IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 30, 2020, accepted May 8, 2020, date of publication May 12, 2020, date of current version May 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2994079

A Novel Attack Detection Scheme for the Industrial Internet of Things Using a Lightweight Random Neural Network

SHAHID LATIF[1], (Student Member, IEEE), ZHUO ZOU[1], (Senior Member, IEEE), ZEBA IDREES[1], AND JAWAD AHMAD[2], (Member, IEEE)

[1]State Key Laboratory of ASIC and System, Fudan University, Shanghai 200433, China
[2]School of Computing, Edinburgh Napier University, Edinburgh EH11 4DY, U.K.

Corresponding authors: Zhuo Zou (zhuo@fudan.edu.cn) and Jawad Ahmad (j.ahmad@napier.ac.uk)

This work was supported in part by NSFC under Grant 61876039 and Grant 6191101443, in part by the Shanghai Municipal Science and Technology Major Project under Grant 2018SHZDZX01, in part by ZJ Lab, and in part by the Shanghai Platform for Neuromorphic and AI Chip (NeuHeilium).

ABSTRACT The Industrial Internet of Things (IIoT) brings together many sensors, machines, industrial applications, databases, services, and people at work. The IIoT is improving our lives in several ways including smarter cities, agriculture, and e-healthcare, etc. Although the IIoT shares several characteristics with the consumer IoT, different cybersecurity mechanisms are adopted for both networks. Unlike consumer IoT solutions that are used by an individual user for a single purpose, IIoT solutions tend to be integrated into larger operational systems. As a result, IIoT security solutions require additional planning and awareness to ensure the security and privacy of the system. In this paper, different cybersecurity attacks such as denial of service (DoS), malicious operation, malicious control, data type probing, spying, scan, and wrong setup are predicted by applying machine learning techniques. To predict the aforementioned attacks, a novel lightweight random neural network (RaNN)-based prediction model has been proposed in this article. To investigate the performance of the RaNN-based prediction model, several evaluation parameters such as accuracy, precision, recall, and F1 score were calculated and compared with the traditional artificial neural network (ANN), support vector machine (SVM) and decision tree (DT). The evaluation results show that the proposed RaNN model achieves an accuracy of 99.20% for a learning rate of 0.01, with a prediction time of 34.51 milliseconds. Other performance parameters such as the precision, recall, and F1 score were 99.11%, 99.13%, and 99.20%, respectively. The proposed scheme improves the attack detection accuracy by an average of 5.65% compared to that of state-of-the-art machine learning schemes for IoT security.

INDEX TERMS Artificial neural network, cybersecurity, industrial Internet of Things, random neural network, support vector machine.

I. INTRODUCTION

The Industrial Internet of Things (IIoT) is an extension of the traditional Internet of Things (IoT) applications in the industrial sector. The IIoT enhances the capabilities of an industry to provide reliability and better efficiency in its industrial operations. In a smart manufacturing system [1], with the integration of other cyber-physical systems and modern communication technologies, the monitoring and control capabilities of an industrial system are significantly improved [2]. To understand the vision of the next generation of the industrial revolution, which is known as Industry 4.0, the concept of smart manufacturing is very important. A great number of sensors, actuators, and advanced technologies are integrated into the industrial sector. According to a recent survey, the market for IoT devices is expected to rise to 75.4 billion US dollars by 2025 [3]. In the context of the modern industry, reliability, response time, and network latency are very important factors. Considering all these factors, data transmission and decision-making technologies should be optimized without human interaction. In recent decades, the IoT has arisen as one of the most attractive research

The associate editor coordinating the review of this manuscript and approving it for publication was Haris Pervaiz.

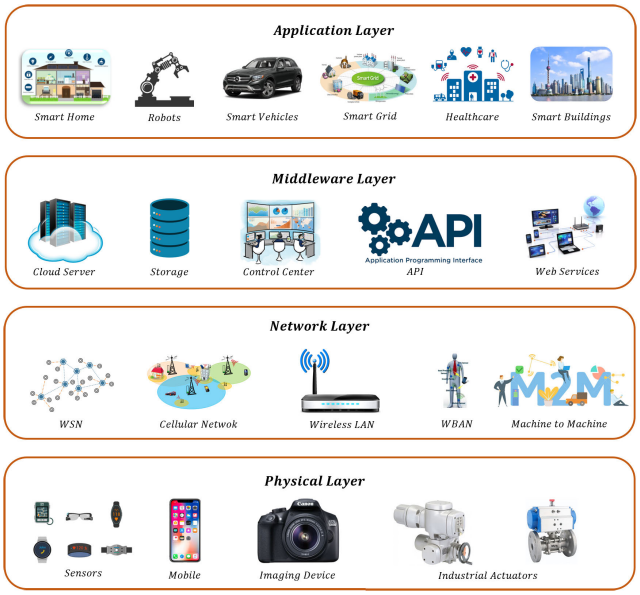


FIGURE 1. Industrial IoT system architecture.

areas; it has been widely used to interconnect unlimited consumer devices to provide facilities and ease in consumers' daily life [4], [5]. According to the vision of Industry 4.0, the utilization of the IoT in the industrial sector improves the production, efficiency, and security of industrial operations [6]. In short, the IIoT is specifically associated with the efficient use of the IoT in industrial processes. The IIoT can be briefly described as a four-layered architecture. In the industrial sector, this architecture consists of physical, network, middleware, and application layers, as shown in Figure 1. The physical layer contains a massive number of installed physical devices, sensors, mobile and computing devices, and other monitoring and automation objects. The network layer comprises several communication networks such as wireless sensor networks, cellular networks, and machine-to-machine interfaces, etc. The middleware layer provides communication between the network layer and the application layer; it has cloud storage, application programming interface, and web services. The application layer is the top layer of an IIoT architecture; it facilitates multiple industrial operations and services including smart factories, smart buildings, smart healthcare, smart vehicles, robotics, etc.

The IIoT is a complete architecture that works for many individual and industries. However, it brings many new challenges in terms of security, privacy, legal and social life. Addressing these issues requires highly scalable solutions. IoT devices are resource-constrained devices that demand security solutions that can fulfill the demands of low storage, low power, and low cost. These solutions must be compatible with standard communication protocols. IoT devices generate vast quantities of data during industrial operations, which can make an IIoT system a favorite target for attackers [7], [8]. Due to the large quantity of data, traditional data processing techniques are not suitable for IoT and IIoT applications. Therefore, machine learning (ML) is considered to be one of the most appropriate computational paradigms to provide embedded intelligence in IoT devices.

This paper presents a novel lightweight random neural network-based scheme for IIoT attack prediction. The proposed scheme detects the IIoT attacks with high accuracy and decreased prediction time by utilizing one of the latest security-related datasets. The performance of the proposed algorithm is evaluated by defining several performance parameters with varying constraints. The simulation results are compared with some other state-of-the-art machine learning classifiers. Finally, the hardware deployment of attack detection system on a single-board computer is briefly described. The rest of the article is organized as follows. Section II presents related work in IoT and IIoT security. Section III elaborates on the dataset selection, preprocessing, and the overall framework of the system. Section IV describes the theoretical aspects of the proposed random neural network. Section V discusses software and hardware implementation and analysis of the results. Section VI presents some important challenges and future research directions. The last section concludes the research.

II. RELATED WORK IN IIoT SECURITY

For large-scale industrial systems in the IIoT, efficient command and control are complex and challenging tasks. Computing platforms must be capable of processing and analyzing big industrial data in a timely and secure manner [9], [10]. Furthermore, the capacity and throughput of the system must be high to provide low latency and high reliability of data transmission. Machine learning (ML) algorithms and models have significantly improved the performance of the industrial sector in terms of reliability and security. These algorithms have great potential to address the security challenges in IIoT systems [11], [12]. In the following, some recent research works related to ML-based security schemes for the IoT and IIoT are presented.

Farahnakian and Heikkonen [13] proposed a deep autoencoder-based model for network attack detection. The researchers used the KDD-CUP 99 dataset for evaluation of their proposed scheme. A 94.71% attack detection accuracy was achieved. Their experimental results proved that the performance of their model is better than that of the deep belief network. Shone *et al.* [14] presented a nonsymmetric deep autoencoder (NDAE) that learns the features in an unsupervised manner. The authors implemented their proposed model in the graphics processing unit (GPU)-enabled TensorFlow and evaluated the model using the NSL-KDD dataset. The attack detection accuracy was 89.22%. Ali *et al.* [15] proposed a fast learning network with a combination of particle swarm optimization. The authors implemented their proposed scheme by using the KDD 99 dataset. The attack prediction accuracy of their proposed model was 98.92%. Although their model gave a satisfactory performance, the complexity of their model was high, which is not suitable for resource-constrained devices. Moukhafi *et al.* [16] proposed

a novel hybrid genetic algorithm and support vector machine with the particle swarm optimization-based scheme for DoS attack detection. The researchers implemented their proposed scheme by using the KDD 99 dataset and achieved an accuracy of 96.38%.

