An Intrusion Detection System for Edge-Envisioned Smart Agriculture in Extreme Environment

Danish Javeed, *Student Member, IEEE*, Tianhan Gao, Muhammad Shahid Saeed, and Prabhat Kumar, *Member, IEEE*

***Abstract*—The deployment of Internet of Things (IoT) systems in smart agriculture (SA) operates in extreme environments, including wind, snowfall, flooding, landscape, and so on for collecting and processing real-time data. The increased connectivity and broad adoption of IoT devices with low-power communications on farmland support farmers in making data-driven decisions using various artificial intelligence (AI) techniques. Furthermore, in such an environment, edge computing is also utilized to provide computationally intensive, latency-sensitive, and bandwidth-demanding services at the edge of the network. However, protecting edge-to-Things in the extreme environment of SA is challenging, due to the volume of data, and also attackers exploit network gateways to perform distributed denial of service (DDoS) attacks. Motivated by the aforementioned challenges, we develop a novel deep learning (DL)-based intrusion detection system (IDS) for edge-envisioned SA in extreme environments. Specifically, a hybrid approach is developed by combining bidirectional gated recurrent unit, long-short-term memory with softmax classifier to detect attacks at the edge of the network. To allow faster learning, the proposed IDS employs the truncated backpropagation through time (TBPTT) approach to handle lengthy sequences of network data. Furthermore, we suggest an attack scenario with deployment architecture for the proposed IDS in the extreme environment of SA. Extensive experiments using three publicly available datasets, namely, CIC-IDS2018, ToN-IoT, and Edge-IIoTset prove the effectiveness of the proposed IDS over some traditional and contemporary state-of-the-art techniques.**

***Index Terms*—Edge computing, extreme environment, Internet of Things (IoT), intrusion detection system (IDS), smart agriculture (SA).**

I. Introduction

THE Internet of Things (IoT) has surprisingly revolutionized the conventional methods of network communication by enabling synchronized connectivity among heterogeneous nodes. The flourishing circle of IoT applications endorses its positive potential toward efficient communications that have led to a new spectrum of scientific innovations [1], [2]. The term smart agriculture (SA) refers to a new approach to agricultural production that integrates information and data technologies with quantitative decision making and intelligent control to improve the productivity and quality of farming. On the other hand, the integration and deployment of IoT in SA, such as water, soil, and air is mostly affected by various extreme environments, including wind, snowfall, flooding, and landscape. However, such deployments are the backbone in various use cases of SA, including, water quality monitoring, precision agriculture, livestock monitoring, climate condition monitoring, and so on for collecting and processing real-time data [3].

Edge computing is considered a remarkable choice to achieve efficient communication for IoTs deployed in the extreme environment of SA. The main aim of edge computing is to process the data at the nearest place where it was actually generated [4]. The edge-to-Things reduces the unnecessary flow of data over the network, which directly results in reduced latency and less consumption of network resources. In scalable, centralized extreme environments, the data needs to be stored at a central location for monitoring and reuse purposes [5]. Edge computing may also serve in this regard by storing the data at a suitable local location accessible to all participant nodes. In the extreme environment of SA, edge computing can also improve communicational efficiency, load management, and network uptime, which will be a remarkable milestone toward more reliable operations [6].

The involvement of heterogeneous sensors in SA communication introduces a variety of security concerns. Especially, in large-scale networks, the abundance of heterogeneous sensors may offer welcoming environments for intruders to shake the integrity of the whole system. In an edge-to-Things system, suitable communication infrastructure is the core element that ensures smooth communication [7]. The presence of malicious entities within the network can manipulate the entire communication infrastructure, resulting in random communication [6]. Such challenging circumstances demand some favorable security solutions to combat emerging security challenges [8]. A competent intrusion detection system (IDS) can diminish the risk of attacks by timely detecting malicious entities present in the network. Over the past decade, deep learning (DL)-based IDS have gained considerable momentum because of their quick response against anomalies [9]. Second, DL-based IDS provides more meticulous results compared to machine learning techniques. In DL-driven IDS, the model

Manuscript received 12 May 2023; revised 7 June 2023; accepted 20 June 2023. Date of publication 22 June 2023; date of current version 8 August 2024. This work was supported by National Natural Science Foundation of China under Grant Number 52130403 and China Fundamental Research Funds for the Central Universities under Grant Number N2017003. *(Corresponding author: Tianhan Gao.)*

Danish Javeed and Tianhan Gao are with the Software College, Northeastern University, Shenyang 110169, China (e-mail: 2027016@stu.neu.edu.cn; gaoth@mail.neu.edu.cn).

Muhammad Shahid Saeed is with the School of Software Technology, Dalian University of Technology, Dalian 116020, China (e-mail: shahidsaeedrana@gmail.com).

Prabhat Kumar is with the Department of Software Engineering, LUT School of Engineering Science, LUT University, 53850 Lappeenranta, Finland (e-mail: prabhat.kumar@lut.fi).

Digital Object Identifier 10.1109/JIOT.2023.3288544

© 2024 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

is first trained on a comprehensive dataset that contains the impression of an attack happening in the targeted application area [10]. After that, the system is deployed into the real-time environment, where it identifies identical attacking scenarios. DL-based IDS undoubtedly provides effective surveillance against suspected entities; however, designing a pertinent IDS is a challenging task. Several factors should be taken into account, such as compatibility, resource consumption, cost, etc., before developing a DL-inspired IDS [11]. In this article, we propose a novel DL-based IDS for edge-envisioned SA in extreme environments.

A. Contribution

The main contributions of this research are summarized as follows.

1) We propose a new DL-based IDS for edge-envisioned SA in extreme environments. The proposed IDS is a hybrid approach and is developed by combining bidirectional gated recurrent unit (BiGRU), long-short-term memory (LSTM) with softmax classifier to detect attacks at the edge of the network. The proposed model proficiently identifies various SA attack types, such as distributed denial of service (DDoS), Ransomware, man-in-the-middle (MITM), Injection, and so on.
2) The proposed IDS employs the truncated backpropagation through time (TBPTT) approach to handle lengthy sequences of SA data. This approach eliminates the need of maintaining the entire history of inputs and activations by recurrent neural network (RNN) and, thus, allows faster learning at the edge of the network.
3) To mitigate the limitations of cloud-based deployment in extreme environments, we present an edge-to-Things envisioned Deployment architecture for the proposed IDS.
4) The experimental results using CIC-IDS2018, ToN-IoT, and Edge-IIoTset datasets shows a proficient performance in terms of all standard and extended evaluation metrics. Finally, to show the superiority of the proposed IDS, we evaluate and compare it with some baseline and recent state-of-the-art techniques.

The remainder of this article is organized as follows. Section II discusses the existing literature. The proposed IA-based threat detection framework is described in Section III. In Section IV, we evaluate the performance of the proposed IDS and perform a comparison with state-of-the-art techniques. Finally, Section V concludes this work with future directions.

II. Related Study

Various DL techniques are used in literature to improve the intrusion detection capabilities of IDS. For instance, Kandhro et al. [12] proposed a DL-driven vulnerabilities assessment framework for IoT-enabled cybersecurity systems. Three state-of-the-art DL classifiers: 1) DNN; 2) CNN; and 3) RNN are employed for this purpose. The designed framework is trained on NSL-KDD, KDDCup99, and UNSW-NB15 datasets. The performance of the proposed model is evaluated on diversified performance metrics. The model has shown competitive performance against Brute Force attacks, Denial of Service (DoS) attacks, infiltration attacks, and botnet attacks. DNN is also utilized in another threat detection model designed for secure cloud communications. In [13], a game theory cloud security deep neural network (GT-CSDNN) is proposed and trained on the CIC-IDS2018 dataset. The proposed IDS has achieved higher accuracy in detecting various attacks.

In [14], a transfer learning approach is used to design an image classification-based threat investigation model for cloud IoT devices. The model is based on CNN, which provides generalized processing strength to the proposed framework. The system includes five pretrained models of CNN known as VGG16, VGG19, EfficientNets, inception, and MobileNet. The model is trained on CICIDS2017 and CIC-IDS2018 datasets and has shown effective detection performance against infiltration attacks, botnet attacks, and port scan attacks. The expanding domain of IoT applications introduces it to maritime transportation systems (MTSs), where IoT brings revolutionary transformations in intelligent communications. In [15], an effective threat detection system is proposed to investigate and classify the existence of common adversaries in such environments. The novel concepts of federated learning generically inspire the system. CNN and multilayer perception (MLP) are employed as the main classifiers, and the model is trained on the NSL-KDD dataset. The simulation results validate the performance of the proposed IDS.

Gradient boosting decision tree (GBDT) algorithm is vital in efficient computation. Light GBM is an enhanced version of GDBT and is an optimal choice for complex DL computations. Light GBM also inspires histogram-based gradient boosting (HBGB). Researchers have used the combined strength of LightGBM and HBGB to formulate a multidimensional attack detection model for IoT empowered multiattack classification environment. The system is trained on the CIC-IDS2018 dataset, where the model has shown remarkable efficiency [16]. In [17], a novel anomaly detection model for next generation IoT (NGIoT) networks is proposed. Wireless spoofing attacks are one of the common categories of mostly occurring attacks. The proposed model is specifically designed to cope with such attacks. The model is equipped with deep auto encoder (DAE) classifier that provides comprehensive durability to the system. The model has shown appropriate resilience against wireless spoofing and botnet attacks. The system has effectively achieved 98.6% anomaly detection accuracy.

