced artificial intelligence algorithms, namely, CNN, LSTM, and combined CNN-LSTM. For computation intelligence, PSO was employed to derive subset features from the entire dataset. The selected subset features were processed using a classification algorithm. We made the following conclusions:

The novel proposed system was evaluated and developed using a new real standard dataset generated from the IoT environment. This was a big challenge to developing the system.

Advanced deep learning algorithms, namely, CNN, LSTM, and CNN-LSTM, were applied for the automatic classification of the intrusions.

The experimental results of the proposed system were superior to a research article that generated the dataset, and the robustness and efficiency of the proposed model will be implemented in our university IoT infrastructure.

Data Availability

The IoTID20 dataset supporting the study was obtained from Kaggle https://sites.Google.com/view/iot-network-intrusion-dataset/home. The newly developed IoTID20 dataset was adopted from Pcap files available online. The dataset contained 80 features and two main label attacks and normal. The IoTID20 dataset attack was generated in 2020. Figure 2 shows the IoT environment of the generated IoTID20 dataset. Table 1 displays all the types of IoTID20 dataset attacks, and the numbers of features for each class label are presented in Figure 4.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Faisal University for funding this research work and APC through the project number no. 206068.

References

[1] H. Alkahtani, T. H. H. Aldhyani, and M. Al-Yaari, "Adaptive anomaly detection framework model objects in cyberspace," *Applied Bionics and Biomechanics*, vol. 6660489, p. 14, 2020.

[2] T. Aldhyani and M. Joshi, "Intelligent time series model to predict bandwidth utilization," *International Journal of Advanced Computer Science and Applications*, vol. 14, pp. 130–141, 2017.

[3] M. Tang, M. Alazab, and Y. Luo, "Big data for cybersecurity: vulnerability disclosure trends and dependencies," *Institute of Electrical and Electronics Engineers Transactions on Big Data*, vol. 5, no. 3, pp. 317–329, 2019.

[4] D. Vasan, M. Alazab, S. Venkatraman, J. Akram, and Z. Qin, "MTHAEL: cross-architecture IoT malware detection based on neural network advanced ensemble learning," *Institute of Electrical and Electronics Engineers Transactions on Computers*, vol. 69, no. 11, pp. 1654–1667, 2020.

[5] A. Karim, S. Azam, B. Shanmugam, K. Kannoorpatti, and M. Alazab, "A comprehensive survey for intelligent spam email detection," *Institute of Electrical and Electronics Engineers Access*, vol. 7, pp. 168261–168295, 2019.

[6] T. H. H. Aldhyani, M. Alrasheedi, M. Y. Alzahrani, A. M. Bamhdi, A. A. Alqarni et al., "Intelligent hybrid model to enhance time series models for predicting network traffic," *Institute of Electrical and Electronics Engineers Access*, vol. 8, pp. 130431–130451, 2020.

[7] G. Press, *Internet of Things by the Numbers: What New Surveys Found*, Springer, Berlin, Germany, 2018.

[8] V. Danish, M. Alazab, W. Sobia, N. Hamad, S. Babak, and Q. Zheng, "IMCFN: Image-based malware classification using fine-tuned convolutional neural network architecture," *Computer Networks*, vol. 171, Article ID 107138, 2020.

[9] M. Alazab, K. Lakshmanna, G. Thippa Reddy, Q.-V. Pham, and P. K. R. Maddikunta, "Multi-objective cluster head selection using fitness averaged rider optimization algorithm for IoT networks in smart cities," *Sustainable Energy Technologies and Assessments*, vol. 43, 2021 ISSN 2213-1388, Article ID 100973.

[10] M. Joshi and T. H. Hadi, "A Review of Network Traffic Analysis and Prediction Techniques," p. 23, 2015, https://arxiv.org/abs/1507.05722.

[11] T. Aldhyani and M. Joshi, "Analysis of dimensionality reduction in intrusion detection," *International Journal of Computational Intelligence and Informatics*, vol. 4, no. 3, pp. 199–206, 2014.

[12] I. V. Sitalakshm and M. Alazab, "Use of data visualisation for zero-day malware detection," *Security and Communication Networks*, vol. 1728303, p. 13, 2018.

[13] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity theft detection in AMI using customers' consumption patterns," *Institute of Electrical and Electronics Engineers Transactions on Smart Grid*, vol. 7, pp. 216–226, 2017.