Vajayanand *et al.* [17] improved the classification accuracy by proposing a support vector machine (SVM)-based model. They conducted their experiments by using the ADFA-LD dataset and achieved an accuracy of 94.51%. Khalvati *et al.* [18] successfully detected and classified IoT attacks by using SVM and Bayesian. The authors implemented their model by using the KDD CUP 99 dataset and achieved an accuracy of 91.50%. James *et al.* [19] proposed a wavelet transform and deep neural network-based model to detect false data injection attacks. The researchers implemented their proposed scheme by using IEEE 118 dataset. The attack detection accuracy of their proposed model was 91.80%. Qureshi *et al.* [20] proposed an anomaly-based intrusion detection scheme. Their approach successfully detected DoS, man-in-the-middle, and SQL injection attacks in IoT and IIoT applications. The researchers evaluated their proposed scheme by using the NSL-KDD dataset, and the attack detection accuracy of their model was 91.65%. Parra *et al.* [21] proposed a cloud-based distributed deep learning framework for phishing and botnet attacks. For phishing and botnet attacks, their experimental results provided accuracy values of 94.30% and 94.80% respectively. Zheng *et al.* [22] proposed a linear discriminant analysis-based extreme learning technique for IoT intrusion detection. The researchers evaluated the accuracy of the proposed scheme by utilizing the NSL-KDD dataset. The accuracy of their approach was 92.35%.

Singh *et al.* [23] presented a comparative analysis of existing machine learning-based techniques for IoT attack detection. Ieracitano *et al.* [24] introduced an autoencoder-driven intelligent intrusion detection scheme. The researchers evaluated their scheme by utilizing the NSL-KDD dataset. Their experimental results provided better efficiency than deep and conventional shallow networks. Yan *et al.* [25] proposed a new hinge classification algorithm for cyber-attack detection. The researchers compared the performance of the proposed scheme with decision tree and logistic regression algorithms. Eskandari *et al.* [26] presented a lightweight intelligent intrusion detection scheme. The authors discussed the deployment of the scheme on IoT gateways. They successfully detected malicious traffic, port scanning, and brute force attacks using their proposed scheme. Saharkhizan *et al.* [27] proposed a hybrid IDS model for remote-to-local (R2L) and user-to-root (U2R) attacks. They successfully detected both of the attacks in IoT networks by utilizing the NSL-KDD dataset. Vinayakumar *et al.* [28] proposed a two-level deep learning framework for botnet detection. The researchers successfully categorized the attacks and normal traffic by utilizing the domain generation algorithm. Their experimental results proved the improved efficiency of their proposed scheme in terms of accuracy, F1 score, and detection speed. Ravi *et al.* [29] proposed a novel semisupervised learning algorithm for DDoS attack detection. The researchers successfully detected DDoS attacks with an accuracy of 96.28%.

In summary, most of the researchers proposed their attack detection schemes by targeting some specific applications of the IoT and IIoT. They mostly evaluated their models by using the publicly available datasets KDD Cup 99 and NSL-KDD. These datasets have been used for a long time and target specific applications of the IIoT. Therefore, according to the modern security requirements of IIoT networks, new datasets are required. One additional limitation of the described related work is that most of the researchers did not discuss the compatibility of their proposed models with resource-constrained devices. The main objective of the present research is to overcome these challenges by introducing a novel ML-based scheme for attack detection by utilizing an emerging IIoT security-related dataset. Here, a novel lightweight random neural network-based technique is proposed by utilizing one of the latest datasets, DS2OS, for attack prediction in IIoT networks.

III. FRAMEWORK OF THE SYSTEM

The infrastructure for attack detection is a combination of several processes. The attack detection mechanism is presented in Figure 2. The first step of this architecture is the dataset collection and its observation. At this stage, the dataset was collected and analyzed according to the data type. In the next step, preprocessing of the dataset was performed, which involves data cleaning, visualization, feature engineering, and vectorization. By applying all these procedures, the data features were extracted. These feature vectors were split into a training set and test set with a ratio of 80% and 20%, respectively. The training set was utilized for the learning process with the proposed random neural network. The final model was evaluated by using a test set according to different evaluation parameters.

A. DATASET DESCRIPTION

An open-source dataset named DS2OS was obtained from Kaggle [30]. This is one of the new generations of IIoT datasets for evaluating the fidelity and efficiency of different cybersecurity applications based on machine/deep learning algorithms. This dataset was provided by Pahl *et al.* [31].

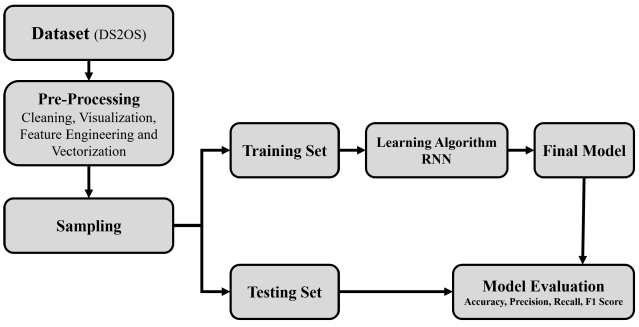


FIGURE 2. Block diagram of the attack detection mechanism.

TABLE 1. Description of attacks in the DS2OS dataset.

Attacks	Description
Denial of Service	A DoS attack is observed when a user is unable to access information systems, devices, and some network resources. This attack can affect various services including emails, websites, online banking accounts, etc. In this attack, the attacker sends too much ambiguous traffic to the network. Due to this process, the network becomes overloaded and services became unavailable to users.
Malicious Control	In MC, attackers can capture network traffic by taking advantage of software vulnerabilities. This attack can provide a user remote access to a computer, which is known as an application backdoor. The backdoor can be created with malicious intent in order to access the confidential customer or industry information.
Data Type Probing	DP occurs when a malicious node writes unwanted data type. Probing attacks are usually used to explore the detailed information of servers. These attacks gather and analyze information to map the network system.
Scan	This attack sends client requests to a range of server port addresses on a host in order to find an active port and exploit a known vulnerability of that service. The idea is to probe as many targets as possible and keep track of those that are receptive or useful to the particular need of the attacker.
Malicious Operation	MO usually happens because of malware. It is an activity that can distract the system from the original operation. The device's performance is badly affected by this attack.
Wrong Setup	In WS, an attacker can get access to important information on the system.
Spying	In spying, system vulnerabilities are exploited by an attacker, who gains access to the important information of the system by using a backdoor channel. In some cases, the attacker manipulates the data, which can be a great hazard to the whole system.
Normal	If data is completely accurate, then it will be called as normal data.

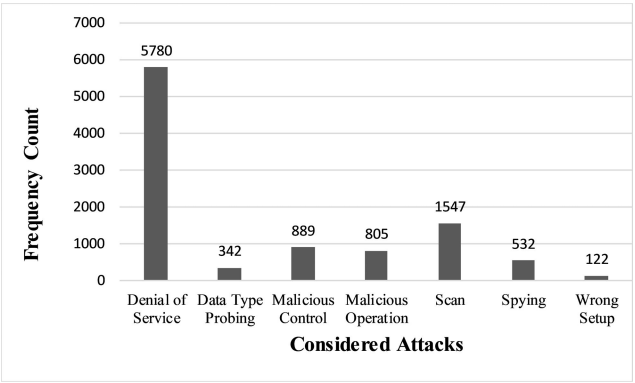


FIGURE 3. Statistics of considered attacks in the DS2OS dataset [32].

It contains attacks on sensors and applications; therefore, it provides details about several attacks and anomalies in IIoT applications including smart homes, smart factories, smart buildings, etc. The dataset consists of 357952 samples and 13 features. It has 347935 normal data values and 10017 anomalous data values, with 8 classes [32], [33]. Two features "Value" and "Accessed Node Type" have 2500 and 148 missing values, respectively. The detailed distribution of different attacks in a dataset is presented in Figure 3. All the attacks that were present in the dataset are briefly described in Table 1.

B. DATASET PREPROCESSING

Machine learning research requires good and comprehensive data analysis. The first step is to arrange data in such a configuration that they will be compatible with the input of any ML algorithm. This dataset contains missing values in two feature columns "Accessed Node Type" and "Value". The "Accessed Node Type" feature column contains 148 "NaN" values. This feature includes categorical data, so if we remove these 148 rows, then there will be a great possibility of losing some valuable information. Therefore, the "NaN" value is replaced with the "Malicious" value. Some data present in the "Value" column are also unassigned. These unexpected values are replaced with some meaningful values. True, False, Twenty, and None are replaced with 1.0, 0.0, 20.0, and 0.0, respectively.

In the next step, the first and most important task is to identify the type of features. This dataset contains numerical and categorical data. Numerical data are further classified into continuous and discrete values. Categorical data are classified into ordinal and nominal values. In the dataset, all columns contain categorical nominal variables, except "Value" and "Timestamp". These two columns consist of continuous numerical variables. The next important step is to convert categorical data into feature vectors. In this research, categorical data are converted into feature vectors via label encoding. In the dataset, most of the features consist of nominal categorical values, so the advantage of label encoding is that the number of features will remain the same. Label-encoded data are easy to fit in ML algorithms, and the processing time is less than that of one-hot encoding. In the next section, the theoretical aspects of the proposed random neural network are described.