In [18], a blockchain-assisted and DL-empowered anomaly detection model for IIoT is proposed. The designed framework is systematically segmented into two major elements. Blockchain ensures secure communication against data poisoning attacks, whereas DL provides privacy preservation for inference attacks by employing deep variational auto encoder (DVAE). Moving forward, A-DGRNN is used to investigate suspicious activities in large-scale IIoT environments. The model is trained on the ToN-IoT and IoT-Botnet datasets.

The authors proposed a network IDS for large-scale general IoT communications. The model is inspired by DNN and is trained on the NSLKDD dataset. During the evaluation

TABLE I
LITERATURE OVERVIEW

Ref	Year	Contribution	Targeted Anomalies	Application Area	Classifier	Dataset
[12]	2023	A vulnerability assessment framework is proposed.	Brute Force XXS, Brute Force WEB, DoS_Hulk , DoS_LOIC_HTTP, Botnet attacks	IoT-enabled cybersecurity systems	DNN, CNN, RNN	NSL-KDD, KDDCup99, UNSW-NB15
[13]	2022	A graph theory-based threat detection model is presented	Cloud-based attacks	Cloud	DNN	CIC-IDS2018
[14]	2023	A transfer learning (CNN) approach is presented to detect intrusions	DDoS, Web attacks, Infiltration attacks, Botnet attacks, Port scan attacks	Cloud IoT devices	CNN	CIC-IDS2017, CIC-IDS2018
[15]	2022	An intelligent adversaries investigation system is proposed	Multifarious cyber attacks	IoT-based Maritime Transportation	CNN, MLP	NSL-KDD
[16]	2021	A novel ensemble attack monitoring framework is devised	Bot, Brute Force attacks, Infiltration attacks, WEB attacks	IoT-empowered multi-attack classification environment	Light GBM HBGB	CIC-IDS2018
[17]	2022	An anomaly detection mechanism is designed	Wireless spoofing attacks, Botnet attacks	NGIoT	DAE	NA
[18]	2022	A blockchain-assisted threat detection framework is developed	Data poisoning attacks, inference attacks	IIoT	DAVE, A-DGRNN	ToN-IoT, IoT-Botnet
[19]	2020	A network IDS is designed	DoS, DDoS	General IoT communications	DNN	NSL-KDD
[20]	2022	A novel threat detection model is presented	Botnet attacks, DoS attacks	IoT-based smart communications	DNNGRU	CIC-IDS2018
[21]	2021	An efficient attack investigation system is designed	DoS, port scan attacks, brute force attacks, Sparta	IoT	CNN	BoT-IoT, MQTT-IoT-IDS2020

process, the model has offered significant resilience against DoS and DDoS attacking categories [19]. DNN is conceptualized in another anomaly detection framework that aims to investigate the presence of suspicious entities in smart communications. The authors trained the model on the CIC-IDS2018 dataset and evaluated it in comparison with long short term memory (LSTM) classifier. Systems have shown promising results in threat detection in IoT-based smart communications [20]. Researchers developed an effective IDS for generic IoT communication environments. The proposed CNN-based model is trained on BoT-IoT and MQTT-IoT-IDS2020 datasets that contain generalized impressions of commonly occurring attacks in IoT communications. The system's performance is evaluated regarding attack detection ACC, PRE, REC, and F1 [21]. Table I summarizes the existing literature.

III. PROPOSED DL-BASED IDS

In this section, we discuss the main components of the proposed DL-based IDS. The proposed IDS is a hybrid model that combines BiGRU, LSTM, and softmax to detect attacks. Furthermore, a TBPTT approach is used to handle lengthy sequences of SA data. The details are explained as follows.

A. *Bidirectional Gated Recurrent Unit*

The RNN and DL architecture with gating properties is called a GRU. It can effectively handle the tasks related to time sequence since it is a connectionist approach along with a self-connected hidden layer and the precise time-series properties of the IoT data. The RNN extracts hierarchical representations from the unprocessed data using a gating function. Although the fundamental RNN model may theoretically store historical data indefinitely, in practice it has the issue of exploding or disappearing gradients [22].

With significant modeling capabilities for long-term dependencies, the GRU and LSTM are upgraded RNN models. Due to its simpler structure and lower computational complexity, GRU is a less complicated variant of LSTM. A BiGRU has the capacity to integrate the cell and hidden states, i.e., to combine the forget and input gates into a single update gate. Furthermore, it consists of an update gate, reset gate, candidate cell, and final state denoted by $\overrightarrow{z_t}$, $\overrightarrow{r_t}$, $\overrightarrow{c_t}$, $\overrightarrow{y_t}$ and $\overleftarrow{z_t}$, $\overleftarrow{r_t}$, $\overleftarrow{C_t}$, $\overleftarrow{y_t}$. The $\rightarrow$ represents the forward process, while $\leftarrow$ represents the backward process, respectively.

GRU can only access old data; it cannot access future data. In an attempt to alleviate this issue, a BiGRU technique is proposed. In BiGRU, one GRU moves in the forward direction, and the other moves in the backward direction, calculating the forward hidden state $(\overrightarrow{y_1}, \overrightarrow{y_2}, \overrightarrow{y_3}, \ldots, \overrightarrow{y_n})$ as well as backward hidden state $(\overleftarrow{y_1}, \overleftarrow{y_2}, \overleftarrow{y_3}, \ldots, \overleftarrow{y_n})$, respectively. The following are the transition functions for BiGRU hidden units [23]:

$$= \begin{cases} \overrightarrow{z_t} = \sigma(\overrightarrow{\mathbb{W}e_z}(\overrightarrow{\mathbb{D}_t}) + \overrightarrow{\mathbb{U}_z}(\overrightarrow{y_{t-1}}) + \overrightarrow{\mathbb{B}s_z}) \\ \overrightarrow{r_t} = \sigma(\overrightarrow{\mathbb{W}e_r}(\overrightarrow{\mathbb{D}_t}) + \overrightarrow{\mathbb{U}_r}(\overrightarrow{y_{t-1}}) + \overrightarrow{\mathbb{B}s_r}) \\ \overrightarrow{c_t} = \tanh(\overrightarrow{\mathbb{W}e_c}(\overrightarrow{\mathbb{D}_t}) + \overrightarrow{r_t} \odot \overrightarrow{\mathbb{U}_c}(\overrightarrow{y_{t-1}}) + \overrightarrow{\mathbb{B}s_c}) \\ \overrightarrow{y_t} = \overrightarrow{z_t} \odot \overrightarrow{y_{t-1}} + \overrightarrow{(1 - z_t)} \odot \overrightarrow{c_t} \end{cases} \quad (1)$$

where, the input's ($\overrightarrow{\mathbb{D}_t}$, $\overleftarrow{\mathbb{D}_t}$) weight matrix for the forward process is represented by $\overrightarrow{\mathbb{W}e_z}$, $\overrightarrow{\mathbb{W}e_r}$, $\overrightarrow{\mathbb{W}e_c}$, while the $\overleftarrow{\mathbb{W}e_z}$, $\overleftarrow{\mathbb{W}e_r}$, $\overleftarrow{\mathbb{W}e_c}$ represents the weight matrix for the backward process. The hidden state of the prior block is denoted by $\overrightarrow{\mathbb{y}_t - 1}$ and $\overleftarrow{\mathbb{y}_t - 1}$, whereas the $\overrightarrow{\mathbb{U}_z}$, $\overrightarrow{\mathbb{U}_r}$, $\overrightarrow{\mathbb{U}_c}$ and $\overleftarrow{\mathbb{U}_z}$, $\overleftarrow{\mathbb{U}_r}$, $\overleftarrow{\mathbb{U}_c}$ represents the weight matrix for $\overrightarrow{\mathbb{y}_t - 1}$ and $\overleftarrow{\mathbb{y}_t - 1}$, respectively. Further, $\overrightarrow{\mathbb{B}s_z}$, $\overrightarrow{\mathbb{B}s_r}$, $\overrightarrow{\mathbb{B}s_c}$ the bias weights for forward process and $\overleftarrow{\mathbb{b}_z}$, $\overleftarrow{\mathbb{B}s_r}$, $\overleftarrow{\mathbb{B}s_c}$ for the backward process. Moreover, the sigmoid operator is represented by σ, tanh is the nonlinear pointwise AF and $\odot$ represents the pointwise multiplication between two vectors

$$= \begin{cases} \overleftarrow{\mathbb{z}_t} = \sigma(\overleftarrow{\mathbb{W}e_z}(\overleftarrow{\mathbb{D}_t}) + \overleftarrow{\mathbb{U}_z}(\overleftarrow{\mathbb{y}_{t-1}}) + \overleftarrow{\mathbb{B}s_z}) \\ \overleftarrow{\mathbb{r}_t} = \sigma(\overleftarrow{\mathbb{W}e_r}(\overleftarrow{\mathbb{D}_t}) + \overleftarrow{\mathbb{U}_r}(\overleftarrow{\mathbb{y}_{t-1}}) + \overleftarrow{\mathbb{B}s_r}) \\ \overleftarrow{\mathbb{c}_t} = \tanh(\overleftarrow{\mathbb{W}e_c}(\overleftarrow{\mathbb{D}_t}) + \overleftarrow{r_t} \odot \overleftarrow{\mathbb{U}_c}(\overleftarrow{\mathbb{y}_{t-1}}) + \overleftarrow{\mathbb{B}s_c}) \\ \overleftarrow{\mathbb{y}_t} = \overleftarrow{z_t} \odot \overleftarrow{\mathbb{y}_{t-1}} + (\overleftarrow{1 - \mathbb{z}_t}) \odot \overleftarrow{\mathbb{c}_t}. \end{cases} \quad (2)$$

Finally, the concatenation of the results of the backward and forward process is denoted by $\mathbb{y}_t$.

$$\mathbb{y}_t = \overrightarrow{\mathbb{y}_t} \oplus \overleftarrow{\mathbb{y}_t} \quad (3)$$

where $\oplus$ represents the elementwise summation.

