IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 January 2024, accepted 31 January 2024, date of publication 5 February 2024, date of current version 9 February 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3362232*

RESEARCH ARTICLE

# Majority Voting Ensemble Classifier for Detecting Keylogging Attack on Internet of Things

**YAHYA ALHAJ MAZ[1], MOHAMMED ANBAR[1], (Member, IEEE), SELVAKUMAR MANICKAM[1], SHAZA DAWOOD AHMED RIHAN[2], BASIM AHMAD ALABSI[2], AND OSAMA M. DORGHAM[3,4]**

[1]National Advanced IPv6 Center (NAv6), Universiti Sains Malaysia (USM), Penang 11800, Malaysia
[2]Applied College, Najran University, Najran 11001, Saudi Arabia
[3]School of Computing, Skyline University College, University City of Sharjah, Sharjah, United Arab Emirates
[4]Prince Abdullah bin Ghazi Faculty of Information and Communication Technology, Al-Balqa Applied University, Al-Salt 19117, Jordan

Corresponding authors: Mohammed Anbar (anbar@usm.my) and Yahya Alhaj Maz (Haaj76@gmail.com)

This work was supported by the Deanship of Scientific Research at Najran University under the General Research Funding Program under Grant NU/RG/SERC/12/3.

**ABSTRACT** An intrusion attack on the Internet of Things (IoT) is any malicious activity or unauthorized access that jeopardizes the integrity and security of IoT systems, networks, or devices. Regarding IoT, intrusions can result in severe problems, including service disruption, data theft, privacy violations, and even bodily injury. One of the intrusion attacks is a keylogging attack, sometimes referred to as keystroke logging or keyboard capture, which is a type of cyberattack in which the attacker secretly observes and records keystrokes made on a device's keyboard. In the context of IoT, where connected objects communicate and exchange data, this assault may be especially concerning. Keylogging attacks can have severe repercussions in the IoT ecosystem since they can compromise sensitive information, including login passwords, personal information, financial information, or confidential communications. This paper explored the possibility of using an ensemble classifier to detect keylogging attacks in IoT networks. We built an ensemble classifier consisting of three classifiers: a convolutional neural network (CNN), a recurrent neural network (RNN), and a long-short memory network (LSTM). A proposed model uses the BoT-IoT dataset to detect a keylogging attack. Results show that the ensemble model can improve the model's performance. The ensemble model had excellent accuracy and a low false positive rate. It also had significantly improved detection rates for keylogging attacks than other classifiers.

**INDEX TERMS** Convolutional neural network, Internet of Things, intrusion detection system, keylogging attacks, long short-term memory network, recurrent neural network.

## I. INTRODUCTION

The Internet was previously exclusively accessible via personal computers, mobile phones, and tablets. IoT has enabled it to link various gadgets and appliances, including televisions, air conditioners, and washing machines, to the web [1], [2]. Some areas where IoT has become very important include agriculture, traffic monitoring, energy savings, water supply, and automobiles [3]. IoT connects more terminal devices and facilities to the network, making it one of the most quickly developing and popular technologies on the Internet [4]. Because of this, millions of gadgets can now communicate with one another, making our lives easier. IoT networks are more difficult to protect than conventional computer networks [5] because of the high volume of terminal devices and data they process.

More and more people are doing important stuff online, so it's essential to ensure the data sent between connected devices is secure. Cyber threats and attacks are getting increasingly sophisticated every day, and as networks get more extensive, attackers get smarter and more powerful. IoT can only be used to its full potential if it's secure. Smart gadgets and technologies like hotspots, the Internet, and other IoT are everywhere and need to be secure [6]. Smart tech

The associate editor coordinating the review of this manuscript and approving it for publication was M. Anwar Hossain.

has many advantages, but it also has some weaknesses that can cause cyberattacks that hurt people. It's crucial to use tech with suitable security measures to stop hacking or fraud. Almost all IoT devices are unencrypted, which means personal and confidential info is exposed on the network. IP phones are primarily used in the workplace, making up 44% of all IoT devices, but they only have 5% of the security issues compared to the rest [7].

Security issues are mainly caused by cameras, which account for 33% of the risk but are used in only 5% of corporate settings [8]. Adding IoT smart devices to homes can be risky, as they are more likely to be hacked than other devices. If these smart home devices are compromised, hackers can use keylogging attacks as an example to invade people's privacy and steal data. Keylogging is malicious software that steals personal information from people without their permission. It is usually non-administered software that runs on users' computers and logs all their keystrokes. It is generally easier to carry out this kind of attack if the antivirus and firewall software on the target computer is old.

Moreover, the heightened dependence on digital platforms and the surge in remote work during the COVID-19 pandemic underscored the necessity for robust security measures. Implementing particularly effective keylogging detection became paramount to these measures, given the increased risk of cyber threats and the sensitive nature of the data being accessed remotely. A deep learning intrusion detection system (IDS) is the most successful. However, it can be hard to analyze a lot of data, like traffic and network information, which can affect their performance [9]. To make it easier to detect attacks, IDS should focus on features that help them be more accurate and reduce false positives. This paper uses a deep learning ensemble approach to detect this type of attack depending on a benchmarked dataset called the BoT-IoT dataset [10], which mimics IoT assaults on a network in real-time. They performed comprehensive data analysis studies.

The main contribution of this paper is organized as follows:

- Designed and developed three DL-based models, namely CNN, RNN, and LSTM, to detect keylogging attacks targeting IoT devices.
- An ensemble model based on majority voting utilizes the prediction results of the three DL models developed to enhance the detection of keylogging attacks further.
- A thorough evaluation of the proposed ensemble and the utilized DL models using various evaluation metrics, including the false positive rate, F1 score, detection accuracy, and precision, and compare the proposed approach with state-of-the-art approaches.

The paper is structured into sections: Section II provides the research background. Section III outlines related papers. Section IV introduces the research methodology. The findings and discussion are presented in Section V. Finally, Section VI concludes the paper.

## II. RESEARCH BACKGROUND

This section briefly discusses the relevant topics related to this work, including IoT, IDS, and ensemble Classifier Methods.

### A. IoT

Smart cities, smart homes, smart medical care, and smart agriculture are all possible thanks to the Internet of Things [11]. These innovations have improved the quality of our lives and the way we go about our everyday business. Data gathered and kept in data centers can include large quantities of information, including people's private information [12], [13]. That's because there are millions of IoT gadgets worldwide, and some aren't exactly easy to find. This has resulted in the emergence of countless hazards, both apparent and unseen, with long-lasting consequences. As a result of the high concentration of data, attackers frequently target storage and service servers. When hackers gain access to targeted systems, leaks of sensitive information are inevitable. Because of their limited local storage and computational capabilities, IoT devices may be unable to detect or protect themselves against online threats [14]. Damage to IoT networks might come from even a minor security compromise [15]. Man-in-the-middle (MiTM), DoS, DDoS, spoofing, and jamming are the most frequent forms of attack.

### B. IDS

IDS plays a crucial role in safeguarding an organization's security by detecting and responding to intentional and unauthorized attempts to access, manipulate, or control an Information System or Network [16]. This process, known as Intrusion Detection, involves identifying significant events in the system and scrutinizing them for signs of intrusion. IDS can be implemented using both Hardware and Software, and their primary objective is to prevent unauthorized access and protect against malicious activities [17]. In today's context, network-based attacks are prevalent, and protecting the network is crucial. Researchers have experimented with various techniques like data mining, soft computing, Machine Learning, deep learning, Artificial Neural Networks, etc., to enhance the performance of IDS.