IV. MATHEMATICAL DESCRIPTION OF THE PROPOSED RANDOM NEURAL NETWORK

The artificial neural network (ANN) brought a revolution in the field of machine learning [34]. Gelenbe introduced an advanced scheme of the ANN, which is called a random neural network (RaNN) [35]. This model is more similar to the biological neural network and can represent the transmission of human brain signals in a better way. RaNN models usually have better predictive capabilities because of their nonnegativity and probability constraints. The RaNN has a highly distributed nature; therefore, it is very suitable for deployment in resource-constrained hardware in IIoT security systems. In the RaNN model, neurons are connected in different layers. These neurons have excitation and inhibition states, which depend on the potential of a received signal. If a neuron encounters a positive signal, then it goes into an excited state, and for the negative signal, it goes into an inhibited state. The state of neuron n_i at time t is represented by $S_i(t)$. Neuron n_i will remain in an idle state until the value of $S_i(t) = 0$. To go into an excited state $S_i(t) > 0$ because $S_i(t)$ is considered a nonnegative integer. In the excited state, a neuron n_i transmits an impulse signal to another neuron n_j at the transmission rate of h_i. The transmitted signal can be received by neuron n_j as a positive signal or a negative signal with the probabilities of $p^+(i, j)$ *and* $p^-(i, j)$, respectively. Furthermore, the signal can also leave the network with a probability of $k(i)$.

Here,

$$k(i) + \sum_{j=1}^{N} p^+(i,j) + p^-(i,j) = 1, \quad \forall_i \tag{1}$$

The weights of neurons n_i and n_j are updated as

$$w^+(i, j) = h_i p^+ + (i, j) \geq 0 \tag{2}$$

and

$$w^-(i, j) = h_i p^- + (i, j) \geq 0 \tag{3}$$

In the RaNN model, the probability of the signal is determined by a Poisson distribution. Therefore, for neuron n_i, positive and negative signals are represented by the Poisson rate $\Lambda(i)$ and $\lambda(i)$, respectively, which can be mathematically described as

$$\lambda^+(j) = \sum_{j=1}^{n} e(j)\ r(j)\ p^+(j, i) + \Lambda(i) \tag{4}$$

$$\lambda^-(j) = \sum_{j=1}^{n} e(j)\ r(j)\ p^-(j, i) + \Lambda(i) \tag{5}$$

The output activation function can be described as

$$e(i) = \frac{\lambda^+(i)}{h(i) + \lambda^-(i)} \tag{6}$$

Here, the transmission rate is represented by h(i), which can be calculated by using Eq.7.

$$h(i) = (1 - k(i))^- 1 \sum_{j=1}^{N} [\, w^+(i,j) + w^-(i,j)\,] \tag{7}$$

In Eq.7, h(i) is the gain of the firing rate. During the training of the RaNN model, probabilities of positive and negative weights are updated, which can be described by Eq.8.

$$h(i) = \sum_{j=1}^{N} [\, w^+(i,j) + w^-(i,j)\,] \tag{8}$$

A. GRADIENT DESCENT ALGORITHM (GD)

The proposed RaNN-based IIoT attack detection system is trained by the gradient descent algorithm. GD is used to obtain the local minima of a function, which helps to reduce the overall mean square error [36]. This algorithm has been successfully used by various researchers for iterative optimization [37]–[39]. The error function is described in Eq.9.

$$E_p = \frac{1}{2} \sum_{i=1}^{n} \alpha_i\, (q_j^p - y_j^p)^2 \ \alpha_i \geq 0 \tag{9}$$

Here, $\alpha \in (0, 1)$ represents the state of output neuron i. The actual differential function and predicted output value are represented by q_j^p and y_j^p, respectively. Weights are updated after training the neurons a and b as $w^+(a, b)$ and $w^-(a, b)$, which are described in Eq.10 and Eq.11, respectively.

$$w_{a,b}^{+t} = w_{a,b}^{+(t-1)} - \eta \sum_{i=1}^{n} \alpha_i (q_j^p - y_j^p)\, [\, \frac{\partial q_i}{\partial w_{a,b}^+}\,]^{t-1} \tag{10}$$

Similarly,

$$w_{a,b}^{-t} = w_{a,b}^{-(t-1)} - \eta \sum_{i=1}^{n} \alpha_i (q_j^p - y_j^p)\, [\, \frac{\partial q_i}{\partial w_{a,b}^-}\,]^{t-1} \tag{11}$$

B. PROPOSED ARCHITECTURE FOR ATTACK DETECTION

In this research, the random neural network (RaNN) is the proposed technique for attack detection in IIoT systems. Like the ANN, the RaNN is also inspired by the human brain. This model contains 1 input layer, 8 hidden layers, and 1 output layer. To implement this model, the DS2OS dataset was obtained. This dataset contains a total of 13 features. Column 1 "Source ID" has no significant contribution to attack prediction. Therefore, during preprocessing, this column was removed. Column 13 is considered an output feature, which indicates the "Normality". Therefore, 11 features were used as input for the RaNN. These input features are named X1, X2 to X11. The dataset is split into a training set and test set, with a ratio of 80% and 20%, respectively. The input layer assigns weights and biasness values and forwards these data to hidden layers for further processing. Learning is very important in hidden layers because it plays a critical role in predicting the output from real features. Hidden layers transfer this information to the output layer for suitable output generation. After learning, the trained model is used to predict attacks by using a test set. The proposed scheme of the RaNN model is presented in Figure 4.

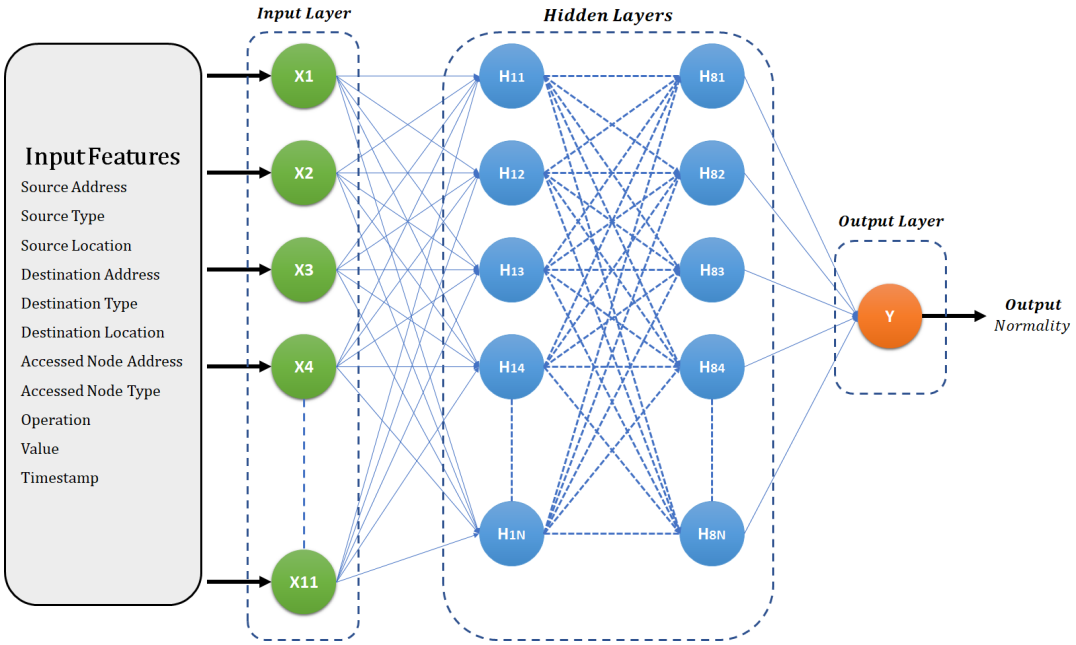


FIGURE 4. Proposed RaNN architecture for IIoT attack detection.

C. EVALUATION PARAMETERS

Several evaluation parameters were used to observe the performance of the proposed RaNN model. In the following, performance parameters that are used to evaluate the proposed algorithm are briefly explained.

1) ACCURACY

It is most preferably used performance parameter for machine learning models. Accuracy is mathematically described as the ratio between accurate positive and negative results to complete the results of the machine learning model.

$$Accuracy = \frac{T_{Pos} + T_{Neg}}{T_{Pos} + T_{Neg} + F_{Pos} + F_{Neg}} \quad (12)$$

2) PRECISION

It is a ratio between truly predicted positive results to true and false-positive results and is mathematically described in Eq.13.

$$Precision = \frac{T_{Pos}}{T_{Pos} + F_{Pos}} \quad (13)$$

3) RECALL

It describes the relation between true positive predictions to true positive and false negative predictions, as shown in Eq.14.

$$Recall = \frac{T_{Pos}}{T_{Pos} + F_{Neg}} \quad (14)$$

4) F1 SCORE

This is a weighted average of precision and recall. The F1 score maintains the balance between precision and recall by considering positive and negative results.

$$F1 - Score = \frac{2\,x\,(Precision + Recall)}{Precision + Recall} \quad (15)$$

V. IMPLEMENTATION AND RESULTS

In this section, software and hardware implementations of the proposed scheme are described in detail, and comparative analysis of the RaNN results with those of other ML classifiers is also presented.

A. SOFTWARE IMPLEMENTATION

The proposed algorithm RaNN is implemented and compared with other ML classifiers by using a Dell G5 gaming desktop computer. The system contains an Intel® Core i7-9700 processor with a processing speed of 4.7 GHz, with turbo boost technology. The installed RAM of the system was DDR4 16 GB. For the efficient and smooth running of these machine learning algorithms, an NVIDIA GeForce GTX Ti 6 GB graphics card was installed in the system. The proposed algorithm is implemented in ''Anaconda Navigator'' by using Python language.