[14] F. A. A. Alseiari and Z. Aung, "Real-time anomaly-based distributed intrusion detection systems for advanced Metering Infrastructure utilizing stream data mining," in *Proceedings of the International Conference on Smart Grid & Clean Energy Technologies*, Offenburg, Germany, October 2015.

[15] R. Vijayanand, D. Devaraj, and B. Kannapiran, "Support vector machine based intrusion detection system with reduced input featuresfor advanced metering infrastructure of smart grid," in *Proceedings of the 4th International Conference on Advanced Computing and Communication Systems*, Coimbatore, India, January 2017.

[16] A. Jindal, A. Dua, K. Kaur, M. Singh, N. Kumar, and S. Mishra, "Decision tree and SVM-based data analytics for theft detection in smart grid," *Institute of Electrical and Electronics Engineers Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1005–1016, 2016.

[17] N. Boumkheld, M. Ghogho, and M. E. Koutbi, "Intrusion detection system for the detection of blackhole attacks in a smart grid," in *Proceedings of the 4th International Symposium on Computational and Business Intelligence*, Olten, Switzerland, September 2016.

[18] P. Jokar and V. Leung, "Intrusion detection and prevention for ZigBee-based home area networks in smart grids," *Institute of Electrical and Electronics Engineers Transactions on Smart Grid*, vol. 9, pp. 1800–1811, 2016, [CrossRef].

[19] M. N. Hasan, R. N. Toma, A.-A. Nahid, M. M. M. Islam, and J.-M. Kim, "Electricity theft detection in smart grid systems: a CNN-LSTM based approach," *Energies*, vol. 12, no. 17, p. 3310, 2019, [CrossRef].

[20] W. Wang, Y. Sheng, J. Wang et al., "HAST-IDS: learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection," *Institute of Electrical and Electronics Engineers Access*, vol. 6, pp. 1792–1806, 2018, [CrossRef].

[21] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, Karnataka, India, September 2017.

[22] A. Ullah, N. Javaid, and S. Omaji, "CNN and GRU based deep neural network for electricity theft detection to secure smart grid," in *Proceedings of the 2020 International Wireless Communications and Mobile Computing*, Limassol, Cyprus, June 2020.

[23] G. Liu and J. Zhang, "CNID: research of network intrusion detection based on convolutional neural network," *Discrete Dynamics in Nature and Society*, vol. 202011 pages, 2020, [CrossRef].

[24] Y. Xiao, C. Xing, T. Zhang, and Z. Zhao, "An intrusion detection model based on feature reduction and convolutional neural networks," *Institute of Electrical and Electronics Engineers Access*, vol. 7, pp. 42210–42219, 2019, [CrossRef].

[25] H. Yang and F. Wang, "Wireless network intrusion detection based on improved convolutional neural network," *Institute of Electrical and Electronics Engineers Access*, vol. 7, pp. 64366–64374, 2019, [CrossRef].

[26] S. S. Chakravarthi and S. Veluru, "A review on intrusion detection techniques and intrusion detection systems in MANETs," in *Proceedings of the International Conference on Computational Intelligence and Communication Networks*, Bhopal, India, November 2014.

[27] L. Santos, C. Rabadao, and R. Goncalves, "Intrusion detection systems in Internet of Things: a literature review," in *Proceedings of the 13th Iberian Conference on Information Systems and Technologies (Cisti)*, Caceres, Spain, June 2018.

[28] A. B. Mohamed, N. B. Idris, and B. Shanmugum, "A brief introduction to intrusion detection system," in *Proceedings of the Trends in Intelligent Robotics, Automation, and Manufacturing, Proceedings of the IRAM 2012*, Communications in Computer and Information Science, Kuala Lumpur, Malaysia, November 2012.

[29] S. G. Ponnambalam, J. Parkkinen, and K. C. Ramanathan, Eds., in *Proceedings of the International Conference on Intelligent Robotics, Automation, and Manufacturing*, vol. 330, Springer, Kuala Lumpur, Malaysia, November 2012.

[30] Y. Fu, Z. Yan, J. Cao, O. Koné, and X. Cao, "An automata based intrusion detection method for internet of things," *Mobile Information Systems*, vol. 2017, 2017 [CrossRef], Article ID 1750637.

[31] A. Kapitonov, S. Lonshakov, A. Krupenkin, and I. Berman, "Blockchain-based protocol of autonomous business activity for multi-agent systems consisting of UAVs," in *Proceedings of theWorkshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, pp. 84–89, [CrossRef], Linkoping, Sweden, October 2017.

