Indian Journal of Modern Research and Reviews

This Journal is a member of the *'Committee on Publication Ethics'*

Online ISSN:2584-184X

Research Paper

A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Networks

Anand Kumar Vishwakarma[1], Pankaj K. Goswami[2], Keshav Sinha[3], Lopamudra Hota[4*]

[1]Department of Computer Science and Engineering, Sarala Birla University, Ranchi, Jharkhand, India

[2]Department of Computer Science and Engineering, Sarala Birla University, Ranchi, Jharkhand, India

[3]Department of Computer Science and Engineering, University of Petroleum and Energy Studies, Dehradun, Uttarakhand, India

[4]Department of Computer Science and Engineering, National Institute of Technology, Rourkela, Odisha, India

Corresponding Author: *Lopamudra Hota

DOI: https://doi.org/10.5281/zenodo.15025268

ABSTRACT

The spreading of Internet of Things (IoT) tools has brought about a revolution in multiple sectors by facilitating smooth connectivity and data transmission. But the rapid expansion has also made IoT net- works vulnerable to serious security risks, triggering the need for reliable Intrusion Detection Systems (IDS). This paper offers an innovative ensemble-based IDS in this research that is tailored to IoT scenarios. By utilizing the Max Voting technique, the approach combines the best features of three machine learning algorithms—Decision Tree (DT), Gaussian Naive Bayes (GNB), and Extreme Gradient Boosting (XGBoost). Through the integration of multiple models, the ensemble method outperforms individual classifiers in detection, hence alleviating their limitations. The empirical results achieve an accuracy of 99.85%, indicating their effectiveness. The findings show that the ensemble approach offers a strong and well-balanced security mechanism against a variety of cyber threats, especially when Max Voting is used.

Manuscript Info.

✓ **ISSN No:** 2584-184X
✓ **Received:** 25-01-2025
✓ **Accepted:** 22-02-2025
✓ **Published:** 13-03-2025
✓ **MRR:3(3):**2025;05-10

✓ **Peer Review Process:** Yes
✓ **Plagiarism Checked:** Yes

How To Cite

Vishwakarma AK, Goswami PK, Sinha K, Hota L. A hybrid ensemble framework for intrusion detection in Internet of Things networks. Indian J Mod Res Rev. 2025;3(3):5-10.

KEYWORDS: IoT Security, Intrusion Detection System, Ensemble Learning, Max Voting, Decision Tree, Gaussian Naive Bayes, XGBoost.

1. INTRODUCTION

The plethora of technological devices that surround the globe nowadays are transforming the lives of individuals. In this context, the Internet of Things (IoT) is emerging as a cutting-edge technology that revolutionizes various industries and makes life easier through intelligent gadgets with improved connections, including smart homes, smart agriculture, smart water management, smart healthcare, and smart environment monitoring. In the context of the IoT, more specifically, many diverse physical devices can collaborate and communicate with one another to transport data over numerous networks without interfering with human-to-human or human-to-device interfaces [1]. Around 41.6 billion (Internet of Things) IoT devices are anticipated to be connected by 2025, which presents numerous obstacles to the actualization of (Internet of Things) IoT in practice [2]. Particularly in big (Internet of Things) IoT networks, where issues with data integrity and confidentiality are present. Security issues have grown in frequency, including zero-day