B. SIMULATION RESULTS AND DISCUSSIONS

As described earlier, the RaNN was implemented by using the DS2OS dataset. To evaluate the performance of the proposed algorithm, several performance parameters accuracy, precision, recall, and F1 score were observed. All the results were generated and analyzed by running the simulations for 100 epochs. A neural network learns a function to best map inputs to outputs from the training dataset. The learning rate controls the learning speed of the designed model. If the learning rate is perfect, then the neural network model will

learn at its best for specific epochs. Usually, a high learning rate allows the model to perform fast learning. A low learning rate enables the model to learn more optimally, but it takes a long time for the learning process. If the learning rate is too high, then gradient descent can increase the output error. In the case of a very small learning rate, the training will be very slow and the system can also become stuck. Therefore, the selection of an accurate learning rate can ensure optimum performance of the model. The second important factor that affects the learning process of the neural network is the number of neurons in hidden layers. If the number of neurons is higher, then it can cause overfitting, and if the number of neurons is much lower than that required by the complexity of the problem, then it can cause underfitting. The correct determination of the number of neurons is very important for designing neural networks. The number of neurons in the hidden layer can be simply determined by using the rule of thumb method: the number of neurons in the hidden layers must be in the range of the size of the input and output layers.

1) ANALYSIS OF LEARNING RATES

The RaNN is a new and advanced scheme of the ANN, so in the first step, the performance of the ANN was analyzed at different learning rates. Six learning rates, 0.001, 0.005, 0.01, 0.05, 0.10 and 0.50, were selected. A comparative analysis of the training accuracy for the ANN is presented in Figure 5(a). According to this graph, the best training accuracy was achieved at a learning rate of 0.10, with a value of 98.58%. A comparison of the testing accuracy of the ANN model is shown in Figure 5(b). The best testing accuracy was achieved at the learning rate of 0.10, with a value of 98.58%. Now, the performance of the proposed RaNN is analyzed by using the same learning rates. A comparative analysis of training accuracy for the RaNN is presented in Figure 5(c). According to this graph, the overall accuracy of the RaNN algorithm is higher. The RaNN model gave the best training accuracy at the learning rate of 0.01, with a value of 99.35%. A comparison of the testing accuracy of the RaNN model is shown in Figure 5(d). This model gave the best testing accuracy at the learning rate of 0.01, with a value of 99.20%. To summarize the effect of the learning rates on all evaluation parameters, a detailed comparison of the ANN and RaNN is presented in Table 2.

2) ANALYSIS OF NUMBER OF NEURONS

In the second phase of our experiments, we analyzed the performance of both models by varying the number of neurons in the hidden layers. According to the results of the first phase, it was concluded that on average, both models gave an optimum performance at a learning rate of 0.01. Therefore, in this phase, the learning rate was fixed at 0.01, and the number of neurons was selected as 5, 10, 15, and 20. A comparative analysis of the training accuracy for the ANN model is presented in Figure 5(e). The ANN algorithm gave the best training accuracy of 98.58% with 20 neurons. A comparison of the testing accuracy is shown in Figure 5(f). According to this graph, the best testing accuracy achieved was 98.55% with 20 neurons. Next, the performance of the RaNN was analyzed for different numbers of neurons. A comparative analysis of the training accuracy for the RaNN is presented in Figure 5(g). The RaNN gave the best training accuracy of 99.36% with 15 neurons. A comparison of the testing accuracy is shown in Figure 5(h). The best testing accuracy achieved was 99.20% with 15 neurons. The effects of changing the number of neurons in a hidden layer on all performance parameters are summarized in Table 3.

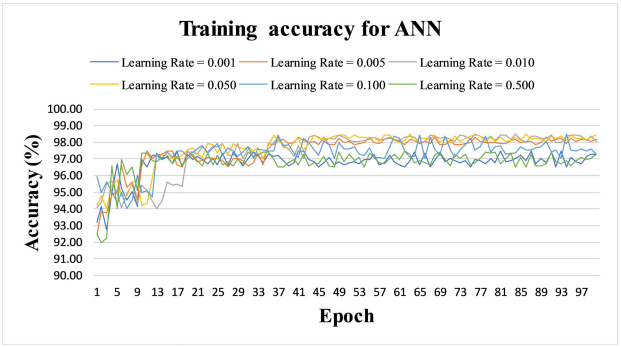
3) ANALYSIS OF PERFORMANCE OPTIMIZATION

In the above discussion, the proposed RaNN model was compared with the ANN and evaluated in terms of the training and testing accuracy with varying learning rates and numbers of neurons. By comparing the results, it was concluded that overall, both models gave satisfactory results at the learning rate of 0.01 with 15 neurons. A comparison of the best training and testing accuracy results for the ANN and RaNN models are presented in Figure 6(a) and Figure 6(b). The ANN model gave the best training accuracy of 98.58% and test accuracy of 98.55%. The RaNN model achieved a training accuracy of 99.36% and testing accuracy of 99.20%. Therefore, in terms of accuracy, the performance of the proposed RaNN model was better than that of the ANN. The accuracy performance of the proposed RaNN model with varying parameters is presented in Figure 7.

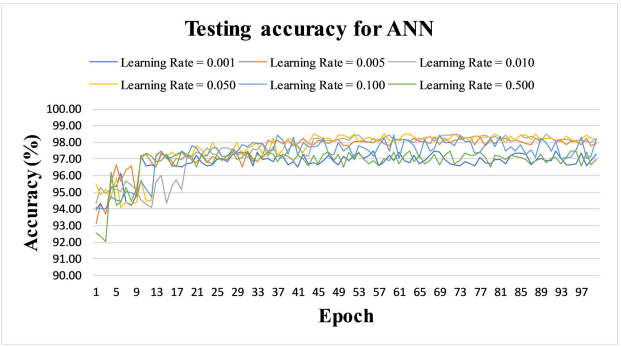
4) DISCUSSION ON ATTACK CLASSIFICATION

In the third phase of our experiments, the real and predicted results were considered for the ANN and RaNN. For the ANN model, a comparison of real and predicted values is presented in the bar graph in Figure 8(a). The ANN classified ''Data Type Probing'', ''Malicious Control'' and ''Spying'' accurately. In the ''Denial of Service'' class, 810 samples were correctly predicted from 1156 samples. The remaining 346 samples were misclassified as ''Normal'' data. Moreover, ''Malicious Operation'', ''Scan'' and ''Wrong Setup'' were misclassified by 2, 5, and 4 samples, respectively, as ''Normal'' data. In the ''Normal'' class, 357 samples were misclassified as anomalous data. Collectively, the overall performance of the ANN was satisfactory.

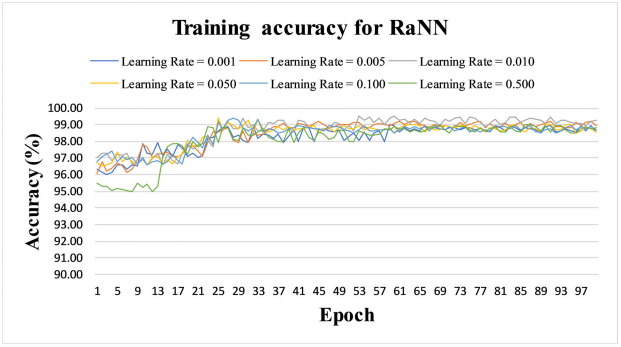
The comparison of real and predicted results for the proposed RaNN model is presented by the bar graph in Figure 8(b). According to the graph, the RaNN model accurately classified ''Data Type Probing'', ''Malicious Operation'', ''Scan'', ''Malicious Control'', and ''Spying''. In the DoS class, the RaNN algorithm misclassified 334 samples as normal samples. For ''Wrong Setup'', 3 samples of the ''Normal'' class were misclassified as WS, and from 71590 ''Normal'' samples, only 337 samples were wrongly predicted as anomalous data. Therefore, the overall performance of the RaNN was excellent, and the comparisons of both models indicate that the proposed RaNN model performed better than the ANN technique.



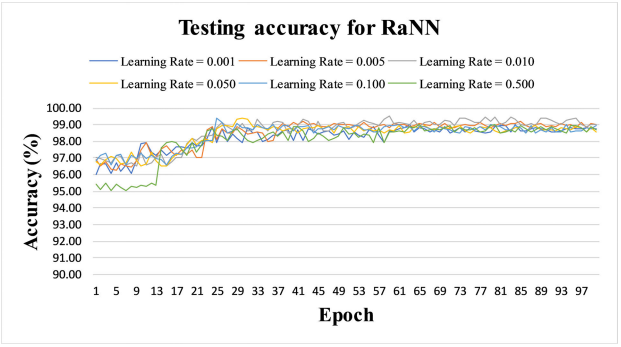
(a) Training accuracy for ANN with varying learning rates



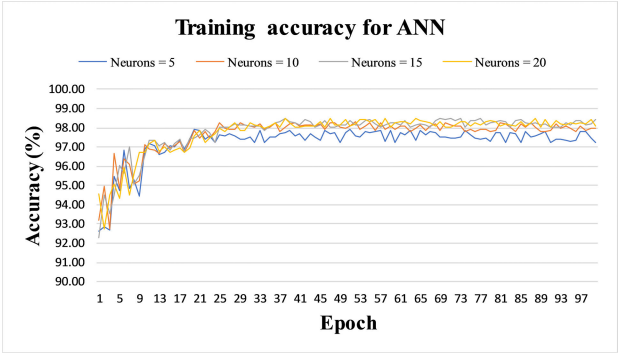
(b) Testing accuracy for ANN with varying learning rates