### C. ENSEMBLE CLASSIFIER METHODS

Methods that employ ensemble classifiers build on the strengths of numerous individual classifiers to produce a superior predictive model. The rising popularity of these techniques can be attributed to their ability to boost classification accuracy while decreasing the likelihood of overfitting [18]. Bagging, boosting, and stacking are just a few examples of ensemble approaches [19]:

- Bagging, or bootstrap aggregating, involves creating multiple training data samples and training individual classifiers on each sample. The results of these classifiers are then combined to create a final prediction. One popular algorithm that uses bagging is Random Forest.

- Boosting involves many weak classifiers trained sequentially using weighted copies of the training data. After each cycle, the weights are modified to emphasize the samples incorrectly labeled. One popular algorithm that uses boosting is AdaBoost.
- Stacking, or meta-learning, involves training multiple base classifiers on the training data and then training a meta-classifier to combine their predictions. The base classifiers can be of different types and trained using different algorithms.

## III. RELATED WORK

In [20], the goal of this study for authors was to improve IDS's performance. They set up a binary system for classifying normal IoT traffic and abnormal traffic. They used a bunch of different supervised machine-learning algorithms as well as ensemble classifiers. Their model was trained on datasets called TON_IoT. They used the following classifiers: Random Forest (RF), decision tree (DT), logistic regression (LR), and K-nearest neighbor (KNN). The four classifiers were combined into two ensemble strategies: voting and stacking. They compared the ensemble methods to see how well they solved the classification challenge. Combining these approaches, the ensemble classifiers had better accuracy than the individual models. The experimental results show that their framework improves IDS's performance by 0.9863.

Furthermore, the authors in [21] suggested an EDL-WADS (Ensemble Deep Learning-based Web Attack Detection System). They developed three deep-learning models to detect certain types of online attacks. The results from these three models are then fed into an ensemble classifier, which makes the ultimate call. The authors do tests on several datasets to evaluate EDL-WADS. Compared to the specified baseline models, EDL-WADS shows remarkable overall performance in experimental findings on the CSIC 2010 benchmark dataset. There is a 99.47% rate of accuracy, a 99.29% rate of True Positive Rate (TPR), and a 99.70% rate of precision with a very low False Positive Rate (FPR) of 0.0033. Studies on a real-world dataset further validate the higher performance of EDL-WADS. However, the article highlights two significant shortcomings that should be addressed in future work. The existing EDL-WADS system can only identify SQL injection and cross-site scripting assaults. Second, the CNN model is not doing as well as expected in EDL-WADS. Hence, other models should be investigated. Therefore, the authors recommend looking at alternative deep learning models and working to enhance the performance of EDL-WADS so that it can identify more varieties of online attacks (such as command injection and file inclusion). The reference research [22] aims to improve dependability by creating a two-stage methodology for anomaly identification in industrial IoT networks. Support Vector Machine (SVM) and Naive Bayes (NB) classifiers are blended using an ensemble blending approach in the first stage. K-fold cross-validation is used on the training data at different training-to-testing ratios to find the best possible sets for each. Ensemble blending uses the RF method for predicting class labels. In addition, they use an ANN classifier with the Adam optimizer to improve the precision of our forecasts. The second step involves feeding the ANN and RF findings to the model's classification unit and selecting the one with the best accuracy. Standardized IoT attack datasets, such as WUSTL_IIoT-2018, N_BaIoT, and Bot_IoT, are utilized to assess the proposed model. The experimental data shows a remarkable 99% accuracy.

In addition, in [23], feature engineering and machine learning approaches were used to create a model for intrusion detection in industrial IoT security. The researchers merged Isolation Forest (IF) with Pearson's Correlation Coefficient (PCC) to shorten the computing time and the time needed to make a forecast. While PCC was used to choose the best features, the IF method detected and removed anomalies from the datasets. PCC and IF might be used interchangeably in their respective contexts, hence the terms PCCIF and IFPCC. Implementing an RF classifier helped the intrusion detection system (IDS) function better. An evaluation was performed using the Bot-IoT and NF-UNSW-NB15-v2 datasets. RF-PCCIF and RF-IFPCC performed exceptionally well on the Bot-IoT dataset, with respective Accuracy (ACC) values of 99.98% and 99.99% and prediction times of 6.18 s and 6.25 s, respectively Prediction times of 6.71 s and 6.87 s were also reached by the two models on the NF-UNSW-NB15-v2 dataset, with an ACC of 99.30% and 99.18%, respectively.

Additionally, researchers in [24] evaluated tree-based ensemble algorithms for their ability to spot network assaults in an IoT framework. They used the publicly available and widely used BoT-IoT dataset to do this. The article used several tree-based ensemble approaches to detect intrusions in the IoT network. These included RF, LGBM, Extra Tree, Gradient Boost, and XGBoost. The researchers computed metrics including ACC, F1-score, Recall, and Precision to assess the efficacy of various techniques. They also compared the methods' respective times of detection. When looking at the combined factor of detection time, LGBM was shown to have the best overall performance of all the techniques tested.

Moreover, in [25], authors presented a way to detect IoT attacks by combining two CNNs, called CNN-CNNs. The first CNN is used to figure out the key attributes that make it able to detect IoT attacks from the raw traffic data. Then, the second CNN uses the same key attributes as the first to build a strong detection model that can pick up on IoT attacks. They tested their approach by using the BOT_IoT dataset. The result was 98.04% accuracy, 98.09% precision, 99.85% recall, and 1.93% false positive rate (FPR). They claimed that their approach outperforms other deep learning and feature selection algorithms.

In addition, authors in [26] show a way to detect IoT network intrusions using feature selection techniques and DL models. They use different filter methods, like variance threshold and mutual information, as well as Chi-square and ANOVA. All these methods work together to make up the ensemble. Still, this union operation can sometimes include too many or even redundant features, leading to an oversized

feature set. They use a wrapper algorithm called RFE to ensure the feature selection is fine-tuned. They also look at how the chosen feature set affects the performance of the DL models, like CNNs, RNNs, GRUs, and LSTMs, using a dataset of IoT-Botnets 2020. The results show that all DL models get the best results regarding detection accuracy and precision, as well as F1-measures and FPRs.

As illustrated in Table 1, previous research has made significant strides in enhancing the efficacy of intrusion detection methods for IIoT/IoT attacks. Notably, despite the incorporation of ensemble approaches in select studies ([20], [22]), a distinct gap emerges in the absence of DL classifiers for detecting IoT attacks within these ensembles. Furthermore, it is evident that these ensembles predominantly rely on shallow ML classifiers, highlighting a research gap in the exploration and integration of DL classifiers for a more robust IoT intrusion detection system.

Additionally, it is noteworthy that weighted voting and blending techniques have been utilized in these approaches, as demonstrated in [20] and [22], respectively. However, there is a notable gap in exploring the effectiveness of majority voting for the ensemble process. Investigating the application of majority voting in ensemble methods could provide valuable insights into optimizing the decision-making process and further enhancing the overall performance of IoT intrusion detection systems. Through this comprehensive lens, our study aims to contribute a robust method that can detect the presence of keylogger attacks in IoT networks with high detection accuracy.

## IV. METHODOLOGY

This paper presents a max voting ensemble approach to detect keylogging attacks in IoT networks, comprising three phases: data processing, building ensemble classifiers, and max voting ensemble phases. The first phase prepares the data for training and testing. Phase 2 builds CNN, RNN, and LSTM classifiers as the learning mechanism. The third phase uses a voting ensemble classifier to detect keylogging attacks. Figure 1 illustrates the general architecture of the proposed classifier.