B. Long-Short-Term Memory

LSTM solved the gradient vanishing problem of RNN by using a gating mechanism, i.e., input, forget, and output gate represented by $\mathbb{I}_t$, $\mathbb{F}_t$, and $\mathbb{O}_t$. The updates to the cell state are handled by the $\mathbb{I}_t$. The steps for updating the cell state are as follows [24]:

$$\mathbb{I}_t = \alpha((\mathbb{W}e_I * \mathbb{y}_{t-1} + \mathbb{W}e_I * \mathbb{Z}_t) + \mathbb{B}s_I) \quad (4)$$

$$\tilde{\mathbb{C}_t} = \tanh((\mathbb{W}e_C * \mathbb{y}_{t-1} + \mathbb{W}e_C * \mathbb{Z}_t) + \mathbb{B}s_C). \quad (5)$$

The $\tilde{\mathbb{C}_t}$ represents the new memory content and the LSTM maintains memory $\mathbb{D}_t$ at each timestamp t

$$\mathbb{D}_t = \mathbb{F}_t * \mathbb{y}_{t-1} + \mathbb{I}_t * \tilde{\mathbb{C}_t}. \quad (6)$$

The $\mathbb{F}_t$ use the current input $\mathbb{Z}_t$ and the prior stage hidden state $\mathbb{y}_{t-1}$ as an inputs. The bias from the bias vector is applied after the input values have been multiplied by the weight matrices

$$\mathbb{F}_t = \alpha((\mathbb{W}e_F * \mathbb{y}_{t-1} + \mathbb{W}e_F * \mathbb{Z}_t) + \mathbb{B}s_F). \quad (7)$$

The $\mathbb{O}_t$ is responsible for regulating the hidden state $\mathbb{y}_t$. Given that it contains all data on prior inputs, the $\mathbb{y}_t$ is necessary in order to make predictions. The following steps are involved for finding the $\mathbb{y}_t$ for the next timestamp:

$$\mathbb{O}_t = \alpha((\mathbb{W}e_O * \mathbb{y}_{t-1} + \mathbb{W}e_O * \mathbb{Z}_t) + \mathbb{B}s_O) \quad (8)$$

$$\mathbb{y}_t = \mathbb{O}_t * \tanh(\mathbb{D}_t) \quad (9)$$

where σ represents the sigmoid operator and tanh is the activation function. Moreover, the weight matrices and their respected biases are denoted by $\mathbb{W}e_I$, $\mathbb{W}e_F$, $\mathbb{W}e_C$, $\mathbb{W}e_O$ and $\mathbb{B}s_I$, $\mathbb{B}s_F$, $\mathbb{B}s_C$, $\mathbb{B}s_O$, respectively. We have further employed the TBPTT algorithm to handle the lengthy sequence of the network data. The edge nodes execute the proposed IDS in the SA network. For a given SA system with state ($\mathbb{S}te$), parameter (ϑ), input ($\mathbb{D}$), the transition function is [25]

$$\mathbb{S}te_{t+1} = \mathbb{F}(\mathbb{D}_{t+1}, \mathbb{S}te_t, \vartheta). \quad (10)$$

The objective is to find a ϑ, which reduces the total loss ($\mathbb{L}oss_T$) at each time step (t) with regard to the desired outputs $\mathbb{O}t_t^*$

$$\mathbb{L}oss_T = \sum_{t=1}^{\mathbb{T}} \mathbb{l}s_t = \sum_{t=1}^{\mathbb{T}} \mathbb{l}s(\mathbb{S}te_t, \mathbb{O}t_t^*). \quad (11)$$

In the case of BiGRU-LSTM, the $\mathbb{S}te_t$=($\mathbb{O}t_t$, $\mathbb{H}_t$), where the $\mathbb{O}t_t$ represents the output's layer activation function and $\mathbb{H}_t$ is the hidden recurrent layer activation. Consequently, the system adopts the following form:

$$\mathbb{H}_{t+1} = \tanh(\mathbb{W}_{\mathbb{d}}\mathbb{D}_{t+1} + \mathbb{W}_{\mathbb{h}}\mathbb{H}_t + \mathbb{B}s) \quad (12)$$

$$\mathbb{O}t_{t+1} = \mathbb{W}_{\mathbb{O}t}\mathbb{H}_{t+1} \quad (13)$$

$$\mathbb{I}_{t+1} = \mathbb{I}(\mathbb{O}t_{t+1}, \mathbb{O}t_{t+1}^*) \quad (14)$$

we have the parameters ϑ=($\mathbb{W}_{\mathbb{d}}$, $\mathbb{W}_{\mathbb{h}}$, $\mathbb{B}s$). The goal is to compute the $\delta\mathbb{L}oss_T/\delta\vartheta$. The backpropagation through time (BPTT) method can be used to perform this calculation. When excessively lengthy sequences are processed using massive networks, the full sequence is processed at each gradient step, which slows the learning. To overcome this problem, TBPTT is used for training the proposed model [26]. Now, the gradient terms becomes $\delta\mathbb{l}s_{t+1}$ ($\delta\mathbb{F}/\delta\mathbb{S}$) ($\mathbb{D}_{t+1}$, $\mathbb{S}te_t$, ϑ) with the transition length $\mathbb{L} < \mathbb{T}$.

If t is multiple of $\mathbb{L}s$ then

$$\delta\hat{\mathbb{l}s}_t = \frac{\delta\mathbb{l}s}{\delta\mathbb{S}te}(\mathbb{S}te_t, \mathbb{O}t_t^*). \quad (15)$$

Otherwise

$$\delta\hat{\mathbb{l}s}_t = \delta\mathbb{l}s_{t+1}\frac{\delta\mathbb{F}}{\delta\mathbb{S}te}(\mathbb{D}_{t+1}, \mathbb{S}te_t, \vartheta). \quad (16)$$

C. Connected Layers

The proposed IDS is furnished with Bi-GRU-LSTM having two Bi-GRU layers with 200 and 100 neurons followed by two LSTM layers of 100 and 50 neurons with 0.3% dropout rate. Moreover, RELU and Softmax functions are used as activation functions, while we have utilized categorical cross-entropy (CC-E) as a loss function. For optimization purposes, we have employed the ADAM optimizer. Finally, the experimentation is conducted for ten epochs with a batch size of 32. Fig. 1 depicts the complete architecture, while the working of the proposed IDS is given in Algorithm 1.

D. Softmax Classifier

Depending on the number of outputs, the BiGRU-LSTM iterates the preceding steps (1)–(9) in various timesteps. Further, the result generated from the BiGRU-LSTM layers is passed to the output layer, i.e., Softmax for the evaluation of the required decision. This is performed by estimating the

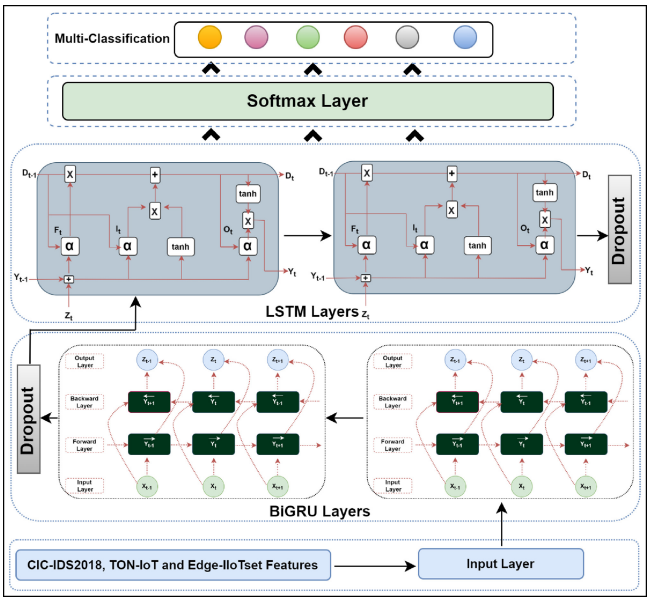


Fig. 1. Proposed IDS architecture.

loss ($\mathbb{L}$) between real ($\mathbb{Re}_o$) and expected ($\mathbb{Exp}_o$) outputs for $\mathbb{k}$ instances using (17) [22]

$$\mathbb{L}\left(\mathbb{Re}_o, \mathbb{Exp}_o\right) = \frac{1}{\mathbb{k}} \sum_{i=1}^{\mathbb{K}} \left(\mathbb{Re}_o - \mathbb{Exp}_o\right)^2. \tag{17}$$

Thus, the proposed IDS framework detects different types of attacks.

E. Deployment Architecture for Proposed IDS

In this section, we have discussed the attack scenario and the deployment/working architecture of the proposed IDS in the extreme environment of SA.

1) Attack Scenario: In the SA environment, most of the IoT devices use a default username or password and sometimes are also used without any passwords. Furthermore, there is also a possibility to perform node-capturing attacks, as IoT devices operate in extreme environments, including wind, snowfall, flooding, landscape, and so on. Thus, they are left unattended for longer periods of time. In such cases, the botmaster, i.e., the attacker who intends to exploit the system vulnerabilities to launch attacks on the target server, installs, or insets malware. Once the IoT system has been compromised, it joins the bot army under the command of its master and can communicate with it for additional guidance. As a result, the hacked IoT devices create a botnet network that is exploited to spread malware [27]. Finally, DDoS attacks are launched against the target edge servers installed in the SA environment using the botnet network.