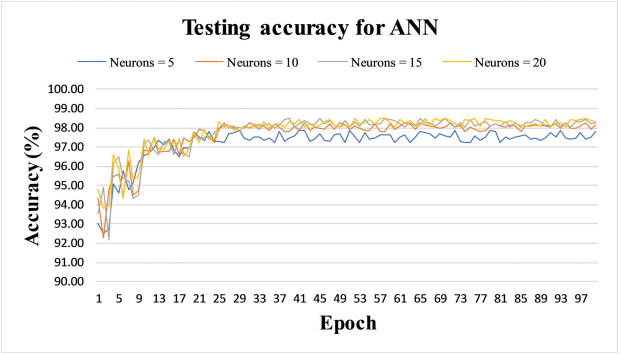
(c) Training accuracy for RaNN with varying learning rates



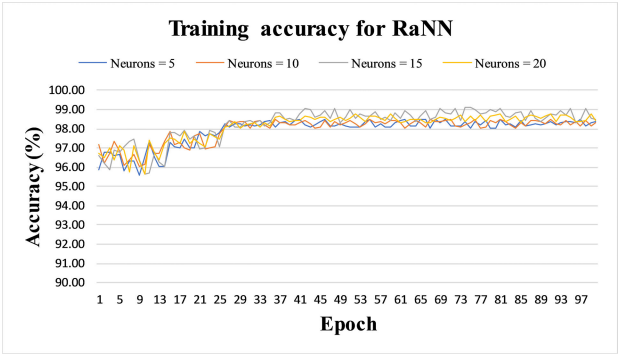
(d) Testing accuracy for RaNN with varying learning rates



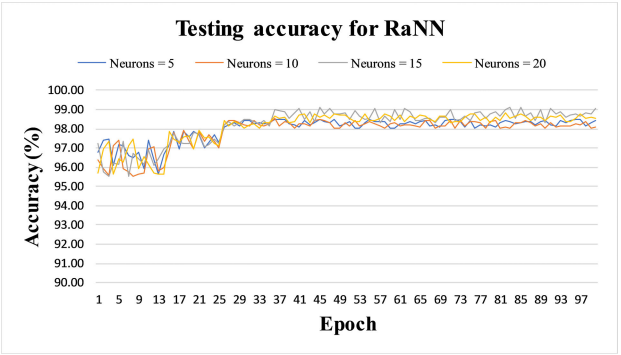
(e) Training accuracy for ANN with varying numbers of neurons



(f) Testing accuracy for ANN with varying numbers of neurons



(g) Training accuracy for RaNN with varying numbers of neurons



(h) Testing accuracy for RaNN with varying numbers of neurons

FIGURE 5. **Performance comparison of ANN and RaNN with varying learning rates and numbers of neurons.**

5) COMPARISON WITH STATE-OF-THE-ART SCHEMES

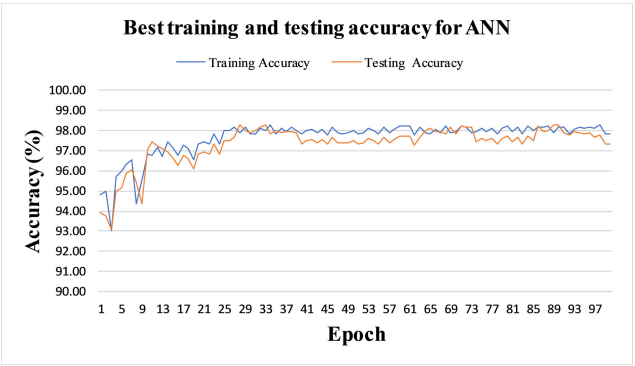
To evaluate the effectiveness and robustness of the proposed RaNN model, the performance is compared with two other classifiers, the support vector machine (SVM) and decision tree (DT). The accuracy of the SVM is low compared to that of the other classifiers. The SVM is not recommended

TABLE 2. Performance analysis of the ANN and RaNN models with varying learning rates.

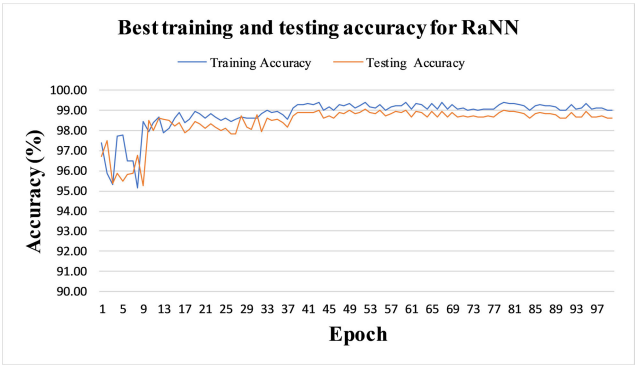
Performance Parameters		Artificial Neural Network (ANN)						Random Neural Network (RaNN)					
		Learning Rates						Learning Rates					
		0.001	0.005	0.010	0.050	0.100	0.500	0.001	0.005	0.010	0.050	0.100	0.500
Training	Accuracy	0.9770	0.9852	0.9843	0.9849	0.9858	0.9798	0.9895	0.9918	0.9935	0.9908	0.9902	0.9893
	Precision	0.9657	0.9802	0.9801	0.9833	0.9854	0.9747	0.9837	0.9895	0.9914	0.9883	0.9899	0.9892
	Recall	0.9771	0.9852	0.9843	0.9849	0.9858	0.9798	0.9805	0.9883	0.9939	0.9902	0.9914	0.9883
	F1 Score	0.9683	0.9812	0.9811	0.9835	0.9854	0.9769	0.3838	0.9918	0.9938	0.9893	0.9911	0.9891
Testing	Accuracy	0.9767	0.9850	0.9837	0.9845	0.9858	0.9794	0.9846	0.9896	0.9920	0.9906	0.9895	0.9896
	Precision	0.9655	0.9796	0.9792	0.9830	0.9875	0.9739	0.9832	0.9896	0.9908	0.9892	0.9892	0.9897
	Recall	0.9767	0.9849	0.9837	0.9845	0.9858	0.9794	0.9849	0.9897	0.9916	0.9914	0.9883	0.9883
	F1 Score	0.9678	0.9809	0.9804	0.9832	0.9854	0.9764	0.9837	0.9894	0.9904	0.9909	0.9898	0.9892

TABLE 3. Performance analysis of the ANN and RaNN models at learning rate = 0.01 with different numbers of neurons.

Performance Parameters		Artificial Neural Network (ANN)				Random Neural Network (RaNN)			
		Number of neurons in hidden layer				Number of neurons in hidden layer			
		05	10	15	20	05	10	15	20
Training	Accuracy	0.9799	0.9823	0.9849	0.9858	0.9844	0.9877	0.9936	0.9909
	Precision	0.9727	0.9766	0.9807	0.9839	0.9824	0.9850	0.9910	0.9899
	Recall	0.9799	0.9823	0.9849	0.9858	0.9838	0.9891	0.9918	0.9897
	F1 Score	0.9763	0.9785	0.9825	0.9833	0.9829	0.9849	0.9920	0.9887
Testing	Accuracy	0.9797	0.9822	0.9849	0.9855	0.9833	0.9872	0.9920	0.9887
	Precision	0.9727	0.9763	0.9807	0.9834	0.9805	0.9844	0.9911	0.9842
	Recall	0.9797	0.9822	0.9849	0.9855	0.9806	0.9899	0.9913	0.9865
	F1 Score	0.9762	0.9785	0.9825	0.9828	0.9812	0.9891	0.9920	0.9893

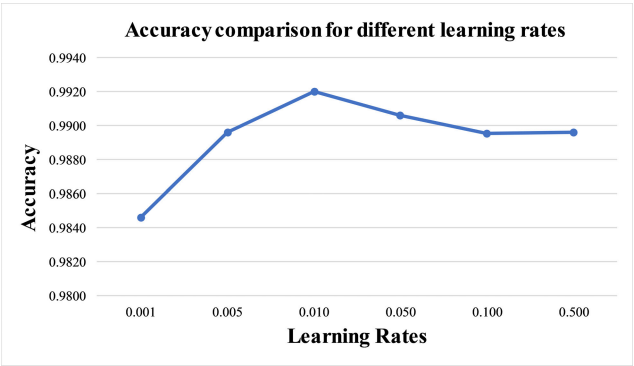


(a) Best performance of ANN

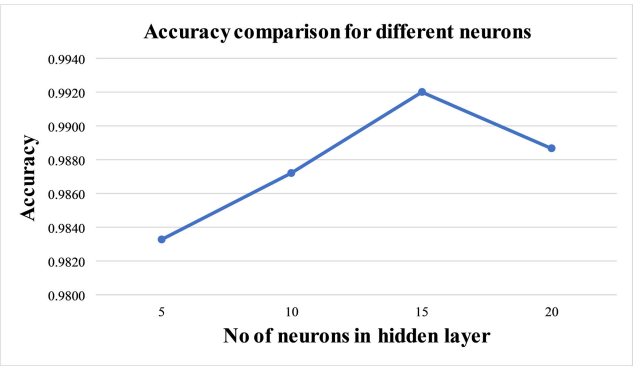


(b) Best performance of RaNN

FIGURE 6. Best performance of the ANN and RaNN for IIoT attack detection.