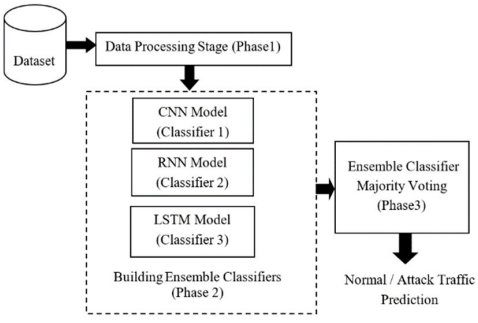



**FIGURE 1.** The general architecture of the proposed classifier.

### A. DATA PROCESSING PHASE

Data processing in deep learning serves to prepare and transform unstructured data into a format that deep learning models can exploit. Deep learning models need a lot of data to learn and produce precise predictions or judgments. Data processing is essential to prepare the input data for deep learning algorithms. In our proposed work, we have incorporated techniques such as removing missing data, data transformation, and normalization. These steps ensure that the input data aligns with the requirements of deep learning algorithms, ultimately enhancing the model's ability to produce precise predictions.

### B. BUILDING ENSEMBLE CLASSIFIERS PHASE

The components of the ensemble classifier built in this phase are as follows:

#### 1) CNN CLASSIFIER

CNN classifier involves building and training the model on the source training dataset and validating its accuracy using the dataset. Two convolutional layers, one with 32 filters and the other with 64, make up the model, with two pooling layers that compress the convolution output and pick out the most relevant features: a flattening layer and a dense layer with a ReLu activation function.

The output layer uses a softmax activation function on its two outputs to distinguish between regular and malicious data. Figure 2 illustrates the stage as follows:

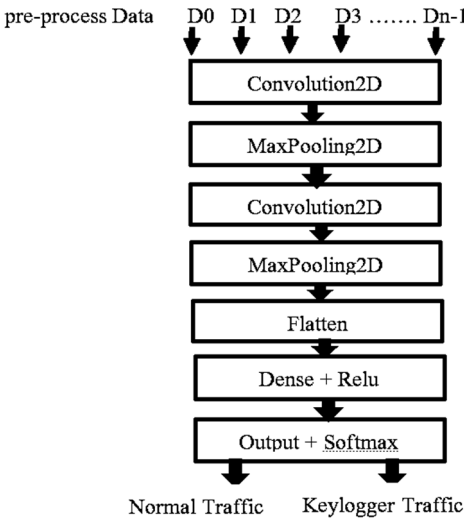



**FIGURE 2.** CNN classifier structure.

#### 2) RNN CLASSIFIER

The RNN classifier has a unidirectional flow of information from input units to hidden units. Figure 3 shows how the previous temporal concealment unit's one-way information flow is synthesized with the current timing hiding unit. The hidden units function as the network's memory, retaining end-to-end information. RNNs are helpful for supervised classification learning because they have a directional loop that can remember and apply previous information to current output. This is a significant difference from traditional FNNs.

**TABLE 1. Summary of the related works.**

| Ref | Proposed Method | Benchmark Dataset | Accuracy | Advantages/Disadvantages |
|---|---|---|---|---|
| [20] | KNN, DT, LR, RF | TON_IoT | 98.63% | The performance of their method is improved. However, they need to test their approach on many datasets. |
| [21] | EDL-WADS | CSIC 2010 | 99.47% | Studies on a real-world dataset further validate the higher performance of EDL-WADS. However, the article highlights two significant shortcomings: The existing EDL-WADS system can only identify SQL injection and cross-site scripting assaults. Second, the CNN model is not doing as well as expected in EDL-WADS; hence, other models should be investigated. |
| [22] | SVM, NB, RF, ANN | WUSTL_IIoT-2018, N_BaIoT, and Bot_IoT | 99% | Create an effective ensemble method for multiple datasets. However, the authors do not mention the limitations of their method. |
| [23] | IF, PCC | Bot-IoT and NF-UNSW-NB15-v2 | 99.98%, 99.30% | The model's performance shows it can overcome the Bot–IoT dataset imbalance. However, it must leverage other datasets, including the TON–IoT dataset containing IoT/IIoT data. |
| [24] | CNN | BoT-IoT | Accuracy:98.04%, precision: 98.09%, recall: 99.85%, FPR: 1.93% | The proposed method outperforms the existing methods, |
| [25] | CNN | BoT_IoT | | The proposed study combined two CNNs to select the essential features related to IoT attacks. While the proposed method needs to explore different feature selection techniques for other machine learning algorithms in conjunction with CNNs |
| [26] | CNN, RNN, GRU, LSTM | IoT_Botnet | Accuracy:97.05% Precision: 97.87% F1-score: 96.99%. | The proposed method enhanced the performance. However, they need to extend their method to other techniques to improve the performance more than they get. |

The previous output in a sequence is related to the current output, and the hidden layer nodes have connections instead of being connectionless. The input and output of both the input layer and the last hidden layer influence the input of the hidden layer. Figure 3 illustrates the steps involved in RNN-IDS.

### 3) LSTM CLASSIFIER

The last classifier in this paper to combat keylogging attacks is LSTM, a type of Deep Learning that belongs to the family of RNNs. LSTM is characterized by its ability to store information in its memory for longer. As illustrated in Figure 4(a), LSTM consists of several components: an Input Gate that decides whether a new input can pass through, a Forget Gate that eliminates information that is not relevant or allows it to affect the output, an Output Gate that determines the output, a single Cell that represents the Constant Error Carousel, and activation functions that calculate the activation of the three gates.

The LSTM model consists of an input layer, an LSTM layer, a dense layer, and an output layer. Then, compile the model with an optimizer and a loss function and train it on the target dataset, as depicted in Figure 4(b).

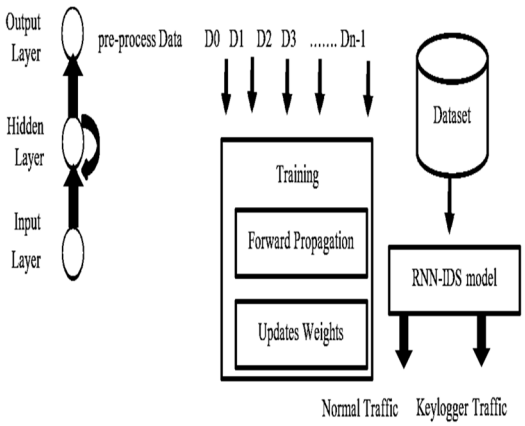
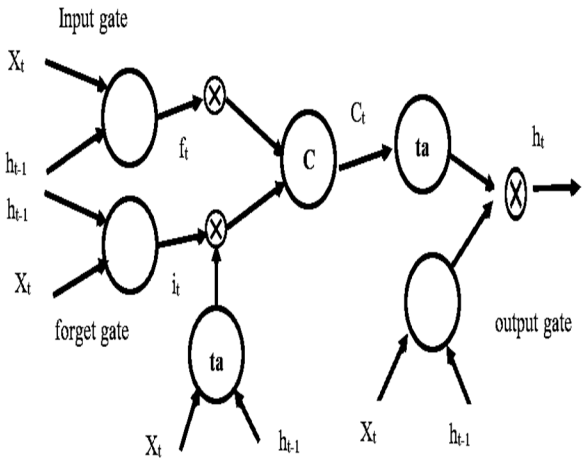


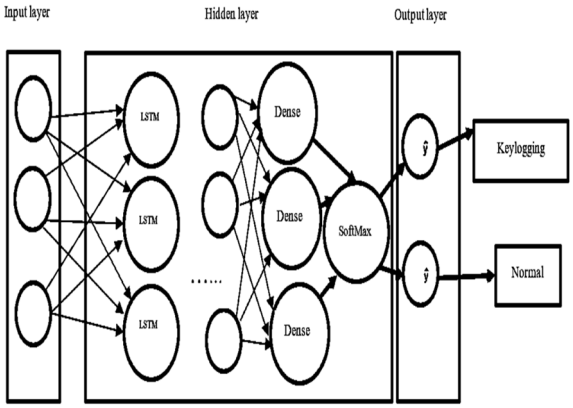

**FIGURE 3. RNN classifier structure.**

## C. MAJORITY VOTING ENSEMBLE CLASSIFIER PHASE

Three base classifiers are selected: CNN, RNN, and LSTM. Each base classifier is trained independently on the training data using a specific training algorithm. The training process involves feeding the training data into each base classifier and adjusting their internal parameters to optimize performance.