assaults directed at internet users. Zeroday attacks had a significant impact because of the extensive usage of the Internet in several countries, including the USA and Australia [3]. Cybersecurity is defined as the field concerned with "the protection of networks, data, and systems in cyberspace" [4]. It is the virtual space "resulting from the interaction of people, software, and services on the Internet using technology devices and networks connected to it" [5]. An essential component of system and network security is achieved by Intrusion Detection Systems (IDS). IDS monitors networks or systems for malicious activity or violations and triggers alerts when suspicious activity is detected [6]. IDS development progressed through different stages. These stages developed side by side with the increasing dependence on devices and automation, and the significant development of Machine Learning (ML) and Deep Learning (DL) techniques [7]. Deep Learning (DL) is defined as a class of neural networks that uses multiple layers to extract higher-level features allowing the modelling of complex problems [8]. There will be 29.3 billion networked devices as a result of this. The research goes on to say that attacks increased by 76% between 100 and 400 Gbps between 2018 and 2019 and that they will keep growing in the upcoming years. However, current (Intrusion Detection System) IDS are unable to identify novel and undiscovered assaults due to the expansion of the attack surface and the complexity of new attacks. (Intrusion Detection System) IDS are a critical component of securing the IoT networks. By understanding the challenges and exploring relevant approaches, one can implement a robust (Intrusion Detection System) IDS strategy that safeguards your connected devices and data. An effective (Intrusion Detection System) IDS helps ensure the continued growth and security of the IoT ecosystem. It analyzes network traffic, system logs, or other data sources to identify potential threats such as malware, Denial-of-service (DoS) attacks, or unauthorized access attempts. When an anomaly is detected, the (Intrusion Detection System) IDS triggers an alert, allowing for timely intervention and mitigation actions.

Problem Statement: The IoT is expanding at a rapid pace, which leaves a large network open to many security risks. The distributed nature, resource limitations, and heterogeneity of IoT devices frequently pose challenges for conventional (Intrusion Detection System) IDS. Attackers may be able to take advantage of these vulnerabilities to compromise confidential information and interfere with vital operations. Existing IDS for IoT based on ML frequently depend on single models, which can have drawbacks. These constraints can include a predisposition to particular kinds of attacks, a challenge to adapting to new types of attacks, or heavy computational requirements.

Contribution: This paper proposes an ensemble machine learning approach using the max voting method for intrusion detection in distributed IoT systems. This approach aims to address the limitations of single models and enhance the overall effectiveness of IDS in IoT distributed environment. The rest of the paper is organized as follows. Section II describes the literature study, followed by the proposed work in section III, describing the methodologies, data processing and analysis, and performance metrics. The modeling of machine learning algorithms and result analysis is presented in section IV. Finally, the paper concludes with the conclusion and future work in section V.

2. RELATED WORK

A popular benchmark dataset for research studies trying to increase intrusion detection success rates is (Knowledge Discovery in a Database) KDDCup 99 [9]. The dataset, which was utilized for the third International Knowledge Discovery and Data Mining Tools Competition, was created as the result of tcpdump data that was taken from the (Defense Advanced Research Projects Agency) DARPA Intrusion Detection Evaluation Network in 1998. Creating a predictive algorithm that classifies network connections as either attack or regular was the main goal. The attacks included were Probe, DoS, R2L, and U2R. A thorough survey of ML intrusion detection using the KDDCup 99 dataset [10]. Through an analysis of the KDDCup 99 dataset, the inquiry assessed the effectiveness of different ML methods in identifying intrusions. The authors' findings indicate that on this specific dataset, decision tree and Naive Bayes methods performed admirably. The authors also pointed out that there is a class imbalance in the dataset, although this issue can be resolved by using techniques like under- and oversampling. A general ML approach for recognising IoT devices is presented by Ali *et al.* [11], the authors also assess the trained models using four publically accessible datasets. NFStream used ML models to better detect IoT devices in the network by extracting 85 attributes from packet capture (.pcap) files. The authors trained six machine learning models in the tests using the information gain strategy to choose 20 attributes. Using random forest and naïve Bayes classifiers, the authors achieved remarkable accuracy in the training phase, reaching 99% for IoT device identification. Two popular intrusion detection datasets, KDDCup99 and NSLKDD, were used by Sapre *et al.* [12] in their investigation. Their main goal was to evaluate the two datasets in-depth by evaluating the output of several ML classifiers trained on them using a wider variety of classification criteria than previous research. The authors concluded that the NSL-KDD dataset is of higher quality than the KDDCup99 dataset since the classifiers trained on the KDDCup99 dataset were, on average, 20.18% less accurate. This is due to the bias towards redundancy in classifiers trained on the KDDCup99 dataset, which enabled them to achieve a higher accuracy of 96.83%. By utilising anomaly and outlier detection methods, presents a random forest strategy for misuse detection [13]. The study found that the hybrid system improved performance by combining anomaly detection and abuse, and 4 *A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Network* that the

misuse strategy produced better results than the KDDCup 99 challenge results [14]. However, the approaches do not implement any ensemble learning mechanism.