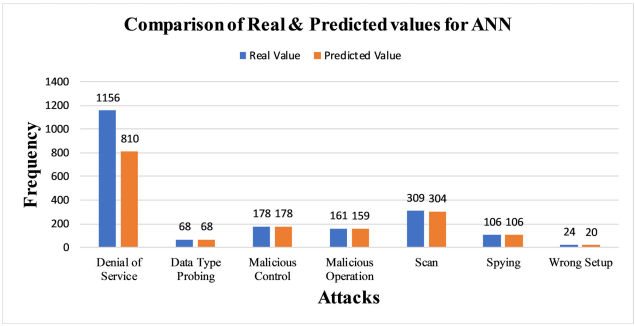
(a) Accuracy vs learning rates



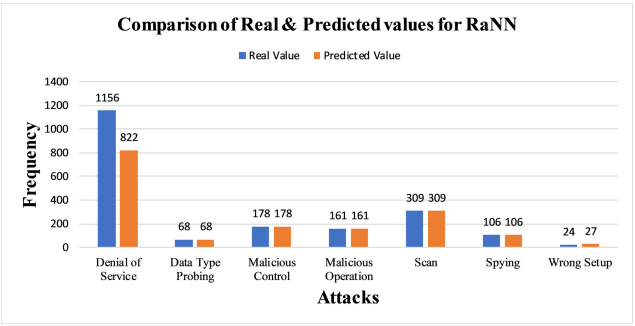
(b) Accuracy vs no. of neurons

FIGURE 7. Accuracy comparison for the proposed RaNN with varying parameters.

for large datasets because its learning time is very high; the training and test accuracies achieved were 98.31% and 98.39%, respectively. The performance of DT is better than that of the SVM and ANN. The training and test accuracy



(a) Statistics of the ANN model



(b) Statistics of the RaNN model

FIGURE 8. Statistics of attacks classification for the ANN and RaNN.

TABLE 4. Performance comparison of the SVM, DT, ANN, and RaNN.

Performance Parameters		Algorithms			
		SVM	DT	ANN	RaNN
Training	Accuracy	0.9831	0.9911	0.9858	0.9936
	Precision	0.9821	0.9912	0.9807	0.9914
	Recall	0.9810	0.9914	0.9849	0.9939
	F1 Score	0.9805	0.9910	0.9825	0.9938
Testing	Accuracy	0.9839	0.9908	0.9855	0.9920
	Precision	0.9821	0.9908	0.9808	0.9908
	Recall	0.9813	0.9911	0.9849	0.9916
	F1 Score	0.9803	0.9910	0.9825	0.9904

of DT was 99.11% and 99.08%, respectively. Compared to these classifiers, the RaNN gave the best results. A detailed comparison of the performance is presented in Table 4.

Finally, the performance comparison of the proposed RaNN with state-of-the-art security schemes is summarized in Table 5. Here, the latest proposed schemes from 2018 to 2020 were considered for comparison. Most of the researchers used the NSL-KDD and KDD-CUP 99 datasets for their studies and evaluated their models in terms of attack prediction accuracy. We have selected one of the latest IIoT security-related datasets, DS2OS. The comparison to other state-of-the-art ML classifiers indicates that the proposed RaNN provided the best attack detection accuracy.

C. PERSPECTIVE OF HARDWARE DEPLOYMENT

The proposed attack detection scheme is based on a lightweight RaNN, so it can be easily deployed on a single-board computer. One of the possible hardware deployment schemes is presented in Figure 9. Various IIoT applications such as smart cars, smart grid, smart factories, and smart home communicate with network layer with diverse communication technologies, and protocols such as Wi-Fi, Bluetooth, and wired mediums are shown. The network layer facilitates the user request to provide Internet services to individual and industrial applications. The proposed attack detection system can be integrated into a network by placing the device within the coverage area of the router and other IoT devices within the network. The attack detection system works on the transport layer and can secure both the incoming traffic and the outgoing traffic based on their placement. The proposed system does not enforce any limitations on specific network topology and can be easily integrated with different network topologies.

The recommended small, cost-effective, and hardware-friendly platform is Raspberry Pi 4B with an Intel® Neural Compute Stick 2. Raspberry Pi 4 Model B is the latest product in the popular Raspberry Pi series of computers. It offers groundbreaking increases in processor speed, multimedia performance, memory, and connectivity, as well as low power consumption. This single-board computer contains a high-performance 64-bit quad-core processor, 4 GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 5.0, Gigabit Ethernet, and USB 3.0 [40]. The Intel® Neural Compute Stick 2 (Intel® NCS2) is an embedded machine intelligence platform from Movidius, an Intel company [41]. The NCS2 is powered by the low-power Movidius Vision Processing Unit (VPU). The convenient USB stick enables developers to create, optimize, and deploy advanced deep learning techniques across a range of devices at the edge. To implement the IIoT attack detection model with Raspberry Pi4 and NCS2, a few steps must first be undertaken. Training cannot be performed directly on the neural computing stick. Therefore, first, a model of the proposed RaNN algorithm is trained by using a Dell G5 gaming desktop computer, which we used for our simulations. This trained model is converted to a deployable graph file using the SDK and NCS applications, which are provided by Intel. Next, a Python script is written that deploys the graph file and processes the operations. Finally, a Python script and graph file is written to the single-board computer Raspberry Pi 4 equipped with an Intel Neural Compute Stick [42].

The testing and training time are important factors to determine the performance of any ML classifier. The Dell G5 gaming desktop computer was a high-performance machine used for simulations. For the proposed RaNN model, the training time was 385.044 seconds, and the prediction time was 34.51 milliseconds. The overall memory usage was 447.187 MB, and the power consumption of this machine was between 410 W to 440 W during the learning process. As the main computer showed low memory usage and reduced processing time with the proposed algorithm, the RaNN model

TABLE 5. Performance comparison of the RaNN with state-of-the-art models.

Author	Dataset	Attack Detection Mechanism	Accuracy
Farahnakian et al. [13]	KDD-CUP 99	Deep Autoencoder (DAE)	94.71%
Shone et al. [14]	NSL-KDD	Nonsymmetric Deep Autoencoder	89.22%
Ali et al. [15]	KDD 99	PSO-FLN	98.92%
Moukhafi et al. [16]	KDD 99	Hybrid GA-SVM with PSO	96.38%
Vajayanand et al. [17]	ADFA-LD	Support Vector Machine (SVM)	94.51%
Khalvati et al. [18]	KDD CUP'99	SVM, Naive Bayes	91.50%
James et al. [19]	IEEE 118	Deep Neural Network	91.80%
Qureshi et al. [20]	NSL-KDD	Anomaly-based IDS	91.65%
Parraa et al. [21]	N BaIoT	LSTM Recurrent Neural Network	94.80%
Zheng et al. [22]	NSL-KDD	LDA-based extreme learning machine (ELM)	92.35%
Saharkhizan et al. [27]	NSL-KDD	Autoencoders	84.86%
Ravi et al. [29]	Own Synthetic	Learning-driven detection mitigation (LEDEM)	96.28%
Our Study	DS2OS	Lightweight Random Neural Network	99.20%

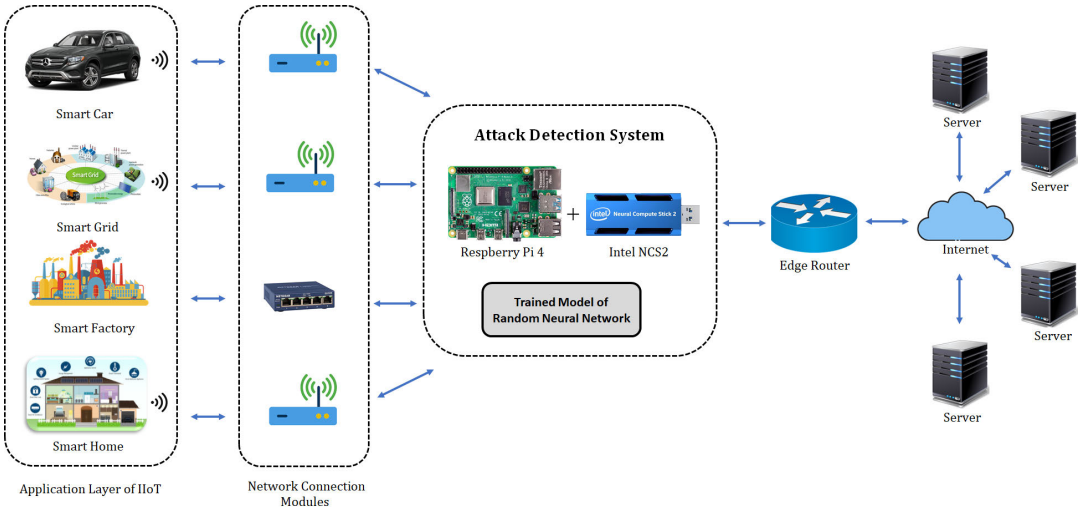


FIGURE 9. Overview of hardware deployment of the proposed attack detection scheme.

can be easily implemented on Raspberry Pi 4 with NCS2. The trained model was built on NCS2 and evaluated by using native Python script on a single-board computer. The prediction time of Raspberry Pi was 120 milliseconds. With necessary peripheral devices, the average power consumption is 2.54 W, and during the processing of algorithms, 2.84 W is consumed. Therefore, because of the low complexity and reduced resource utilization of the proposed RaNN algorithm, it can be easily implemented on a high-performance Raspberry Pi 4 single-board computer.