(a) LSTM cell architecture




(b) Target LTSM IDS architecture

**FIGURE 4.** LSTM classifier structure.

After that, each trained base classifier is given an input instance or test sample to classify. Based on its learned model, each base classifier separately predicts the class label or probability for the input instance.

Regarding classification, each base classifier generates a predicted class label for the input instance. All three base classifiers' projected class labels are gathered. The ensemble's prediction is determined using a majority vote procedure in the final stage. The projected class label with the highest frequency of occurrence among the three base classifiers is chosen as the ensemble's final prediction. A tie-breaking technique, such as selecting the class label with the highest confidence score or utilizing a specified order of preference, can be used in the event of a tie. Figure 5 illustrates the proposed ensemble classifier.

It is worth mentioning that the results of keylogger attack detection can serve as a valuable countermeasure. For instance, if suspicious patterns indicative of keylogging activities are identified in the connection flow, immediate actions, such as triggering account lockouts or alerts, can be implemented. Furthermore, this information can inform firewall

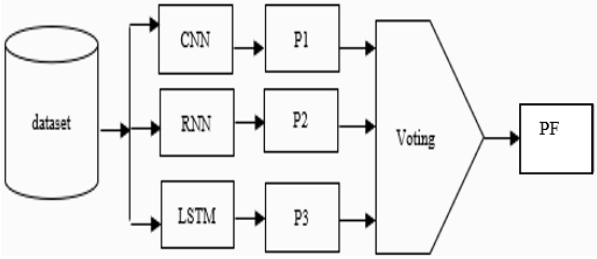



**FIGURE 5.** Ensemble classifier structure.

and proxy rules, enabling the restriction or monitoring of traffic to and from potentially compromised devices. However, it is essential to note that implementing countermeasures to mitigate keylogger attacks is beyond the scope of this work.

## V. RESULT AND DISCUSSION

This section describes a potential use case for the ensemble approach in IoT. To demonstrate the effectiveness of this approach, we selected the BoT-IoT dataset, which contains a massive quantity of regular IoT network activity.

### A. DATASET DESCRIPTIO

Data from a simulated IoT environment includes benign traffic and malicious attacks, including DOS, DDOS, surveillance, and information theft. There are 46 features in the dataset, with five output classes (1 for regular traffic and 4 for the different attack types). The BoT-IoT dataset contains over 73 million records; however, only 10% of the whole dataset is utilized in this research to facilitate simpler management while maintaining proportionality between the various kinds of attacks. Table 2 provides an overview and description of the BoT-IoT dataset.

### B. ENSEMBLE CLASSIFIER PHAS

We computed the performance measures (referenced in equations 1, 2, 3, 4, and 5) for CNN, RNN, LSTM, and ensemble classifiers to determine the most effective model for spotting keylogging attacks. In the following section, outcomes will be introduced.

### C. PERFORMANCE METRIC

We employ numerous measures, including accuracy, precision, recall, FPR, and F1-score, to measure the efficacy of the ensemble approach for detecting keylogging attacks. These measures are derived from the confusion matrix that summarizes the results of the classification process in terms of the proportions of correct classifications of attack and normal records (true positives and true negatives) and incorrect classifications of normal and attack records (false positives and false negatives). The accuracy of a model is evaluated by how many of its predictions on a given dataset were accurate. See Eq. (1) and Eq. (2) [27] for the formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

**TABLE 2.** Overview and description of BOT_IoT.

| Category | Subcategory | Records | Description |
|---|---|---|---|
| Normal | Normal | 9543 | Normal transaction data |
| DoS | TCP, UDP, HTTP | 38532480 | An attack that aims to turn off the services provided by a website or network by flooding many requests, overwhelming the target's infrastructure. |
| DDoS | TCP, UDP, HTTP | 33005194 | A Distributed Denial of Service (DDoS) attack is a type of cyberattack where multiple compromised computer systems attack a target by flooding the target with a large amount of traffic. |
| Reconnaissance | OS fingerprinting<br>Service Scanning | 1821639 | Attack involves collecting information about a target, such as its network or system vulnerabilities, to exploit them. |
| Information theft | Keylogging<br>Data exfiltration | 1469<br>118 | The act of stealing personal information belonging to an individual. |

$$FPR = \frac{FP}{TN + FP} \tag{2}$$

Recall measures the model's ability to classify all relevant items in a dataset correctly. It is calculated as the number of correctly classified items divided by the total number of relevant items. To rephrase, it represents the percentage of actual positive cases that the model correctly identified, see Eq. (3) [28]:

$$R = \frac{TP}{TP + FN} \tag{3}$$

Precision measures the model's ability to identify positive cases in a dataset correctly. It is calculated as the number of correctly classified items divided by the total number of items predicted as positive by the model. In other words, it represents the proportion of predicted positive cases that were correct, see Eq. (4) [28]:

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$F1 - score = \frac{2 * (p + r)}{p + r} \tag{5}$$

### D. SYSTEM SETUP

The HP laptop 15-da2xxx with an Intel(R) Core (TM) i7-10510U CPU @ 1.80 GHz, 2.30 GHz, and 8 GB RAM is the basis for the lab's testing setup. TensorFlow [29], Keras [30], Pandas, and Scikitlearn are used to build models. The software specifications used to implement and test the effectiveness of the proposed method are windows 10 home edition and python language.

The dataset used to train the detection models consists of 3000 records (1469 keylogging records and 1531 normal records). Despite the small amount of data used to train the detection models, ensemble techniques can be effective even with limited datasets.

### E. TRAINING PARAMETER

The BoT-IoT dataset is used to train the suggested method. All models (CNN, RNN, LSTM, and ensemble) are trained using a batch size 2048, a learning rate of 3 ^10-5, the Adam optimizer, and the definite cross-entropy loss function with the 20 epochs. Table 3 summarizes the various model-training settings that might be used.

**TABLE 3.** Training parameters for CNN, RNN, LSTM, ensemble models.

| Model | Epochs | Batch Size | Optimizer | Learning Rate | Loss |
|---|---|---|---|---|---|
| CNN Model | 20 | 2048 | Adam | $3 \wedge 10^{-5}$ | categorical cross-entropy |
| RNN Model | 20 | 2048 | Adam | $3 \wedge 10^{-5}$ | categorical cross-entropy |
| LSTM Model | 20 | 2048 | Adam | $3 \wedge 10^{-5}$ | categorical cross-entropy |
| Ensemble Model | 20 | 2048 | Adam | $3 \wedge 10^{-5}$ | categorical cross-entropy |

### F. RESULT

The proposed approach is evaluated with the BoT-IoT dataset containing keylogging attacks. Four classifiers (CNN, RNN, LSTM, and ensemble) are used to validate the effectiveness of the proposed approach in detecting keylogging attacks: CNN, RNN, LSTM, and ensemble models. The CNN classifier obtains an accuracy of 94.62%, precision of 94.75%, recall of 94.82%, false positive rate of 5.3%, and F1-score of 94.78% when evaluated using the BoT-IoT dataset. The RNN classifier attains an accuracy of 87.01%, precision of 86.95%, recall of 87.23%, false positive rate of 12.77%, and F1-score of 87.09%. The LSTM classifier attains an accuracy of 88.20%, precision of 88.40%, recall of 88.52%, false positive rate of 11.50%, and F1-score of 88.46%. The ensemble classifier attains an accuracy of 97.67%, precision of 97.72%, recall of 97.68%, false positive rate of 2.32%, and F1-score of 97.70%. Figures 6, 7, 8, and 9 show the Accuracy, Precision, Recall, and FPR improvement, respectively.