2) Deployment and Working of Proposed IDS: The deployment architecture and working of the proposed IDS are shown in Fig. 2. In the SA environment, we have multiple entities, such as IoT devices, edge servers, IDS, and cloud servers working together to collect and process SA data. The proposed IDS is RNN-based IDS and, therefore, to enhance the training procedure we have initially used the TBPTT approach to train the model. Moreover, on regular network flow, i.e., when the system is operating under the attack-free mode, the

Algorithm 1 Proposed IDS

procedure INPUT:(Read the Datasets (CIC-IDS2018, ToN-IoT, and Edge-IIoTset))
  OUTPUT: Multiclass classification (Type of Attack)
  Datasets Pre-processing Delete NaN values, Imputation of Infinity values, and Data Normalization (0 and 1)
  Datasets division into $Data_{Train}$, $Data_{Val}$ and $Data_{Test}$.
  Build the intrusion detection model using BiGRU, LSTM, and Softmax
  Add BiGRU layers and perform encoding

$$\overrightarrow{\mathbb{z}_t} = \sigma(\overrightarrow{\mathbb{We}_{\mathbb{z}}}(\overrightarrow{\mathbb{D}_t}) + \overrightarrow{\mathbb{U}_{\mathbb{z}}}(\overrightarrow{\mathbb{y}_{t-1}}) + \overrightarrow{\mathbb{Bs}_z})$$
$$\overrightarrow{\mathbb{r}_t} = \sigma(\overrightarrow{\mathbb{We}_{\mathbb{r}}}(\overrightarrow{\mathbb{D}_t}) + \overrightarrow{\mathbb{U}_{\mathbb{r}}}(\overrightarrow{\mathbb{y}_{t-1}}) + \overrightarrow{\mathbb{Bs}_r})$$
$$\overrightarrow{\mathbb{c}_t} = \tanh(\overrightarrow{\mathbb{We}_c}(\overrightarrow{\mathbb{D}_t}) + \overrightarrow{r_t} \odot \overrightarrow{\mathbb{U}_c}(\overrightarrow{\mathbb{y}_{t-1}}) + \overrightarrow{\mathbb{Bs}_c})$$
$$\overrightarrow{\mathbb{y}_t} = \overrightarrow{z_t} \odot \overrightarrow{\mathbb{y}_{t-1}} + (\overrightarrow{1 - \mathbb{z}_t}) \odot \overrightarrow{\mathbb{c}_t}$$
$$\overleftarrow{\mathbb{z}_t} = \sigma(\overleftarrow{\mathbb{We}_{\mathbb{z}}}(\overleftarrow{\mathbb{D}_t}) + \overleftarrow{\mathbb{U}_{\mathbb{z}}}(\overleftarrow{\mathbb{y}_{t-1}}) + \overleftarrow{\mathbb{Bs}_z})$$
$$\overleftarrow{\mathbb{r}_t} = \sigma(\overleftarrow{\mathbb{We}_{\mathbb{r}}}(\overleftarrow{\mathbb{D}_t}) + \overleftarrow{\mathbb{U}_{\mathbb{r}}}(\overleftarrow{\mathbb{y}_{t-1}}) + \overleftarrow{\mathbb{Bs}_r})$$
$$\overleftarrow{\mathbb{c}_t} = \tanh(\overleftarrow{\mathbb{We}_c}(\overleftarrow{\mathbb{D}_t}) + \overleftarrow{r_t} \odot \overleftarrow{\mathbb{U}_c}(\overleftarrow{\mathbb{y}_{t-1}}) + \overleftarrow{\mathbb{Bs}_c})$$
$$\overleftarrow{\mathbb{y}_t} = \overleftarrow{z_t} \odot \overleftarrow{\mathbb{y}_{t-1}} + (\overleftarrow{1 - \mathbb{z}_t}) \odot \overleftarrow{\mathbb{c}_t}$$

  Concatenate the output of the backward and forward process
$$\mathbb{y}_t = \overrightarrow{\mathbb{y}_t} \oplus \overleftarrow{\mathbb{y}_t}$$
  Add LSTM layers
$$\mathbb{I}_t = \alpha((\mathbb{We}_I * \mathbb{y}_{t-1} + \mathbb{We}_I * \mathbb{Z}_t) + \mathbb{Bs}_I)$$
$$\tilde{\mathbb{C}}_{\mathbb{t}} = \tanh((\mathbb{We}_C * \mathbb{y}_{t-1} + \mathbb{We}_C * \mathbb{Z}_t) + \mathbb{Bs}_C)$$
$$\mathbb{D}_t = \mathbb{F}_t * \mathbb{y}_{t-1} + \mathbb{I}_t * \tilde{\mathbb{C}}_{\mathbb{t}}$$
$$\mathbb{F}_t = \alpha((\mathbb{We}_F * \mathbb{y}_{t-1} + \mathbb{We}_F * \mathbb{Z}_t) + \mathbb{Bs}_F)$$
$$\mathbb{O}_t = \alpha((\mathbb{We}_O * \mathbb{y}_{t-1} + \mathbb{We}_O * \mathbb{Z}_t) + \mathbb{Bs}_O)$$
$$\mathbb{y}_t = \mathbb{O}_t * \tanh(\mathbb{D}_t)$$

  Train classifier using TBPTT approach to handle the lengthy sequence of SA data
$$\mathbb{Ste}_{t+1} = \mathbb{F}(\mathbb{D}_{t+1}, \mathbb{Ste}_t, \vartheta)$$
  Now the gradient terms become
$$\delta\mathbb{Is}_{t+1} \frac{\delta\mathbb{F}}{\delta\mathbb{S}} (\mathbb{D}_{t+1}, \mathbb{Ste}_t, \vartheta)$$
  If t is multiple of $\mathbb{Ls}$ then: $\delta\hat{\mathbb{Is}}_t = \frac{\delta\mathbb{Is}}{\delta\mathbb{Ste}}(\mathbb{Ste}_t, \mathbb{Ot}_t^*)$
  Otherwise:
$$\delta\hat{\mathbb{Is}}_t = \delta\mathbb{Is}_{t+1} \frac{\delta\mathbb{F}}{\delta\mathbb{Ste}} (\mathbb{D}_{t+1}, \mathbb{Ste}_t, \vartheta)$$

  Add softmax layer and calculate the loss
$$\mathbb{L}(\mathbb{Re}_o, \mathbb{Exp}_o) = \frac{1}{\mathbb{k}} \sum_{i=1}^{\mathbb{K}} (\mathbb{Re}_o - \mathbb{Exp}_o)^2$$
  Perform Testing using $Data_{Test}$
  Evaluate performance using various metrics
end procedure

training stage of the proposed IDS can be completed. The trained model can be stored/deployed in the internal memory of the gateway, i.e., edge servers, and can be customized according to commercial vendors. Whenever the network configuration changes, a user can start a new training session to keep the trained model informed (e.g., after the provision of new IoT devices). The Web-based management interface, offered by the edge service providers [e.g., SaaS, Infrastructure as a Service (IaaS)] can be used to keep the retraining process simple. Thus, the proposed IDS works on the ingress edge servers and monitors the incoming packets coming from IoT devices, and generates alerts when any malicious activities are seen. Moreover, the deployment of the proposed IDS on an edge server addresses the challenges related to low latency, geo-distribution, mobility support, and location awareness.

IV. Performance Analysis

In this section, we describe the experimental setup. We then discuss the dataset details, preprocessing steps, and metrics used for evaluation. Then, we discuss the proposed IDS's performance by employing all evaluation metrics. Further, the

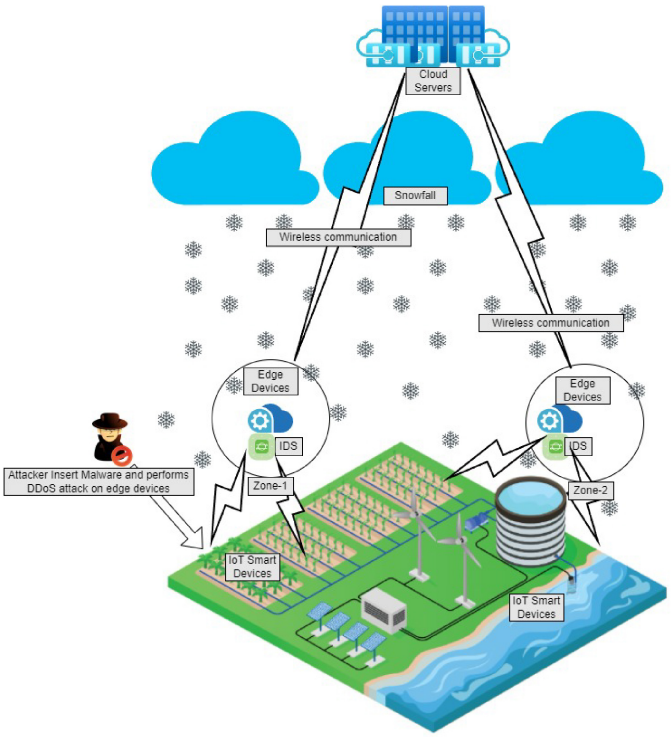


Fig. 2. Deployment architecture for proposed IDS in the extreme environment of SA.

proposed IDS's performance is compared against the traditional modules along with the recent detection schemes from the literature.