Proposed Work: To produce an overall prediction that is reliable and accurate, ensemble learning integrates predictions from several machine learning models. When it comes to intrusion detection systems, this entails fusing the advantages of many models to increase intrusion detection precision.

3. METHODOLOGY

3.1 Dataset: The popular benchmark dataset for intrusion detection studies is the KDD Cup 99 dataset. The dataset consists of 494021 data points and 42 features. Some of the data points classes are normal, Neptune, back, teardrop, satan, etc. But it has several drawbacks one of which includes class imbalance. The distribution of target classes in training data is depicted in Figure 1. There are far more instances of regular traffic in the dataset than intrusions, resulting in a severely imbalanced dataset. Due to this, ML models may be biased in favor of the majority class (regular traffic), which could result in subpar intrusion detection. By using diverse models, the ensemble can potentially adapt to various attack types present in the KDD Cup dataset, even if some attack types are less frequent. Ensemble methods can help mitigate the bias towards normal traffic that can occur with single models due to the class imbalance in the KDD Cup dataset. By combining predictions, the ensemble can achieve higher overall accuracy in intrusion detection compared to a single model.

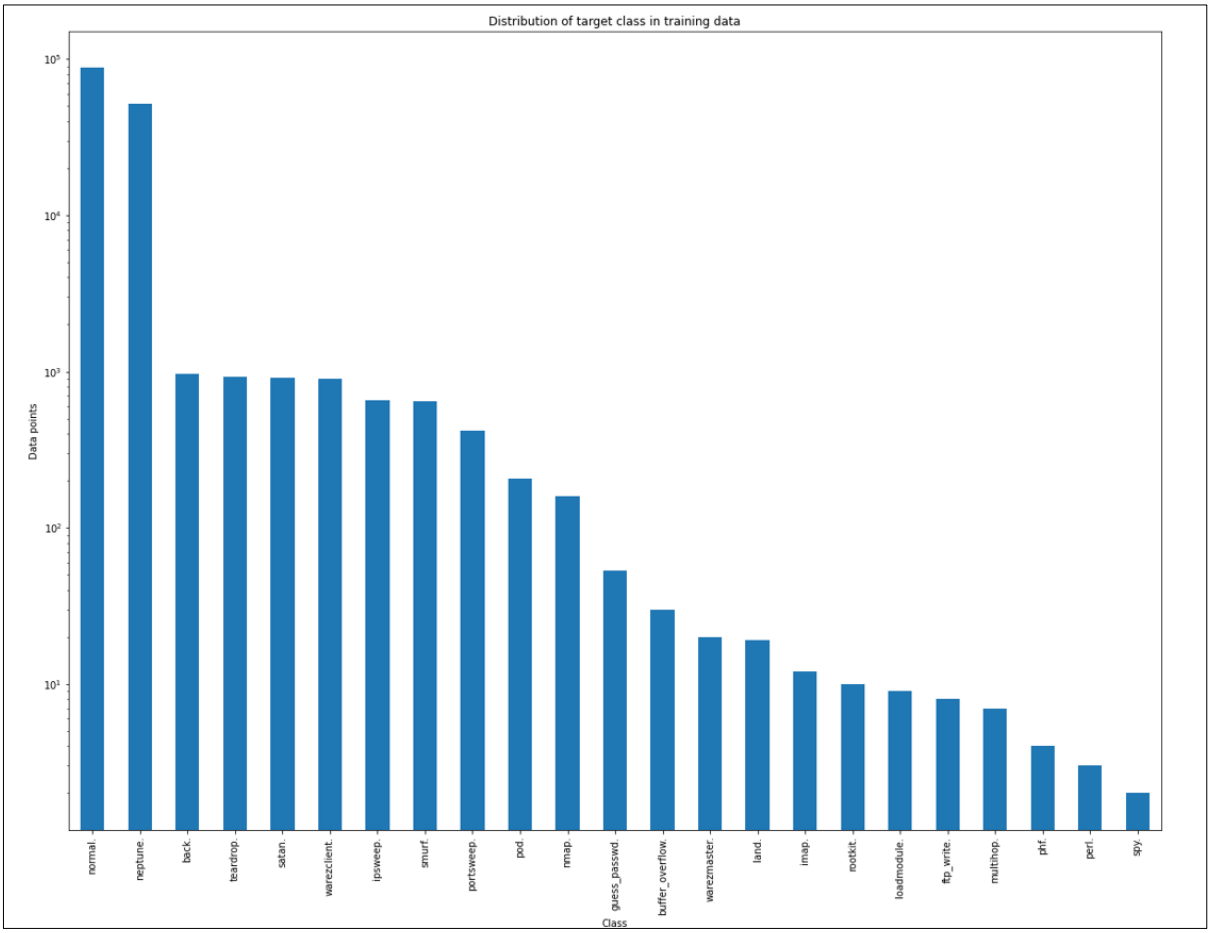

Figure 1: Class Distribution

A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Networks.

3.2 Data Pre-Processing

This procedure addresses the class imbalance by techniques like oversampling (creating more intrusion instances) or under sampling (reducing normal traffic instances) or using specialized algorithms for imbalanced datasets. The steps are as follows;

Handling Missing Value

Examine the data for any missing values. Typical techniques to deal with them consist of:

Imputation: Using statistical (such as mean/median imputation) or model-based (such as KNN imputation) approaches to fill in missing variables.

Removal: Eliminating missing value cases, if they make up a small percentage of the data and their absence, has no discernible effect on the study. Feature Scaling Data about network traffic may contain elements with various sizes. By guaranteeing that every feature has a comparable range of values, feature scaling keeps models from overvaluing features with wider scales. Typical scaling methods consist of:

Standardization: Using z-score normalization to transform features so that their mean is equal to 1 and their standard deviation is 0.

Normalization: It is the process of scaling features to a range, usually between 0 and 1.