To validate the efficiency of the ensemble model in detecting keylogger attacks, we tested it on the CSE-CIC-ID2018 dataset, which includes keylogger attacks. The ensemble model achieved an accuracy of 95.03 %, precision of 95.09 %,

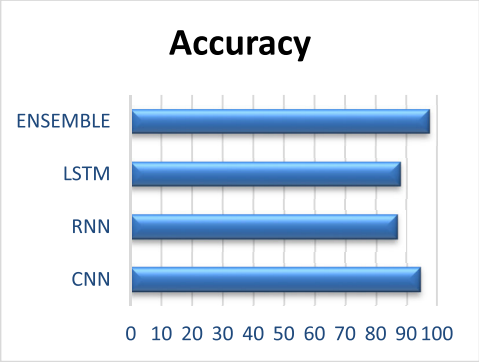



**FIGURE 6.** An improvement in accuracy.

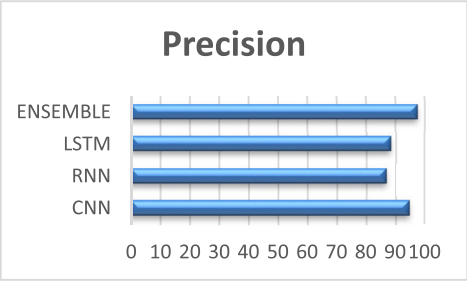



**FIGURE 7.** An improvement in precision.

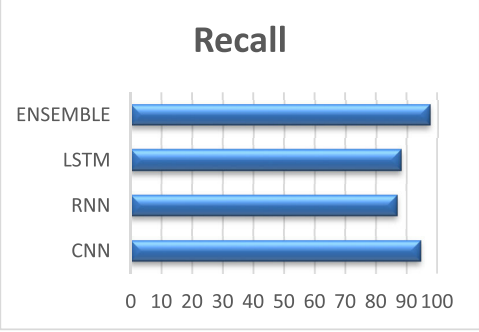



**FIGURE 8.** An improvement in recall.

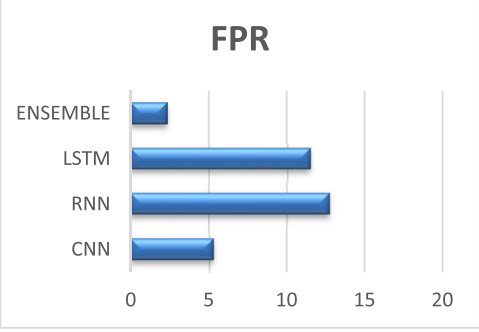



**FIGURE 9.** An improvement in FPR.

recall of 95.05 %, FPR of 4.82 %, and F1-score of 95.07 %. The CNN model achieved an accuracy of 93.03 %, precision of 93.09 %, recall of 93.05 %, FPR of 7.82 %, and F1-score of 93.07 %. The RNN model achieved an accuracy of 87.02 %, a precision of 87.07 %, a recall of 87.99 %, an FPR of 13.03 %, and an F1-score of 87.53 %. The LSTM model achieved an accuracy of 88.22 %, precision of 88.37 %, recall of 88.69 %, FPR of 12.3 %, and F1-score of 88.53 %.

Overall, the proposed ensemble method achieved a very low FPR of 4.82 % and had a detection rate of 95.03%. Table 4 illustrates the performance metrics using the CSE-CIC-ID2018 dataset.

**TABLE 4.** Summary of the performance metrics using CSE-CIC-ID2018 dataset.

| Model | Detection Rate (%) | Precision (%) | Recall (%) | FPR (%) |
|---|---|---|---|---|
| CNN | 93.03 | 93.09 | 93.05 | 7.82 |
| RNN | 87.02 | 87.07 | 87.99 | 13.03 |
| LSTM | 88.22 | 88.37 | 88.69 | 12.3 |
| Ensemble | 95.03 | 95.09 | 95.05 | 4.82 |

The digital landscape has witnessed exponential growth in IoT device adoption, amplifying the urgency to safeguard them against looming cyber threats. Over the years, the research community has responded by introducing many machine learning-driven methods. Among these protective measures, ensemble classifiers, particularly the Max Voting Ensemble Classifier, have emerged as promising contenders in the battle against keylogging attacks.

In our quest to evaluate these classifiers, we began with the BoT-IoT dataset. With its accuracy of 94.62% and balanced precision and recall, the CNN Classifier underscored its potential to distinguish between attack and non-attack instances. However, its false positive rate of 5.3% indicated a slight propensity to mislabel legitimate activities. The RNN Classifier, while displaying a respectable accuracy of 87.01%, had a slightly elevated false positive rate, suggesting potential areas of refinement. LSTM, another model under consideration, exhibited performance metrics somewhat superior to the RNN, but there remained room for enhancement.

However, it was the Ensemble Classifier that genuinely shone in our study. Achieving an accuracy of 97.67% on the BoT-IoT dataset, it outperformed its counterparts in all metrics, notably maintaining a commendably low false positive rate. This fidelity in threat detection, combined with minimized false alarms, becomes indispensable in real-world applications. To further solidify our confidence in the ensemble approach, we undertook validation on the CSE-CIC-ID2018 dataset. The ensemble classifier's consistency was evident, registering an accuracy of 95.03%. While individual classifiers such as CNN, RNN, and LSTM showcased appreciable results, the ensemble model's prowess was incontrovertible.

Diving into the scientific underpinnings of our findings, several insights emerged:

- Ensemble's Collective Wisdom: The ensemble classifier's ability to tap into the strengths of multiple models, making decisions via majority or weighted voting, offers a comprehensive detection mechanism. This naturally results in an uplifted performance, reducing biases and outliers inherent in individual models.

- Tackling Underrepresented Threats: Key to the ensemble's success is its proficiency in identifying subtle or infrequently occurring attack patterns. Harnessing insights from various classifiers ensures no threat, no matter how underrepresented, slips through the cracks.

In encapsulation, our ensemble classifier is a formidable tool in IoT's security arsenal, consistently outperforming individual models. As our research progresses, our focal point will be to optimize ensemble methodologies further and ascertain their adaptability across diverse threat landscapes.

The accompanying Figures 6 through 9 provide visual corroboration to our assertions, enhancing the narrative's clarity.

### G. DISCUSSION

Overall, the ensemble classifier outperformed the CNN, RNN, and LSTM classifiers in all metrics. The ensemble classifier had a 3.05% improvement in accuracy for the CNN classifier, a 10.66% improvement for the RNN classifier, and a 9.47% for the LSTM classifier. This improvement for the ensemble classifier depends on the final prediction determined by the base classifiers' majority vote or weighted vote. The ensemble classifier reduces biases and improves generalization performance. Therefore, the ensemble classifier is more effective at detecting keylogging attacks, particularly those underrepresented in the dataset, compared to the CNN, RNN, and classifiers. Figure 6 depicts the improvement in accuracy for the ensemble classifier.