A. Experimental Setup

The proposed IDS is designed on Lenovo Legion PC With 24-GB RAM and a HexaCore Processor operating at 2.60 GHz with NVIDIA GEFORCE RTX 2060, 8-GB GPU. Further, Keras, Numpy, Pandas, etc., libraries based on Python are employed for DL techniques.

B. Dataset and Preprocessing

The IoT-based datasets CIC-IDS2018 [28], ToN-IoT [29], and Edge-IIoTset [30] datasets are used to evaluate the performance and efficacy of the proposed IDS. These datasets comprise real-world IIoT network characteristics and attack instances, such that DDoS, XSS, FTP-Patator, SSH-Patator, Botnet, MITM, Injection, Ransomware, Password, Injection, Backdoor, and DoS and its subclasses, i.e., DoS Goldeneye, Hulk, Slowloris, Portscan, DDoS_ICMP, DDoS_TCP, and Vulnerability_Scanner along with Benign instances for assessing artificial intelligence (AI) and cybersecurity systems in applications, such as IDS, adversarial machine learning (AML) [31], and threat intelligence (TI). This work is concerned with ten classes of CIC-IDS2018, nine classes of ToN-IoT and eight classes of Edge-IIoTset datasets as they are the most frequent attacks in such an environment. Finally, preprocessing and data normalization is employed based on [32].

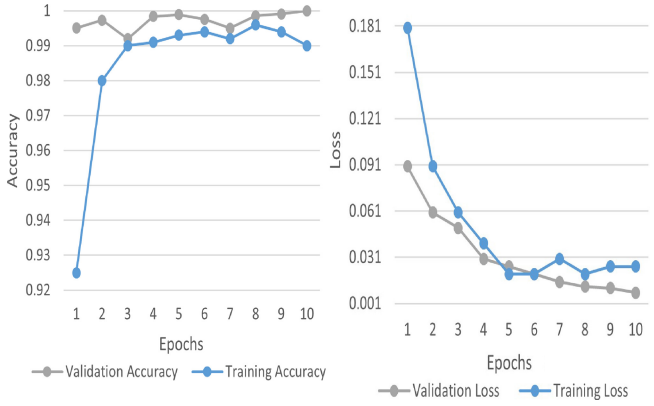


Fig. 3. Accuracy versus loss under CIC-IDS2018 dataset.

C. Evaluation Metrics

All the standard and extended metrics of evaluation, i.e., accuracy (ACC), recall (REC), F1-score (F1), precision (PRE), true positive rate (TPR), confusion matrix (CNM), false positive rate (FPR), ROC curve, true negative rate (TNR), false discovery rate (FDR), Mathhews correlation coefficient (MCC), false omission rate (FOR), and false negative rate (FNR) are used to thoroughly evaluate the performance of the proposed IDS. The following equations compute these metrics:

$$\text{ACC} = \frac{\text{TPR} + \text{TNR}}{\text{TPR} + \text{TNR} + \text{FPR} + \text{FNR}} \tag{18}$$

$$\text{REC} = \frac{\text{TP}}{\text{TPR} + \text{FNR}} \tag{19}$$

$$F1 = 2 \times \frac{\text{PRE} \times \text{REC}}{\text{PRE} + \text{REC}} \tag{20}$$

$$\text{PRE} = \frac{\text{TPR}}{\text{TPR} + \text{FPR}}. \tag{21}$$

D. Performance Analysis of Proposed IDS Against Traditional Detection Schemes

We examine the efficiency of the proposed IDS in this section. Due to its architecture, the proposed IDS framework is a good choice for time-series data analysis. The evaluation findings on ACC versus loss using the CIC-IDS2018, ToN-IoT, and Edge-IIoTset datasets with ten epochs clearly illustrate this in Figs. 3–5. The proposed IDS achieved a validation loss of 0.0032% and validation ACC of 99.82% on the CIC-IDS2018 dataset and validation ACC and validation loss of 99.55% and 0.0028%, respectively, with the ToN-IoT dataset, while it achieved 98.32% validation ACC and 0.0023 under Edge-IIoTset dataset, respectively.

We have also used the CNM and the ROC curve to further assess the efficiency of the BiGRU-LSTM-based scheme. The CNM summarizes the number of records detected successfully or inaccurately by the proposed technique. In CNM, each column signifies events in a predicted class while each row represents events of a class in a given class. Table II represents the CNM of the BiGRU-LSTM-based framework on the CIC-IDS2018 dataset, whereas Tables III and IV depicts the CNM on the ToN-IoT dataset and Edge-IIoTset dataset. The tables portray that the majority of the occurrence in these datasets

TABLE II
CONFUSION MATRIX UNDER CIC-IDS2018 DATASET

Actual / Predicted	*Normal*	*FTP-Patator*	*Hulk*	*SSH-Patator*	*Goldeneye*	*Infiltration*	*Slowloris*	*Botnet*	*Port Scan*	*Web Attack*
Normal	**52023**	0	0	0	5	1	0	1	0	0
FTP-Patator	0	**1541**	0	0	0	5	0	0	0	0
Hulk	1	0	**16843**	0	0	0	0	0	1	0
SSH-Patator	0	0	0	**1525**	0	0	1	0	0	0
Goldeneye	2	0	0	0	**1431**	0	0	0	0	1
Infiltration	0	0	0	1	0	**1676**	0	0	3	0
Slowloris	0	0	1	0	5	0	**1196**	0	0	0
Botnet	5	0	0	0	0	1	0	**1327**	0	0
Port Scan	0	0	1	0	0	0	0	0	**1246**	0
Web Attack	1	0	0	1	0	0	0	0	0	**1397**

TABLE III
CONFUSION MATRIX UNDER TON-IOT DATASET

Actual / Predicted	*Normal*	*MiTM*	*DoS*	*DDoS*	*Password*	*Injection*	*XSS*	*Ransomware*	*Backdoor*
Normal	**78369**	0	0	0	0	0	0	0	0
MiTM	0	**329**	0	0	2	0	5	0	0
DoS	2	0	**5426**	0	1	0	10	1	0
DDoS	0	1	0	**5986**	0	0	0	0	0
Password	0	0	1	0	**6015**	0	0	0	0
Injection	0	2	0	1	0	**5864**	0	0	0
XSS	0	0	0	5	0	0	**5946**	0	0
Ransomware	2	0	0	1	0	0	0	**5973**	0
Backdoor	0	0	0	0	0	3	0	0	**6012**

TABLE IV
CONFUSION MATRIX UNDER EDGE-IIOTSET DATASET

Actual / Predicted	*Normal*	*Backdoor*	*Uploading*	*DDoS_ICMP*	*DDoS_TCP*	*DDoS_UDP*	*Vulnerability_Scanner*	*Ransomware*
Normal	**58899**	41	0	0	0	0	0	8
Backdoor	12	**3112**	5	0	46	0	4	7
Uploading	5	0	**2505**	0	2	0	0	4
DDoS_ICMP	0	0	0	**3715**	0	0	0	1
DDoS_TCP	0	0	0	0	**3605**	0	0	0
DDoS_UDP	0	0	0	0	0	**3637**	0	0
Vulnerability_Scanner	5	27	2	0	0	0	**2946**	2
Ransomware	6	2	0	0	7	0	0	3132

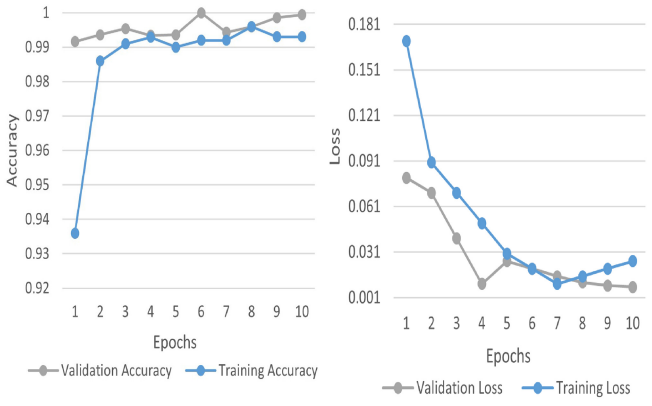


Fig. 4. Accuracy versus loss under ToN-IoT dataset.