VI. CHALLENGES AND FUTURE RESEARCH DIRECTIONS

The latest developments in learning techniques are very helpful for the development of new ML and DL schemes to address the security challenges in IoT and IIoT networks. However, many challenges must be addressed to fulfill the complex requirements of IoT devices. In the recent era, academia and industry have shown great interest in IoT, edge, and cloud computing architectures [43]. In this direction, many security and privacy problems have arisen for IoT devices and networks. To address these challenges, a few research directions are presented here.

A. GENERATION OF NEW SECURITY-RELATED DATASETS

In IIoT systems, the generation of realistic and high-quality security-related datasets is a major challenge. The quality of the dataset is very important for evaluating the performance of ML or DL schemes. In this direction, the crowdsourcing technique can help to generate high-quality IoT and IIoT security datasets [44]. These datasets can be used for the further evaluation of the RaNN as well as newly proposed algorithms in the future.

B. IMPROVEMENTS IN EXISTING ML SCHEMES FOR LOW-QUALITY AND NOISY DATASETS

The IoT and IIoT system contains a large number of connected devices. Several constraints of these devices such as memory, power, computing capabilities usually affect the

quality of data [45]. Therefore, the improvements in existing proposed schemes and the development of new algorithms are required to deal with low-quality and noisy data. In this direction, multimodal and effective ML- and DL-based algorithms can be developed that can handle any kind of data.

C. IMPLEMENTATION OF LEARNING SCHEMES AT THE EDGE

Edge computing is an important solution that provides IoT services at the edge of the network. This approach enhances the efficiency and scalability of lightweight IoT devices [46]. Our proposed scheme can be easily implemented on a single-board computer. However, the complexity of the proposed scheme can be further reduced for implementation on several lightweight IoT devices. Therefore, implementation of ML-based solutions at the edge can help to establish an effective and secure data processing framework in the IIoT field.

D. FOG DOMAIN SECURITY

Fog computing addresses the inherent problems in cloud-based architectures such as lack of mobility support, latency, and location awareness [47]. Fog is a decentralized platform, which can make it ideal for several IoT applications. This type of computing has great capabilities of local data processing and ease of installation on heterogeneous hardware. Because of the resource limitations of fog and IoT devices, lightweight security schemes are highly desirable. The proposed lightweight RaNN model can be considered for future implementation in fog domain security for IIoT applications.

E. BLOCKCHAIN-BASED SECURE MACHINE LEARNING SCHEMES FOR IIoT SECURITY

In IoT and IIoT applications, conventional cloud-based architectures are being replaced by distributed schemes. Edge and fog computing represent revolutionary data processing techniques. These techniques provide great benefits in terms of energy efficiency, network load optimization, and latency control [43]. Due to the inherently decentralized nature of fog computing, many security threats have arisen in the fog layer and IoT devices [48]. These attacks are usually DDoS, malware, and advanced persistent threats [49]. Machine learning and blockchain are promising techniques for IIoT security. The integration of both technologies can establish a decentralized network that enables the process of decision making on a digitally encrypted platform for secure data sharing without the involvement of any third party [50]. In the future, the proposed scheme can be integrated with blockchain technology to develop a robust security mechanism for IIoT networks.

VII. CONCLUSION

In this paper, a novel lightweight RaNN-based approach has been proposed for the detection of numerous attacks and anomalies in Industrial IoT systems. Attacks classified in this research were denial of service (DoS), malicious operation, malicious control, data type probing, spying, scan, and wrong setup attacks. Compared to other methods, the proposed RaNN accurately detects the aforementioned attacks with a higher accuracy of more than 99% and a prediction time of 34.51 milliseconds. The best results for the RaNN were with a learning rate of 0.01. The accuracy of the proposed RaNN-based prediction was higher than that of other machine learning algorithms such as the ANN, SVM, and DT. Additionally, the values of other parameters such as precision, recall, and F1 score were higher for the proposed RaNN model. This paper also discussed possible hardware deployment of the attack detection system. A Raspberry Pi 4 and Intel Neural Computing Stick-based architecture is suggested for IIoT attack detection at the edge. The proposed model is tested only on a single open-source dataset known as DS2OS. However, more detailed experiments can be conducted according to the described future direction to further validate the accuracy and feasibility of the proposed model. In the future, more detailed and real-time experiments will be conducted on the proposed RaNN-based model.

REFERENCES

[1] S. Jeong, W. Na, J. Kim, and S. Cho, "Internet of Things for smart manufacturing system: Trust issues in resource allocation," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4418–4427, Dec. 2018.

[2] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.

[3] L. Columbus. (2016). *Roundup of Internet of Things Forecasts and Market Estimates, 2016*. [Online]. Available: https://www.forbes.com/sites/louiscolumbus/2016/11/27/roundup-of-internet-of-things-forecasts-and-market-estimates-2016/#a121f9e292d5

[4] R. Yunus, O. Arif, H. Afzal, M. F. Amjad, H. Abbas, H. N. Bokhari, S. T. Haider, N. Zafar, and R. Nawaz, "A framework to estimate the nutritional value of food in real time using deep learning techniques," *IEEE Access*, vol. 7, pp. 2643–2652, 2019.

[5] F. Anwaar, N. Iltaf, H. Afzal, and R. Nawaz, "HRS-CE: A hybrid framework to integrate content embeddings in recommender systems for cold start items," *J. Comput. Sci.*, vol. 29, pp. 9–18, Nov. 2018.

[6] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," *Manuf. Lett.*, vol. 3, pp. 18–23, Jan. 2015.

[7] Z. Ling, J. Luo, Y. Xu, C. Gao, K. Wu, and X. Fu, "Security vulnerabilities of Internet of Things: A case study of the smart plug system," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1899–1909, Dec. 2017.

[8] S.-U. Hassan, M. Shabbir, S. Iqbal, A. Said, F. Kamiran, R. Nawaz, and U. Saif, "Leveraging deep learning and SNA approaches for smart city policing in the developing world," *Int. J. Inf. Manage.*, early access, Nov. 30, 2019, doi: 10.1016/j.ijinfomgt.2019.102045.

[9] J. Chen, J. Wu, H. Liang, S. Mumtaz, J. Li, K. Konstantin, A. K. Bashir, and R. Nawaz, "Collaborative trust blockchain based unbiased control transfer mechanism for industrial automation," *IEEE Trans. Ind. Appl.*, early access, Dec. 13, 2019, doi: 10.1109/TIA.2019.2959550.

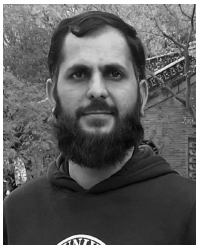
[10] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24411–24432, 2018.

[11] M. Lavassani, S. Forsström, U. Jennehag, and T. Zhang, "Combining fog computing with sensor mote machine learning for industrial IoT," *Sensors*, vol. 18, no. 5, p. 1532, 2018.

[12] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A tensor-train deep computation model for industry informatics big data feature learning," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3197–3204, Jul. 2018.

[13] F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," in *Proc. 20th Int. Conf. Adv. Commun. Technol. (ICACT)*, Feb. 2018, pp. 178–183.