The research contributions presented in this study represent a significant advancement in the field of Cybersecurity, specifically in the context of detecting keylogging attacks targeting IoT devices. The development of three deep learning-based models—CNN, RNN, and LSTM—demonstrates a commitment to exploring multiple avenues for threat detection. Each of these models brings its unique capabilities to the table, and this diverse approach ensures a more comprehensive coverage in identifying malicious activities on IoT devices.

The introduction of an ensemble model utilizing majority voting is a pioneering approach to enhancing the detection of keylogging attacks. By combining the predictions of the three DL models, the ensemble classifier outperformed its counterparts across various metrics. The ensemble's ability to improve accuracy, mainly by reducing bias and enhancing generalization performance, underscores its efficacy in addressing keylogging attacks. This novel approach not only increases the overall robustness of the detection system but also aids in capturing threats that may be underrepresented or novel in the dataset.

The research's thorough evaluation using diverse metrics, including false positive rate, F1 score, detection accuracy, and precision, and the subsequent comparison with state-of-the-art approaches further solidify the significance of these contributions. The ensemble classifier's remarkable improvements, such as a 10.66% accuracy boost compared to the RNN classifier, indicate its potential to revolutionize the detection of keylogging attacks in IoT devices. Figure 6 serves as a visual testament to the substantial accuracy enhancements brought about by the ensemble classifier, providing a clear representation of its superiority in safeguarding IoT devices against a range of keylogging threats.

In sum, the ensemble model based on majority voting represents a significant leap in enhancing the detection of keylogging attacks targeting IoT devices. It leverages three deep learning models—CNN, RNN, and LSTM—each with distinct strengths, ensuring a comprehensive approach to threat detection. This complementary learning strategy allows the ensemble to capture a wide range of attack patterns, making it adaptable to the diverse tactics employed by malicious actors.

Moreover, the ensemble model excels in its robustness to noise and variations in data. Unlike individual models, which can be sensitive to outliers and fluctuations, the majority voting mechanism mitigates the impact of errors, reducing the chances of false positives or negatives. This robustness is invaluable in real-world scenarios where data quality may vary.

Bias reduction and enhanced generalization are additional advantages of the ensemble approach. By combining multiple models, the ensemble balances any biases exhibited by individual models, ensuring fair and unbiased detection. Furthermore, the ensemble's improved generalization performance makes it more adaptable to unseen attack variations, increasing its overall detection capabilities.

Ultimately, the ensemble model's primary goal is to enhance detection accuracy, a crucial requirement for safeguarding IoT devices. The research results confirm its superiority, with substantial accuracy improvements over individual classifiers. In summary, the ensemble model based on majority voting is a pioneering solution that not only strengthens the Cybersecurity of IoT devices but also serves as a promising approach for improving threat detection in various domains.

This paper aligns with and effectively addresses the challenges identified in the research problem as it provides:

- Feature importance & reduction of false positives:

The essence of our research problem emphasizes the pivotal role of accurate feature selection in refining IDS. As an answer to this challenge, our ensemble model doesn't exclusively rely on a singular algorithm's output. Instead, it synergistically aggregates insights from an array of models. This aggregation mechanism depends on all models' most consistent and imperative features. Such a strategy is more likely to home in on genuine threats, thereby significantly mitigating the rate of false positives. This method ensures a more refined and precise threat detection system.

- Overcoming limitations of simple heuristics:

It's a recognized challenge in the cybersecurity domain that while simple heuristics provide an initial line of detection, they often falter in the face of evolving cyber threats due to their static nature. In contrast, our deep ensemble classifier adopts a layered analytical stance. By transcending basic heuristic methods, it harnesses the capability to discern intricate attack patterns, offering a more robust detection

mechanism that is especially vital for covert operations like Keylogging.

- Tackling Keylogging in the IoT ecosystem:

By its very nature, Keylogging operates covertly, making its detection a task requiring nuanced analysis. Our ensemble classifier's strength lies in pooling insights from diverse algorithms, granting it a holistic vantage point. This comprehensive view heightens the chances of detecting such covert operations, especially in the IoT realm, where devices constantly communicate and exchange data. Our solution's sheer adaptability and scalability make it optimally suited for the dynamic environment characteristic of IoT.

- Navigating the challenge of limited data:

A salient challenge underscored in our problem statement is the limited data availability on keylogging attacks. Ensemble methods inherently present a novel approach to counteract this data scarcity. By leveraging multiple models, they maximize the extraction of actionable intelligence from available datasets. The Majority Voting mechanism within our ensemble acts as a safeguard; even if individual models falter due to data constraints, the collective decision retains its reliability.

- Empirical validation of the proposed solution:

Beyond the theoretical alignment with the research problem, the empirical results after the deployment of our Majority Voting Ensemble Classifier provide a compelling testament to its efficacy. Our findings showcased a distinct improvement in keylogging attack detection rates. More notably, there was a demonstrable reduction in false positives when benchmarked against traditional methods. This amalgamation of quantitative evidence, combined with the ensemble model's operational insights, reaffirms the potency of our approach.

However, the majority voting ensemble classifier stands out due to its holistic approach, aggregating decisions from multiple models to enhance threat detection accuracy. This ensemble nature inherently reduces overfitting, as individual biases are balanced by collective decision-making. Its adaptable design allows for integrating new models as threats evolve. The classifier demonstrates a notable reduction in false positives, making it a reliable tool in Cybersecurity. Furthermore, in contexts where data is limited, the ensemble method capitalizes on the strengths of individual models, offering richer insights than standalone classifiers.

Despite its advantages, the classifier presents challenges. Its multi-model operation can lead to increased computational costs, which might not be suitable for resource-constrained IoT devices. There's a risk that if ensemble models are too alike, the collective benefit diminishes. Implementing and maintaining such a system can be complex, demanding specialized expertise. The involvement of numerous models can sometimes obscure the rationale behind specific decisions, complicating result interpretation. Additionally, processing through multiple models might introduce undesirable latency in real-time detection scenarios.

In the rapidly evolving landscape of IoT security, the continuous emergence of diverse attack vectors necessitates the development of robust and adaptive detection mechanisms. Keylogging, a pernicious attack that aims to record user inputs surreptitiously, poses a significant threat to the integrity and privacy of IoT systems. Addressing this, our research introduces a Majority Voting Ensemble Classifier tailored for detecting keylogging attacks on IoT devices. The substantial impacts of this work on the broader research field are manifold, and they offer promising avenues for both current and future cybersecurity endeavors, including:

1. Enhanced Security for IoT Devices: IoT devices present a vast attack surface for cyber threats, given their pervasive nature and integration into everyday life. Keylogging attacks can result in unauthorized access to sensitive data, leading to privacy breaches. By developing a Majority Voting Ensemble Classifier to detect such attacks, our research strengthens the security measures for IoT devices, making them more resilient against potential threats.
2. Introduction of Ensemble Techniques: Ensemble methods, particularly the majority voting mechanism, have succeeded in various domains, such as image and speech recognition. By introducing and validating such a technique for keylogging attack detection in the IoT context, we bring a novel approach that can potentially improve the accuracy and robustness of intrusion detection systems.
3. Reduction in False Positives/Negatives: A reliable intrusion detection system should minimize false positives and negatives. By leveraging an ensemble classifier, we can combine the strengths of multiple models to potentially reduce inaccuracies, ensuring genuine threats are detected and benign activities aren't flagged incorrectly. This can translate to efficient threat mitigation and less overhead for security personnel.
4. Adaptability and Scalability: Ensemble classifiers are inherently adaptable. By integrating newer models or algorithms into the ensemble as they are developed, our approach can stay relevant and adaptive to evolving threats. Additionally, given the scalable nature of ensemble methods, our classifier can be deployed across a wide range of IoT devices and networks.
5. Contribution to the Knowledge Base: Our research also substantially contributes to the academic and professional knowledge base, providing insights, methodologies, and results that can be a foundation for subsequent studies and practical implementations.