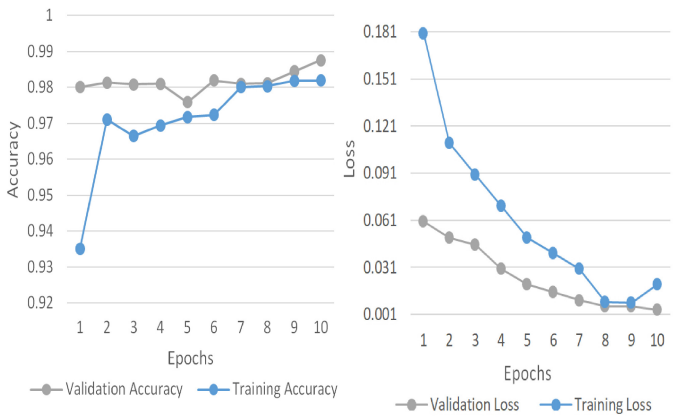


Fig. 5. Accuracy versus loss under edge-IIoTset dataset.

are successfully detected by the proposed scheme. Further, for the ROC curve, the area under the curve (AUC) is determined. The AUC value specifies the efficiency of a model in threat detection. The AUC with high values indicates the model's efficiency in separating the output classes. The ROC curve of the proposed scheme is shown in Fig. 6(a)–(c) for CIC-IDS2018, ToN-IoT datasets, and EDGE-IIoTset datasets. It demonstrates that the AU-ROC scores for all vectors in the datasets are nearly equivalent to one. Moreover, the proposed model is having macro-average of 0.99 and a micro-average

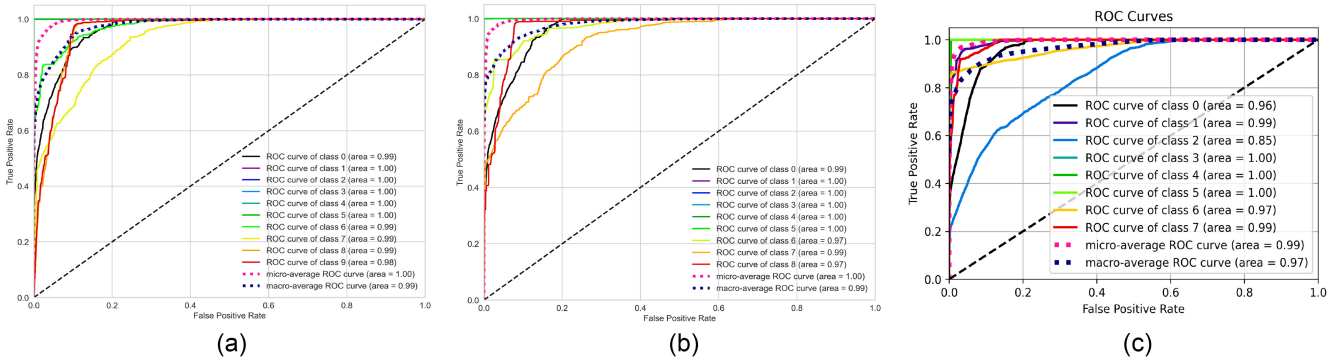


Fig. 6. ROC curve of the proposed IDS. (a) Under CIC-IDS2018 dataset. (b) Under ToN-IoT dataset. (c) Under edge-IIoTset dataset.

TABLE V
PER-CLASS DETECTION RATE OF THE PROPOSED IDS AGAINST TRADITIONAL SCHEMES UNDER CIC-IDS2018 DATASET

Model	Normal	SSH-Patator	FTP-Patator	DoS-Goldeneye	Infiltration	Dos-Hulk	DoS-Slowloris	DDoS-Portscan	Botnet	XSS
Proposed IDS	100%	99.76%	99.89%	99.91%	99.86%	99.51%	99.84%	99.83%	99.92%	99.73%
RF	98.96%	98.83%	98.69%	72.65%	99.26%	99.12%	99.65%	98.79%	99.15%	99.36%
LSTM	99.73%	98.65%	98.59%	98.96%	94.56%	97.59%	99.25%	99.86%	98.65%	98.19%
GRU	98.81%	99.15%	98.69%	94.59%	68.56%	99.28%	99.73%	99.56%	99.36%	99.52%

TABLE VI
PER-CLASS DETECTION RATE OF THE PROPOSED IDS AGAINST TRADITIONAL SCHEMES UNDER TON-IOT DATASET

Model	Normal	MITM	DoS	DDoS	Password	Injection	XSS	Ransomware	Backdoor
Proposed IDS	100%	99.34%	99.36%	99.59%	99.54%	99.36%	99.86%	99.16%	99.94%
RF	99.53%	99.24%	78.69%	91.56%	98.79%	99.53%	99.62%	97.86%	97.25%
LSTM	99.86%	98.76%	99.1%	99.26%	99.46%	98.76%	98.89%	99.48%	98.76%
GRU	96.72%	68.96%	98.56%	99.68%	98.86%	98.79%	99.56%	99.12%	99.31%

TABLE VII
PER-CLASS DETECTION RATE OF THE PROPOSED IDS AGAINST TRADITIONAL SCHEMES UNDER EDGE-IIOTSET DATASET

Model	Normal	Backdoor	Uploading	DDoS_ICMP	DDoS_TCP	DDoS_UDP	Vulnerabiity_Scanner	Ransomware
Proposed IDS	94.29%	98.65%	96.32%	99.99%	99.50%	99.99%	99.30%	98.58%
RF	93.24%	98.81%	77.81%	100%	89.57%	99.97%	99.59%	80.11%
LSTM	96.09%	91.39%	82.07%	99.91%	91.62%	99.83%	93.60%	92.12%
GRU	96.20%	98.39%	81.78%	95.95%	96.50%	99.90%	98.84%	92.06%

of 1 for the CIC-IDS2018 and ToN-IoT datasets. However, for Edge-IIoTset dataset, it has a micro and macro average of 0.99 and 0.97, respectively.

Additionally, a complete performance evaluation of the proposed scheme is conducted by comparing its performance with some traditional threat detection schemes, i.e., RF. LSTM, and GRU. Tables V–VII present the classwise detection ACC achieved by the proposed threat detection scheme with the CIC-IDS2018, ToN-IoT, and Edge-IIoTset datasets and their comparison with the traditional intrusion detection frameworks in recognizing various types of attacks. The Bi-GRU-LSTM-based scheme achieved detection ACC of 100% for normal class, while for other classes it achieved 99.51% to 99.91% accuracy under the CIC-IDS2018 dataset. For the ToN-IoT dataset, the proposed scheme achieved almost 99% to 100% detection ACC, while for the Edge-IIoTset dataset, it achieved values between 94% to 99.99%. On the other hand, the RF, LSTM, and GRU have shown a bad performance in detecting some attack types, i.e., DoS-Goldeneye, DoS, MITM, and achieved a low detection accuracy for other classes as well as against our proposed scheme.

Furthermore, a threat detection scheme is considered proficient if it has high ACC, PRE, REC, and F1 values. An overall comparison of the proposed scheme in terms of the aforementioned evaluation metrics against these traditional schemes can be witnessed in Fig. 7(a)–(c). The proposed scheme achieved a 99.82% ACC, 99.62% PRE, 99.59% REC, and F1 of 99.67% for the CIC-IDS2018 dataset and an ACC, PRE, REC, and F1 of 99.55%, 99.31%, 99.24%, and 99.39% for ToN-IoT dataset, while it achieved a 98.32% ACC, 98.78% PRE, 97.22% REC, and 97.82% F1 under EDGE-IIoTset dataset, respectively. On the other hand, RF, LSTM, and GRU have shown a low performance. This comparison is evident that the BiGRU-LSTM-based threat detection scheme outclassed the traditional schemes by achieving higher values in the aforementioned evaluation metrics.

For further performance evaluation, we have provided the FPR, FNR, FDR, and FOR of the proposed scheme with traditional schemes in Fig. 8(a) for the CIC-IDS2018 dataset, Fig. 8(b) for the ToN-IoT dataset, and Fig. 8(c) for Edge-IIoTset dataset. The proposed IDS achieved an FPR of 0.0369% with FNR, FDR, and FOR of 0.0295%, 0.0245%,

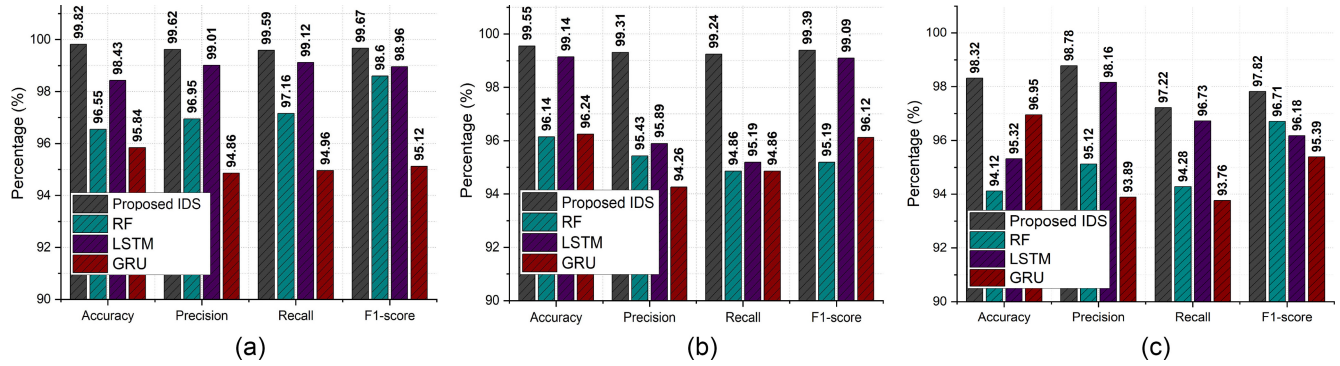


Fig. 7. Overall comparison against traditional schemes. (a) Under CIC-IDS2018 dataset. (b) Under ToN-IoT dataset. (c) Under edge-IIoTset dataset.