[14] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 1, pp. 41–50, Feb. 2018.
[15] M. H. Ali, B. A. D. Al Mohammed, A. Ismail, and M. F. Zolkipli, "A new intrusion detection system based on fast learning network and particle swarm optimization," *IEEE Access*, vol. 6, pp. 20255–20261, 2018.
[16] M. Moukhafi, K. El Yassini, and S. Bri, "A novel hybrid GA and SVM with PSO feature selection for intrusion detection system," *Int. J. Adv. Sci. Res. Eng.*, vol. 4, no. 5, pp. 129–134, 2018.
[17] R. Vijayanand, D. Devaraj, and B. Kannapiran, "A novel intrusion detection system for wireless mesh network with hybrid feature selection technique based on GA and MI," *J. Intell. Fuzzy Syst.*, vol. 34, no. 3, pp. 1243–1250, Mar. 2018.
[18] L. khalvati, M. Keshtgary, and N. Rikhtegar, "Intrusion detection based on a novel hybrid learning approach," *J. AI Data Mining*, vol. 6, no. 1, pp. 157–162, 2018.
[19] J. J. Q. Yu, Y. Hou, and V. O. K. Li, "Online false data injection attack detection with wavelet transform and deep neural networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3271–3280, Jul. 2018.
[20] H. Larijani, A. Javed, N. Mtetwa, and J. Ahmad, "Intrusion detection using swarm intelligence," in *Proc. UK/China Emerg. Technol. (UCET)*, Aug. 2019, pp. 1–5.
[21] G. D. L. T. Parra, P. Rad, K.-K. R. Choo, and N. Beebe, "Detecting Internet of Things attacks using distributed deep learning," *J. Netw. Comput. Appl.*, vol. 163, Sep. 2020, Art. no. 102662.
[22] D. Zheng, Z. Hong, N. Wang, and P. Chen, "An improved LDA-based ELM classification for intrusion detection algorithm in IoT application," *Sensors*, vol. 20, no. 6, p. 1706, 2020.
[23] T. Singh and N. Kumar, "Machine learning models for intrusion detection in IoT environment: A comprehensive review," *Comput. Commun.*, early access, Feb. 26, 2020, doi: 10.1016/j.comcom.2020.02.001.
[24] C. Ieracitano, A. Adeel, F. C. Morabito, and A. Hussain, "A novel statistical analysis and autoencoder driven intelligent intrusion detection approach," *Neurocomputing*, vol. 387, pp. 51–62, Apr. 2020.
[25] X. Yan, Y. Xu, X. Xing, B. Cui, Z. Guo, and T. Guo, "Trustworthy network anomaly detection based on an adaptive learning rate and momentum in IIoT," *IEEE Trans. Ind. Informat.*, early access, Feb. 20, 2020, doi: 10.1109/TII.2020.2975227.
[26] M. Eskandari, Z. H. Janjua, M. Vecchio, and F. Antonelli, "Passban IDS: An intelligent anomaly based intrusion detection system for IoT edge devices," *IEEE Internet Things J.*, early access, Jan. 30, 2020, doi: 10.1109/JIOT.2020.2970501.
[27] M. Saharkhizan, A. Azmoodeh, H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, and G. Srivastava, "A hybrid deep generative local metric learning method for intrusion detection," in *Handbook of Big Data Privacy*. Cham, Switzerland: Springer, 2020, pp. 343–357.
[28] R. Vinayakumar, M. Alazab, S. Srinivasan, Q.-V. Pham, S. K. Padannayil, and K. Simran, "A visualized botnet detection system based deep learning for the Internet of Things networks of smart cities," *IEEE Trans. Ind. Appl.*, early access, Feb. 6, 2020, doi: 10.1109/TIA.2020.2971952.
[29] N. Ravi and S. M. Shalinie, "Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3559–3570, Apr. 2020.
[30] M.-O. Pahl and F.-X. Aubet. (2018). *Ds2Os Traffic Traces IoT Traffic Traces Gathered in a The Ds2Os IoT Environment*. [Online]. Available: https://www.kaggle.com/francoisxa/ds2ostraffictraces
[31] M.-O. Pahl and F.-X. Aubet, "All eyes on you: Distributed multi-dimensional IoT microservice anomaly detection," in *Proc. 14th Int. Conf. Netw. Service Manage. (CNSM)*, Nov. 2018, pp. 72–80.
[32] M. Hasan, M. M. Islam, M. I. I. Zarif, and M. M. A. Hashem, "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches," *Internet Things*, vol. 7, Sep. 2019, Art. no. 100059.
[33] O. Brun, Y. Yin, and E. Gelenbe, "Deep learning with dense random neural network for detecting attacks against IoT-connected home environments," *Procedia Comput. Sci.*, vol. 134, pp. 458–463, Jan. 2018.
[34] G. M. Khan, "Artificial neural network (ANNs)," in *Evolution of Artificial Neural Development*. Cham, Switzerland: Springer, 2018, pp. 39–55.
[35] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution," *Neural Comput.*, vol. 1, no. 4, pp. 502–510, Dec. 1989.
[36] J. Ahmad, H. Larijani, R. Emmanuel, M. Mannion, A. Javed, and M. Phillipson, "Energy demand prediction through novel random neural network predictor for large non-domestic buildings," in *Proc. Annu. IEEE Int. Syst. Conf. (SysCon)*, Apr. 2017, pp. 1–6.
[37] A. Tahir, J. Ahmad, G. Morison, H. Larijani, R. M. Gibson, and D. A. Skelton, "Hrnn4F: Hybrid deep random neural network for multi-channel fall activity detection," *Probab. Eng. Informational Sci.*, pp. 1–14, Aug. 2019.
[38] A.-U.-H. Qureshi, H. Larijani, N. Mtetwa, A. Javed, and J. Ahmad, "RNN-ABC: A new swarm optimization based technique for anomaly detection," *Computers*, vol. 8, no. 3, p. 59, 2019.
[39] H. Larijani, J. Ahmad, and N. Mtetwa, "A heuristic intrusion detection system for Internet-of-Things (IoT)," in *Intelligent Computing*. Cham, Switzerland: Springer, pp. 86–98.
[40] Raspberry Pi Foundation. (2019). *Raspberry Pi 4 Model B*. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-4-model-b/
[41] Intel. (2018). *Intel Neural Compute Stick 2*. [Online]. Available: https://software.intel.com/en-us/neural-compute-stick
[42] A. Herrold and N. Smith. (2019). *Raspberry Pi 4 and Intel Neural Compute Stick 2 Setup*. [Online]. Available: https://software.intel.com/en-us/articles/raspberry-pi-4-and-intel-neural-compute-stick-2-setup
[43] S. S. Gill, S. Tuli, M. Xu, I. Singh, K. V. Singh, D. Lindsay, S. Tuli, D. Smirnova, M. Singh, U. Jain, H. Pervaiz, B. Sehgal, S. S. Kaila, S. Misra, M. S. Aslanpour, H. Mehta, V. Stankovski, and P. Garraghan, "Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges," *Internet Things*, vol. 8, Dec. 2019, Art. no. 100118.
[44] V. Sharma, I. You, D. N. K. Jayakody, and M. Atiquzzaman, "Cooperative trust relaying and privacy preservation via edge-crowdsourcing in social Internet of Things," *Future Gener. Comput. Syst.*, vol. 92, pp. 758–776, Mar. 2019.
[45] A. S. Lalos, A. P. Kalogeras, C. Koulamas, C. Tselios, C. Alexakos, and D. Serpanos, "Secure and safe IIoT systems via machine and deep learning approaches," in *Security and Quality in Cyber-Physical Systems Engineering*. Cham, Switzerland: Springer, 2019, pp. 443–470.
[46] F. Liang, W. Yu, X. Liu, D. Griffith, and N. Golmie, "Towards edge-based deep learning in industrial Internet of Things," *IEEE Internet Things J.*, early access, Jan. 1, 2020, doi: 10.1109/JIOT.2019.2963635.
[47] K. Kaur and M. Sachdeva, "Fog computing in IoT: An overview of new opportunities," in *Proc. ICETIT*. Cham, Switzerland: Springer, 2019, pp. 59–68.
[48] S. S. Gill and R. Buyya, "SECURE: Self-protection approach in cloud resource management," *IEEE Cloud Comput.*, vol. 5, no. 1, pp. 60–72, Jan. 2018.
[49] S. Feng, Z. Xiong, D. Niyato, and P. Wang, "Dynamic resource management to defend against advanced persistent threats in fog computing: A game theoretic approach," *IEEE Trans. Cloud Comput.*, early access, Jan. 31, 2019, doi: 10.1109/TCC.2019.2896632.
[50] J. Wan, J. Li, M. Imran, D. Li, and Fazal-e-Amin, "A blockchain-based solution for enhancing security and privacy in smart factory," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3652–3660, Jun. 2019.



SHAHID LATIF (Student Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from HITEC University, Taxila, Pakistan, in 2013 and 2018, respectively. He is currently pursuing the Ph.D. degree with the School of Information Science and Engineering, Fudan University, Shanghai, China. He has served as a Lecturer at the Department of Electrical Engineering, HITEC University, from 2015 to 2019. During his teaching carrier, he has supervised several projects in the field of Electronics, Embedded systems, Control Systems, and the Internet of Things. He is currently working in the research area of cybersecurity of the Industrial Internet of Things (IIoT).

ZHUO ZOU (Senior Member, IEEE) received the Ph.D. degree in electronic and computer systems from the KTH Royal Institute of Technology, Sweden, in 2012. He is currently with Fudan University, Shanghai, as a Professor, where he is conducting research on integrated circuits and systems for IoT and ubiquitous intelligence. Prior to joining Fudan, he was an Assistant Director and a Project Leader at the VINN iPack Excellence Center, KTH, where he coordinated the research project on ultra-low-power embedded electronics for wireless sensing. He has been an Adjunct Professor and Docent with the University of Turku, Finland. He is also the Vice Chairman of IFIP WG-8.12.

ZEBA IDREES received the bachelor's degree in telecommunication engineering from the Government College University Faisalabad, Pakistan, and the master's degree in electrical engineering from the National University of Sciences and Technology (NUST), Islamabad, Pakistan, in 2012 and 2014, respectively. She is currently pursuing the Ph.D. degree with the School of Information Science and Engineering, Fudan University, China. She possesses professional and research experience of more than five years in academia as well as industry. She was with FAST National University as a Lecturer for one year. She has been serving as a Lecturer at the Electrical Engineering Department, UET Lahore–Faisalabad, since 2015. Her current research interests include electronic circuits, wireless sensors networks and systems for ambient intelligence, cognitive radio networks, and the Internet of Things.

JAWAD AHMAD (Member, IEEE) is currently an Experienced Researcher with more than ten years of cutting-edge research and teaching experience in prestigious institutes, including Edinburgh Napier University, U.K., Glasgow Caledonian University, U.K., Hongik University, South Korea, and HITEC University, Taxila, Pakistan. He has coauthored more than 50 research articles, in international journals and peer-reviewed international conference proceedings. He has taught various courses both at Undergraduate (UG) and Postgraduate (PG) levels during his career. He regularly organizes timely special sessions and workshops for several flagship IEEE conferences. He is an invited reviewer for numerous world-leading high-impact journals (reviewed more than 50 journal articles to date). His research interests include cybersecurity, multimedia encryption, machine learning, and application of chaos theory in cybersecurity.