In conclusion, the outcomes of our research not only offer a promising method to counter keylogging attacks on IoT devices but pave the way for further exploration of ensemble methods in cybersecurity contexts. We believe our findings have the potential to influence the design of future IoT security systems, making them more robust and adaptable.

### H. COMPARISON WITH OTHER ENSEMBLE METHODS

This section investigates the outcomes of the proposed model as compared with other related ensemble classifiers, including [31], and [32]. Each experiment has been executed fifty

times. The mean values of accuracy, recall, precision, and F1-score are reported in Table 5. It is clearly shown that the proposed model can outperform other ensemble classifiers in all computed measures.

**TABLE 5. Comparison with other works.**

| Model | Accuracy (%) |
|---|---|
| Proposed model | 97.67 |
| [31] | 90.51 |
| [32] | 93.88 |

In evaluating the efficacy of IDS, it is paramount to juxtapose the proposed solution against state-of-the-art ensemble classifiers to glean a comparative perspective. Our endeavor in this research was precisely aimed at achieving this benchmarking.

Table 5 provides a synoptic view of the comparative performance. For systematic rigor, each experiment was subjected to fifty iterations, and the mean values for key metrics like accuracy, recall, precision, and F1-score were extracted.

Upon examining the table, it becomes conspicuously evident that the proposed model manifests a palpable edge over other ensemble classifiers cited in [31], and [32]. The proposed model clinched an accuracy of 97.67%, which is appreciably higher than the rest. To put this into perspective, the closest competing model ([32]) lagged by almost fourth percentage points, a significant margin in the intrusion detection domain.

Diving deeper into the reasons behind such stark superiority, one compelling hypothesis emerges: The proposed model's architecture and integration methodology might be inherently more adept at processing and deciphering the nuanced patterns typical of keylogging attacks. Moreover, combining insights from multiple algorithms, the ensemble approach's versatility provides a more comprehensive view of potential threats. This holistic analysis reduces the chances of misclassification, ensuring both genuine threats and benign activities are correctly identified.

In summary, while existing ensemble classifiers have their merits, the proposed IDS demonstrates its technical prowess in raw performance metrics and accentuates the importance of adaptive and integrated methodologies in cyber threat detection. Such superiority underscores its potential as a robust tool in safeguarding IoT networks.

## VI. CONCLUSION

This paper ventured into the critical domain of Cybersecurity within IoT networks, explicitly targeting the detection of keylogging attacks. The Majority Voting Ensemble Classifier was presented as a potential solution, validated using the BOT-IoT dataset. As the results illustrated, the ensemble classifier achieved commendable performance gains and set itself apart with its remarkable accuracy and a minimized false positive rate.

Beyond these metrics, the insights drawn from this study underscore the potential of ensemble methods in enhancing the security fabric of IoT networks. Given IoT's dynamic and expansive nature, adopting such classifiers can pave the way for a more resilient and adaptive threat detection framework. Comparatively, our classifier demonstrated superiority over previous deep learning-based classifiers in detecting Keylogging, emphasizing the need for diversified and integrated model approaches in Cybersecurity.

In future work, we plan to address the computational limitations of the proposed approach for low-power IoT devices and refine our methodology. Agents will be strategically deployed to collect traffic from IoT devices passively. These agents will serve as data collectors, efficiently gathering information without imposing significant processing demands on the devices. The collected traffic data will be transmitted to a centralized, pre-trained model. This model, trained offline, will then be deployed on edge devices. The centralized model, residing on higher-specification edge devices, will handle computationally intensive tasks such as real-time detection and analysis. By offloading the processing-heavy functions to the edge, we ensure that the resource constraints of individual IoT devices are not compromised, making our solution well-suited for practical deployment. Furthermore, we plan to evaluate the model on handheld IoT devices using genuine network traffic data. Finally, we plan to analyze model generation and detection processing time.

## REFERENCES

[1] D. E. Kouicem, A. Bouabdallah, and H. Lakhlef, "Internet of Things security: A top-down survey," *Comput. Netw.*, vol. 141, pp. 199–221, Aug. 2018.

[2] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of Things (IoT) communication protocols: Review," in *Proc. 8th Int. Conf. Inf. Technol. (ICIT)*, Amman, Jordan, May 2017, pp. 685–690.

[3] M. M. Islam, A. Rahaman, and M. R. Islam, "Development of smart healthcare monitoring system in IoT environment," *Social Netw. Comput. Sci.*, vol. 1, no. 3, pp. 1–11, May 2020.

[4] G. Mois, S. Folea, and T. Sanislav, "Analysis of three IoT-based wireless sensors for environmental monitoring," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 8, pp. 2056–2064, Aug. 2017.

[5] T. A. Al-Amiedy, M. Anbar, B. Belaton, A. A. Bahashwan, I. H. Hasbullah, M. A. Aladaileh, and G. A. Mukhaini, "A systematic literature review on attacks defense mechanisms in RPL-based 6LoWPAN of Internet of Things," *Internet Things*, vol. 22, Jul. 2023, Art. no. 100741.

[6] J. Tong, W. Sun, and L. Wang, "An information flow security model for home area network of smart grid," in *Proc. IEEE Int. Conf. Cyber Technol. Automat., Control Intell. Syst.*, Nanjing, China, May 2013, pp. 456–461.

[7] P. K. Reddy Maddikunta, G. Srivastava, T. Reddy Gadekallu, N. Deepa, and P. Boopathy, "Predictive model for battery life in IoT networks," *IET Intell. Transp. Syst.*, vol. 14, no. 11, pp. 1388–1395, Nov. 2020.

[8] C. Iwendi, P. K. R. Maddikunta, T. R. Gadekallu, K. Lakshmanna, A. K. Bashir, and M. J. Piran, "A metaheuristic optimization approach for energy efficiency in the IoT networks," *Softw., Pract. Exper.*, vol. 51, no. 12, pp. 2558–2571, Feb. 2020.

[9] Y. Sanjalawe and T. Althobaiti, "DDoS attack detection in cloud computing based on ensemble feature selection and deep learning," *Comput., Mater. Continua*, vol. 75, no. 2, pp. 3571–3588, 2023.

[10] *BoT-IoT Dataset*. Accessed: May 4, 2023. [Online]. Available: https://research.unsw.edu.au/projects/bot-iot-dataset

[11] A. Salam and S. Shah, "Urban underground infrastructure monitoring IoT: The path loss analysis," in *Proc. IEEE 5th World Forum Internet Things (WF-IoT)*, Apr. 2019, pp. 398–401.

[12] Y. Hajjaji, W. Boulila, I. R. Farah, I. Romdhani, and A. Hussain, "Big data and IoT-based applications in smart environments: A systematic review," *Comput. Sci. Rev.*, vol. 39, Feb. 2021, Art. no. 100318.