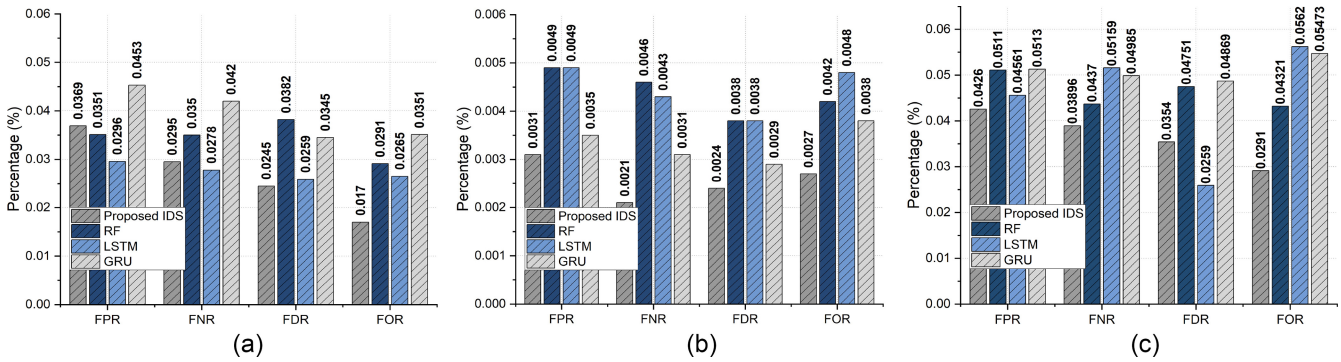


Fig. 8. FPR, FNR, FDR, and FOR comparison. (a) Under CIC-IDS2018 dataset. (b) Under ToN-IoT dataset. (c) Under edge-IIoTset dataset.

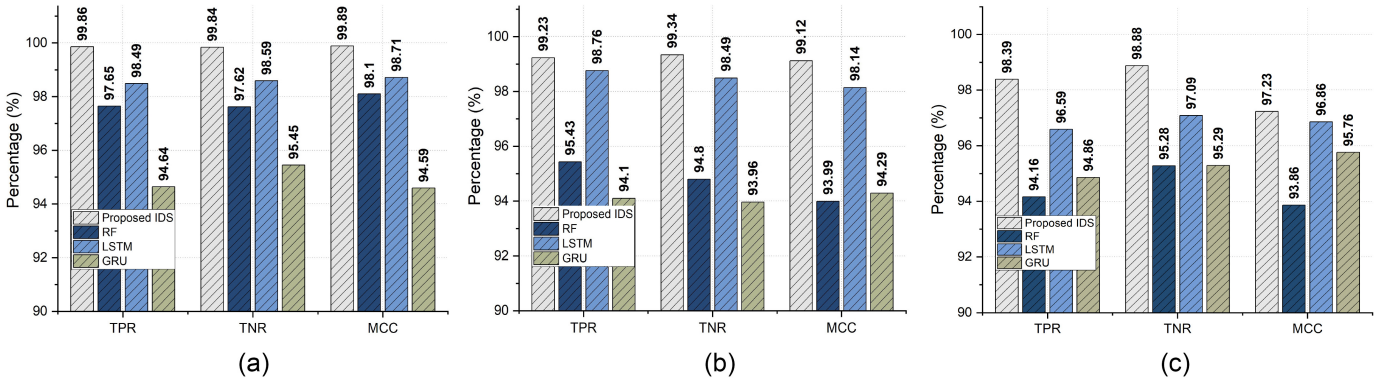


Fig. 9. Comparison in terms of TPR, TNR, and MCC. (a) Under CIC-IDS2018 dataset. (b) Under ToN-IoT dataset. (c) Under edge-IIoTset dataset.

and 0.017% for the CIC-IDS2018 dataset. Similarly, for ToN-IoT, the proposed scheme has an FPR, FNR, FDR, and FOR of 0.0031%, 0.0021%, 0.0024%, and 0.0027%, respectively. Moreover, the proposed IDS achieved an FPR of 0.0462% with FNR, FDR, and FOR of 0.03896%, 0.0354%, and 0.0291% under the Edge-IIoTset dataset, respectively. On the other hand, the LSTM shows a better performance than RF and GRU with the CIC-IDS2018 dataset while under the ToN-IoT dataset, the GRU performed better than RF and LSTM in terms of these metrics. This comparison further proves the efficacy of our proposed threat detection framework by showing higher performance than the traditional detection schemes.

Finally, we have also compared the performance in terms of TPR, TNR, and MCC. Fig. 9(a)–(c) depicts the comparison of the proposed BiGRU-LSTM-based scheme with the aforementioned threat detection schemes. The proposed framework has shown comparatively better performance than these schemes with a TPR of 99.86%, TNR of 99.84%, and MCC of 99.89%, respectively, with the CIC-IDS2018 dataset. For the ToN-IoT dataset, the proposed threat detection framework achieved TPR, TNR, and MCC values of 99.23%, 99.34%, and 99.12%. However, the proposed IDS achieved 98.39% TPR, 98.88% TNR, and 97.23% MCC under the Edge-IIoTset dataset, which is comparatively higher than the other schemes, thus, proving its proficiency. As a result of the findings, we may conclude that the proposed IDS outclassed the baseline threat detection schemes and proves that the proposed BiGRU-LSTM-based scheme is more effective than other techniques at detecting various types of threats. Finally, we provide the testing time of the proposed threat

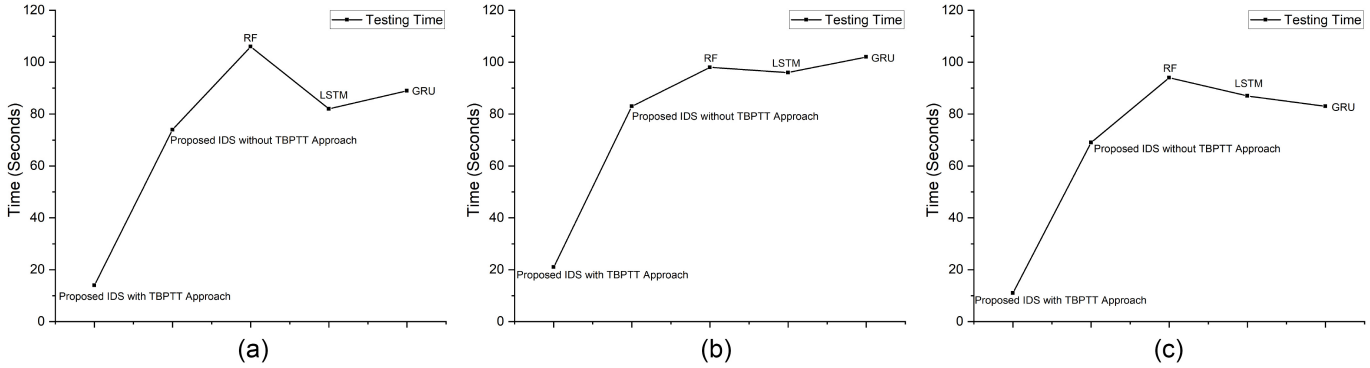


Fig. 10. Comparison in terms of testing time. (a) Under CIC-IDS2018 dataset. (b) Under ToN-IoT dataset. (c) Under edge-IIoTset dataset.

TABLE VIII
COMPARISON WITH RECENT IDS

Reference	Year	Model	Dataset	ACC	PRE	REC
[19]	2020	DNN	NSL-KDD	97.89%	97%	96%
[16]	2021	Light GBM HBGB	CIC-IDS2018	96.97%	NA	97.40%
[13]	2022	GT-CSDNN	CIC-IDS2018	97.96%	98.65%	NA
[20]	2022	DNNGRU	CIC-IDS2018	96.37%	97.78%	97.84%
This Work	2023	Proposed IDS	CIC-IDS2018	99.82%	99.62%	99.59%
			ToN-IoT	99.55%	99.31%	99.24%
			Edge-IIoTset	98.32%	98.78%	97.22%

detection framework and the baseline detection schemes in Fig. 10. We have not considered the training time as it is mostly done offline. Fig. 10(a) depicts the testing time of the proposed IDS and baseline techniques under the CIC-IDS2018 dataset. It can be seen that the proposed IDS has achieved a testing time of 14 s with the TBPTT approach and 76 s without it. Further, Fig. 10(b) depicts the testing time under the ToN-IoT dataset, where the model has a testing time of only 21 s with the TBPTT approach. However, without the TBPTT approach, the model is having a testing time of 83 s. Moreover, for the Edge-IIoTset dataset, we provide the testing time in Fig. 10(c), where the model achieved a testing time of only 11 s with the TBPTT approach, while without it, the model achieved a testing time of 69 s.

E. Performance Comparison With Recent Intrusion Detection Approaches From Existing Literature

Finally, we made the performance comparison of the proposed IDS with recent threat detection approaches from the existing literature such that [13], [16], [19], [20]. A complete comparison is depicted in Table VIII regarding ACC, PRE, and REC. The table is evident that the proposed IDS achieved better results than the existing threat detection frameworks under both datasets and proves its efficacy by outclassing the baseline and recent detection approaches. Thus, ensure the security of IoT devices in such an environment.

V. CONCLUSION

Most of the current IoT devices works in various extreme environment to collect, process, and send real-time data. In this article, we considered edge-to-Things SA scenario in extreme environments and developed a DL-based IDS. The proposed IDS was designed by combining BiGRU, and LSTM with Softmax classifier to detect attacks at the edge of the network. Further, to enhance the training time of the RNN-based IDS, we employed TBPTT mechanism. Thus, the proposed approach eliminates the requirement for a full retrace over the entire data stream at each level. We also designed an attack scenario and deployment architecture for the proposed IDS in the extreme environment of SA. Finally, the proposed IDS outperformed some baselines and state-of-the-art techniques and achieved 99.82%, 99.55%, and 98.32% accuracy and reduced FPR of 0.0369%, 0.0031%, and 0.0426% using CIC-IDS2018, ToN-IoT, and Edge-IIoTset datasets, respectively. Future research will include integrating blockchain and explainable AI techniques with the proposed IDS to enhance the security and privacy of SA in extreme environments.

REFERENCES

[1] P. D. Rosero-Montalvo, Z. István, P. Tözün, and W. Hernandez, "Hybrid anomaly detection model on trusted IoT devices," *IEEE Internet Things J.*, vol. 10, no. 12, pp. 10959–10969, Jun. 2023.