[32] C. Liang, B. Shanmugam, S. Azam, M. Jonkman, F. D. Boer, and G. Narayansamy, "Intrusion detection system for internet of things based on a machine learning approach," in *Proceedings of the International Conference on Vision towards Emerging Trends in Communication and Networking (ViTE-CoN)*, pp. 1–6, [CrossRef], Vellore, India, March 2019.

[33] C. Savaglio, G. Fortino, M. Ganzha, M. Paprzycki, C. Badica, and M. Ivanovic, "Agent-based internet of things: state-of-the-art and research challenges," *Future Generation Computer Systems*, vol. 102, 2019, [CrossRef].

[34] L. Liu, B. Xu, X. Zhang, and X. Wu, "An intrusion detection method for internet of things based on suppressed fuzzy clustering," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, p. 113, 2018, [CrossRef].

[35] P. Kasinathan, G. Costamagna, H. Khaleel, C. Pastrone, and M. A. Spirito, "DEMO: an IDS framework for internet of things empowered by 6LoWPAN," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, Berlin, Germany, November 2013.

[36] J. M. R. Danda and C. Hota, "Attack identification framework for IoT devices," *Advances in Intelligent Systems and Computing. In Information Systems Design and Intelligent Applications*, Springer India, New Delhi, India, pp. 505–513, 2016.

[37] K. A. P. Da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. De Albuquerque, "Internet of Things: a survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147–157, 2019, [CrossRef].

[38] A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for Internet of Things," *Future Generation Computer Systems*, vol. 82, pp. 761–768, 2018, [CrossRef].

[39] M. A. A. Da Cruz, J. J. P. C. Rodrigues, J. Al-Muhtadi, V. V. Korotaev, and V. H. C. De Albuquerque, "A reference model for internet of things middleware," *Institute of Electrical and Electronics Engineers Internet of Things Journal*, vol. 5, no. 2, pp. 871–883, 2018, [CrossRef].

[40] A. Azmoodeh, A. Dehghantanha, and K.-K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning," *Institute of Electrical and Electronics Engineers Transactions on Sustainable Computing*, vol. 4, pp. 88–95, 2018, [CrossRef].

[41] X. Larriva-Novo, V. A. Villagrá, M. Vega-Barbas, D. Rivera, and M. Sanz Rodrigo, "An IoT-focused intrusion detection system approach based on preprocessing characterization for cybersecurity datasets," *Sensors*, vol. 21, no. 2, p. 656, 2021.

[42] J. Kennedy and R. C. Eberhart, "Particle swarm optimization,," in *Proceedings of the IEEE Int. Conf. Neural Networks*, pp. 1942–1948, Perth, Australia, November 1995.

[43] Y. Y. Chung and N. Wahid, "A hybrid network intrusion detection system using simplified swarm optimization (SSO)," *Applied Soft Computing*, vol. 12, no. 9, pp. 3014–3022, 2012.

[44] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: a review," *Applied Soft Computing*, vol. 10, no. 1, pp. 1–35, 2010.

[45] C. D. McDermott, F. Majdani, and A. V. Petrovski, "Botnet detection in the internet of things using deep learning approaches," in *Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, [CrossRef], Rio de Janeiro, Brazil, July 2018.

[46] T. H. H Aldhyani, M. Al-Yaari, H. Alkahtani, and M. Maashi, "Water quality prediction using artificial intelligence algorithms," *Applied Bionics and Biomechanics*, vol. 2020, Article ID 6659314, 2020.

[47] J. Bassey, D. Adesina, X. Li, L. Qian, A. Aved, and T. Kroecker, "Intrusion detection for IoT devices based on RF fingerprinting using deep learning," in *Proceedings of the 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*, pp. 98–104, [CrossRef], Rome, Italy, June 2019.

[48] T. Al-Mughanam, T. H. H. Aldhyani, B. Alsubari, and M. Al-Yaari, "Modeling of compressive strength of sustainable self-compacting concrete incorporating treated palm oil fuel ash using artificial neural network," *Sustainability*, vol. 12, no. 22, Article ID 9322, 2020.

[49] I. Ullah and Q. H. Mahmoud, "A scheme for generating a dataset for anomalous activity de-tection in IoT networks," in *Advances in Artificial Intelligence. Canadian AI 2020. Lecture Notes in Computer Science*, C. Goutte and X. Zhu, Eds., vol. 12109, Berlin, Germany, Springer, 2020.