Encoding of Categorical Features

Categorical features that reflect protocols, service kinds, etc., are present in intrusion detection datasets. These must be transformed into numerical formats that are comprehensible to machine learning models. Typical encoding methods consist of:

One-Hot Encoding: For every distinct category value, a new binary feature is created. As an illustration, the" protocol" feature with the values" TCP" and" UDP" would be transformed into the two new features" is TCP" and" is UDP".

Encoding Labels: putting a distinct number in front of every category value. Although this approach is more straightforward, it has the potential to create an ordinal relationship where none previously existed (for example, encoding" TCP" as 1 and" UDP" as 2 could suggest an unreal hierarchy).

Identifying and Managing Outliers

Data points that substantially differ from the rest are called outliers. They might cause models to be misled. Among the methods for managing outliers are:

Clipping: Limiting the value of outliers to a predetermined level (for example, swapping out a very high value with the 99th percentile).

Elimination: Eliminating outliers that are deemed abnormal and unrepresentative of the real network traffic. A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Network

3.3 Performance Metrics

Most of the data points are from the "normal" (good connections) category, which is around 60.33%. In the categories that belong to bad connections," Neptune." (35.594%) and" back (0.665%) have the highest no. of data points. Classes" rootkit."," loadmodule."," ftp write."," multichip."," phf."," perl.", "spy." has the least no. of data points with less than 10 data points per class. The dataset is highly imbalanced; thus, we will need to build a model which should be able to classify data points from these low distribution classes accurately. As the dataset is highly imbalanced, we will need to build a model which should be able to classify the intrusion categories accurately. Using src bytes as a feature for analysis, the intrusion category" portsweep" is distinguishable from the rest of the categories. Using dest bytes as a feature for analysis, the intrusion categories" normal"," imap"," multichip", and" warez master" are distinguishable from the rest of the categories. As we have a relatively high no of classes, the Univariate analysis using boxplots and violin plots does not give us clear and satisfactory results. Thus, the pairplots for BiVariate Analysis or with PCA/TSNE is used to reduce the no. of dimensions and perform Bi/Tri- Variate Analysis. The result of the correlation matrix is depicted in Figure 2.