[2] D. Xu, C. Peng, W. Wang, K. Dev, S. A. Khowaja, and Y. Tian, "Multi-keyword ranked search scheme supporting extreme environments for the Internet of Vehicles," *IEEE Internet Things J.*, early access, May 15, 2023, doi: 10.1109/JIOT.2023.3275386.

[3] F. K. Shaikh, S. Karim, S. Zeadally, and J. Nebhen, "Recent trends in Internet-of-Things-enabled sensor technologies for smart agriculture," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23583–23598, Dec. 2022.

[4] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE Netw.*, vol. 33, no. 2, pp. 111–117, Mar./Apr. 2019.

[5] W. Zhou et al., "Priority-aware resource scheduling for UAV-mounted mobile edge computing networks," *IEEE Trans. Veh. Technol.*, early access, Feb. 22, 2023, doi: 10.1109/TVT.2023.3247431.

[6] H. Zhou, Z. Wang, N. Cheng, D. Zeng, and P. Fan, "Stackelberg-game-based computation offloading method in cloud–edge computing networks," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 16510–16520, Sep. 2022.

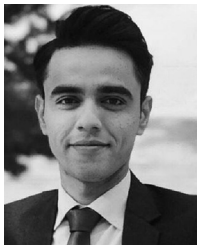
[7] H. Moudoud, Z. Mlika, L. Khoukhi, and S. Cherkaoui, "Detection and prediction of FDI attacks in IoT systems via hidden Markov model," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 5, pp. 2978–2990, Sep./Oct. 2022.

[8] J. Song, Q. Zhong, W. Wang, C. Su, Z. Tan, and Y. Liu, "FPDP: Flexible privacy-preserving data publishing scheme for smart agriculture," *IEEE Sensors J.*, vol. 21, no. 16, pp. 17430–17438, Aug. 2021.

[9] X. Ma, Q. Jiang, M. Shojafar, M. Alazab, S. Kumar, and S. Kumari, "DisBezant: Secure and robust federated learning against Byzantine attack in IoT-enabled MTS," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2492–2502, Feb. 2023.

[10] Y. Otoum, D. Liu, and A. Nayak, "DL-IDS: A deep learning–based intrusion detection framework for securing IoT," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 3, p. e3803, 2022.

[11] S. T. Mehedi, A. Anwar, Z. Rahman, K. Ahmed, and R. Islam, "Dependable intrusion detection system for IoT: A deep transfer learning based approach," *IEEE Trans. Ind. Informat.*, vol. 19, no. 1, pp. 1006–1017, Jan. 2023.
[12] I. A. Kandhro et al., "Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures," *IEEE Access*, vol. 11, pp. 9136–9148, 2023.
[13] A. Jain, K. Tripathi, A. Jatain, and M. Chaudhary, "A game theory based attacker defender model for IDS in cloud security," in *Proc. 9th Int. Conf. Comput. Sustain. Global Develop. (INDIACom)*, 2022, pp. 190–194.
[14] O. D. Okey, D. C. Melgarejo, M. Saadi, R. L. Rosa, J. H. Kleinschmidt, and D. Z. Rodríguez, "Transfer learning approach to IDS on cloud IoT devices using optimized CNN," *IEEE Access*, vol. 11, pp. 1023–1038, 2023.
[15] W. Liu et al., "Intrusion detection for maritime transportation systems with batch federated aggregation," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2503–2514, Feb. 2023.
[16] S. Seth, K. K. Chahal, and G. Singh, "A novel ensemble framework for an intelligent intrusion detection system," *IEEE Access*, vol. 9, pp. 138451–138467, 2021.
[17] M. R. Nosouhi, K. Sood, M. Grobler, and R. Doss, "Towards spoofing resistant next generation IoT networks," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 1669–1683, 2022.
[18] P. Kumar, R. Kumar, G. P. Gupta, R. Tripathi, and G. Srivastava, "P2TIF: A blockchain and deep learning framework for privacy-preserved threat intelligence in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6358–6367, Sep. 2022.
[19] G. C. Amaizu, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, "Investigating network intrusion detection datasets using machine learning," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, 2020, pp. 1325–1328.
[20] S. Seth, G. Singh, and K. Kaur, "Smart intrusion detection system using deep neural network gated recurrent unit technique," in *Proc. 4th Int. Conf. Commun. Cyber Phys. Eng.*, 2022, pp. 285–293.
[21] I. Ullah and Q. H. Mahmoud, "Design and development of a deep learning-based model for anomaly detection in IoT networks," *IEEE Access*, vol. 9, pp. 103906–103926, 2021.
[22] M. Al-Hawawreh, N. Moustafa, S. Garg, and M. S. Hossain, "Deep learning-enabled threat intelligence scheme in the Internet of Things networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 4, pp. 2968–2981, Oct.–Dec. 2021.
[23] X. Luo, W. Zhou, W. Wang, Y. Zhu, and J. Deng, "Attention-based relation extraction with bidirectional gated recurrent unit and highway network in the analysis of geological data," *IEEE Access*, vol. 6, pp. 5705–5715, 2017.
[24] R. C. Staudemeyer and E. R. Morris, "Understanding LSTM—A tutorial into long short-term memory recurrent neural networks," 2019, *arXiv:1909.09586*.
[25] R. Kumar, P. Kumar, R. Tripathi, G. P. Gupta, N. Kumar, and M. M. Hassan, "A privacy-preserving-based secure framework using blockchain-enabled deep-learning in cooperative intelligent transport system," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16492–16503, Sep. 2022.
[26] C. Tallec and Y. Ollivier, "Unbiasing truncated backpropagation through time," 2017, *arXiv:1705.08209*.
[27] S. A. Khanday, H. Fatima, and N. Rakesh, "Implementation of intrusion detection model for DDoS attacks in lightweight IoT networks," *Expert Syst. Appl.*, vol. 215, Apr. 2023, Art. no. 119330.
[28] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. ICISSp*, vol. 1, 2018, pp. 108–116.
[29] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustain. Cities Soc.*, vol. 72, Sep. 2021, Art. no. 102994.
[30] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022.
[31] P. Wang, Z. Li, X. Zhou, C. Su, and W. Wang, "FlowADGAN: Adversarial learning for deep anomaly network intrusion detection," in *Proc. 18th Int. Workshop Security Trust Manage.*, Copenhagen, Denmark, 2023, pp. 156–174.
[32] D. Javeed, T. Gao, M. T. Khan, and I. Ahmad, "A hybrid deep learning-driven SDN enabled mechanism for secure communication in Internet of Things (IoT)," *Sensors*, vol. 21, no. 14, p. 4884, 2021.



Danish Javeed (Student Member, IEEE) received the M.E. degree in computer applied technology from Changchun University of Science and Technology, Changchun, China, in 2020, under the prestigious fellowship of Ministry of Education funded by the Government of China. He is currently pursuing the Ph.D. degree in software engineering (specializing in Information Security with the Software College), Northeastern University, Shenyang, China, under the prestigious fellowship of Ministry of Education funded by the Government of China.

He has authored or coauthored over ten publications in high-ranked journals and conferences. He has many research contributions in the area of deep learning, cybersecurity, intrusion detection and prevention system, the Internet of Things, software-defined networking and edge computing.

Tianhan Gao received the B.E. degree in computer science and the M.E. and Ph.D. degrees in computer application technology from Northeastern University, Shenyang, China, in 1999, 2001, and 2006, respectively.

He started as a Lecturer with the Software College, Northeastern University, in 2006, and was quickly promoted to an Associate Professor in 2010. From 2011 to 2012, he was a Visiting Scholar with the Department of Computer Science, Purdue University, West Lafayette, IN, USA. He has authored or co-authored of over 60 research publications. His key research interests include next-generation network security, security and privacy in ubiquitous computing, and virtual/augmented reality.

Dr. Gao was awarded the title of Doctoral Tutor in 2016.

Muhammad Shahid Saeed is currently pursuing the Ph.D. degree in software engineering from Dalian University of Technology, Dalian, China, under the Chinese Government Scholarship.

He is also working on various projects in collaboration with researchers from Northeastern University, Shenyang, China. He has a few research contributions in the area of the Internet of Things, Industry 4.0, and intrusion detection system.

Prabhat Kumar (Member, IEEE) received the Ph.D. degree in information technology from the National Institute of Technology Raipur, Raipur, India, under the prestigious fellowship of Ministry of Human Resource and Development funded by the Government of India, in 2022.

Thereafter, he worked with Indian Institute of Technology Hyderabad, Hyderabad, India as a Postdoctoral Researcher under project "Development of Indian Telecommunication Security Assurance Requirements for IoT devices." He is currently working as Postdoctoral Researcher with the Department of Software Engineering, LUT School of Engineering Science, LUT University, Lappeenranta, Finland. He has many research contributions in the area of machine learning, deep learning, federated learning, big data analytics, cybersecurity, blockchain, cloud computing, Internet of Things, and software defined networking. He has authored or coauthored over 35+ publications in high-ranked journals and conferences, including 13+ IEEE TRANSACTIONS paper.

Dr. Kumar has served as a Program Co-Chair and a Technical Program Committee Member for major conferences, including IEEE ICCE and ACM CCS. One of his Ph.D. publication was recognized as a top cited article by WILEY in 2020–2021. He is a IEEE Consumer Technology Society Technical Committee Member in Machine learning, Deep learning, and AI in Consumer Electronics.