[13] A. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. Alikarar, "A smart home energy management system using IoT and big data analytics approach," *IEEE Trans. Consum. Electron.*, vol. 63, no. 4, pp. 426–434, Nov. 2017.
[14] A. Churcher, R. Ullah, J. Ahmad, S. Ur Rehman, F. Masood, M. Gogate, F. Alqahtani, B. Nour, and W. J. Buchanan, "An experimental analysis of attack classification using machine learning in IoT networks," *Sensors*, vol. 21, no. 2, p. 446, Jan. 2021.
[15] A. Shafique, J. Ahmed, W. Boulila, H. Ghandorh, J. Ahmad, and M. U. Rehman, "Detecting the security level of various cryptosystems using machine learning models," *IEEE Access*, vol. 9, pp. 9383–9393, 2021.
[16] b. I. Farhan and A. D. Jasim, "A survey of intrusion detection using deep learning in Internet of Things," *Iraqi J. Comput. Sci. Math.*, vol. 3, pp. 83–93, Jan. 2022.
[17] M. Ge, X. Fu, N. Syed, Z. Baig, G. Teo, and A. Robles-Kelly, "Deep learning-based intrusion detection for IoT networks," in *Proc. IEEE 24th Pacific Rim Int. Symp. Dependable Comput. (PRDC)*, Dec. 2019, p. 256.
[18] A. Abbas, M. A. Khan, S. Latif, M. Ajaz, A. A. Shah, and J. Ahmad, "A new ensemble-based intrusion detection system for Internet of Things," *Arabian J. Sci. Eng.*, vol. 47, no. 2, pp. 1805–1819, Feb. 2022.
[19] V. Bolón-Canedo and A. Alonso-Betanzos, "Recent advances in ensembles for feature selection," *Intell. Syst. Reference Library*, vol. 147, no. 1, p. 188, 2018.
[20] Y. Alotaibi and M. Ilyas, "Ensemble-learning framework for intrusion detection to enhance Internet of Things' devices security," *Sensors*, vol. 23, no. 12, p. 5568, Jun. 2023.
[21] C. Luo, Z. Tan, G. Min, J. Gan, W. Shi, and Z. Tian, "A novel web attack detection system for Internet of Things via ensemble classification," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5810–5818, Aug. 2021.
[22] I. S. Thaseen, T. R. Gadekallu, M. K. Aboudaif, and E. A. Nasr, "Robust attack detection approach for IIoT using ensemble classifier," *Comput., Mater. Continua*, vol. 66, no. 3, pp. 2457–2470, 2021.
[23] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, M. Azrour, and Y. Farhaoui, "An ensemble learning based intrusion detection model for industrial IoT security," *Big Data Min. Anal.*, vol. 6, no. 3, pp. 273–287, 2023.
[24] P. Chauhan and M. Atulkar, "Selection of tree based ensemble classifier for detecting network attacks in IoT," in *Proc. Int. Conf. Emerg. Smart Comput. Informat. (ESCI)*, Pune, India, Mar. 2021, pp. 770–775.
[25] B. Alabsi, M. Anbar, and S. Rihan, "CNN-CNN: Dual convolutional neural network approach for feature selection and attack detection on Internet of Things networks," *Sensors*, vol. 23, no. 14, p. 6507, Jul. 2023.
[26] S. D. A. Rihan, M. Anbar, and B. A. Alabsi, "Approach for detecting attacks on IoT networks based on ensemble feature selection and deep learning models," *Sensors*, vol. 23, no. 17, p. 7342, Aug. 2023.
[27] M. Anbar, R. Abdullah, I. H. Hasbullah, Y.-W. Chong, and O. E. Elejla, "Comparative performance analysis of classification algorithms for intrusion detection system," in *Proc. 14th Annu. Conf. Privacy, Secur. Trust (PST)*, Auckland, New Zealand, Dec. 2016, pp. 282–288.
[28] K. Kim, M. Erza, A. Harry, and C. Tanuwidjaja, "Network intrusion detection using deep learning," *Nature*, vol. 1, no. 1, pp. 1–79, 2018.
[29] *TensorFlow*. Accessed: Apr. 15, 2023. [Online]. Available: https://www.tensorflow.org
[30] *Keras: The Python Deep Learning Library*. Accessed: Apr. 14, 2023. [Online]. Available: https://keras.io/
[31] M. A. Jabbar, R. Aluvalu, and S. S. Reddy S, "RFAODE: A novel ensemble intrusion detection system," *Proc. Comput. Sci.*, vol. 115, pp. 226–234, Jan. 2017.
[32] N. Thockchom, M. M. Singh, and U. Nandi, "A novel ensemble learning-based model for network intrusion detection," *Complex Intell. Syst.*, vol. 9, no. 5, pp. 5693–5714, Oct. 2023.

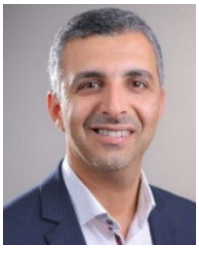
**YAHYA ALHAJ MAZ** received the B.S. degree in computer engineering from Aleppo University, and the M.Sc. degree in computer science from Al al-Bayt University (AABU), in 2006. He is currently pursuing the Ph.D. degree with the National Advanced IPv6 Centre (Nav6), Universiti Sains Malaysia (USM). His research interests include security and privacy issues in the Internet of Things (IoT), and intrusion detection systems (IDSs).

**MOHAMMED ANBAR** (Member, IEEE) received the B.Sc. degree in software engineering from Al-Azhar University, Palestine, in 2008, the M.Sc. degree in information technology from Universiti Utara Malaysia, in 2009, and the Ph.D. degree in advanced internet security and monitoring from Universiti Sains Malaysia (USM), in 2013. He is currently a Senior Lecturer with the National Advanced IPv6 Centre (NAv6), USM. His current research interests include malware detection, intrusion detection systems (IDSs), intrusion prevention systems (IPSs), network monitoring, the Internet of Things (IoT), software-defined networking (SDN) security, cloud computing security, and IPv6 security.

**SELVAKUMAR MANICKAM** is currently an Associate Professor in cybersecurity, the Internet of Things, industry 4.0, and machine learning. He has authored or coauthored more than 160 articles in journals, conference proceedings, and book reviews, and graduated 13 Ph.D. degree students. He has ten years of industrial experience prior to joining academia. He is a member of technical forums at national and international levels. He has also experience in building the IoT, embedded servers, and mobile- and web-based applications.

**SHAZA DAWOOD AHMED RIHAN** received the B.S. degree in computer engineering from the University of Gezira, Sudan, in 2002, the M.Sc. degree in information system from the Arab Academy for Science and Technology, Egypt, in 2007, and the Ph.D. degree in information systems from Omdurman Islamic University, Sudan, in 2016. She is currently an Assistant Professor with Najran University. Her current research interests include computer networks, cybersecurity, and distributed databases.

**BASIM AHMAD ALABSI** received the B.Sc. degree in computer science from Al-Azhar University, Palestine, in 2000, the M.Sc. degree in computer science from Aman Arab University, Jordan, in 2005, and the Ph.D. degree in internet infrastructure security from Universiti Sains Malaysia (USM), in 2020. He is currently an Assistant professor with Najran University. His current research interests include the Internet of Things (IoT), routing protocol for low-power and lossy networks (RPL) security, intrusion detection systems (IDSs), intrusion prevention systems (IPSs), and IPv6 security.



**OSAMA M. DORGHAM** received the B.Sc. degree in computer science from Princess Sumaya University for Technology, Jordan, the M.Sc. degree in computer science from Al-Balqa Applied University, Jordan, and the Ph.D. degree in computing sciences from the University of East Anglia, Norwich, U.K. His research interests include artificial intelligence, image processing, parallel processing, and cyber security. He is an active member in many academic and industrial organizations; in addition, he serves as a member in many international scientific journals. He has been awarded Erasmus grants and international awards during the past few years.