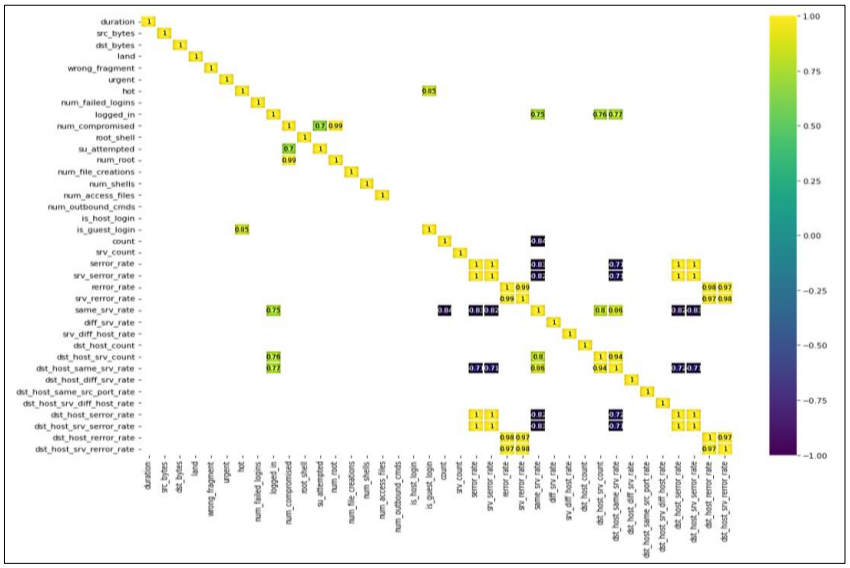

Figure 2: Correlation Matrix

Being the most used protocol, it is observed that TCP has the highest number of good and bad connections among the given data set. There are very few cases here a root shell has been obtained. This is because the root shell is generally used by system administrators only. For attackers, they need to get access to the box as a user before escalating their privileges to root. A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Networks. It was further observed that whenever there was a case of root shell access, a buffer overflow attack was encountered. Reason being when escalating privileges from user to root, generally only 2 types of attack are possible, mis-configuration in permissions or buffer overflows. Buffer overflows are more common, as most Kernel exploits are buffer overflows. In Neptune attacks, the attacker sends a flood of SYN packets and the target sends back SYN-ACK packets in reply. From this, the attacker comes to know that the target is alive and sends a packet with REJ and S0 flags. Furthermore, there are a lot of packets with SF flags in normal secure connections.

3.4 Model Building

The problem of IDS is taken as a binary classification problem. The organizations are more concerned about Normal and Bad connections getting classified correctly rather than each of the bad categories getting misclassified so that no Bad connections are allowed to gain access to the internal network of the organization by getting misclassified as a normal connection, which may otherwise result in a security threat.

The steps for Max-Voting Ensemble Model building include;

- Train each model here, the models taken are Gaussian Naive Bayes, DT, XGBoost.
- Make Predictions from Each Model.
- Implement the Max-Voting method.
- Evaluate the Ensemble method performance.

The ensemble ids max voting function takes the training data features (X train), the training data labels (y train), and the testing data features (X test) as input. Three individual models are defined: dt clf for decision tree, gnb clf for Gaussian Naive Bayes, and xgb clf for XGBoost. Each model is trained on the provided training data (X train, y train). Prediction: Each model makes predictions on the testing data (X test), resulting in separate prediction arrays dt preds, gnb preds, and xgb preds. Max Voting: The np.bincount function counts the occurrences of each predicted class (normal or intrusion) across all three models. np.argmax is used to find the class with the most votes, which becomes the ensemble prediction for each data point in the test set. The function returns the final ensemble predictions (ensembl preds) as a NumPy array. The confusion matrix of the Max-Vote Model is depicted in Figure 3. The model comparison is present in Table 1. From the Table, we infer that XGBoost stands out as the best individual model with the highest accuracy and the lowest number of false positives. The Decision Tree also performs well, especially when considering the balance between accuracy and false positives. The Max Voting Technique shows the effectiveness of ensemble methods in achieving high accuracy, though it still trails slightly behind XGBoost in A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Network terms of false positives. Gaussian Naive Bayes, while much simpler, lags significantly in performance, highlighting the trade-off between model complexity and accuracy.

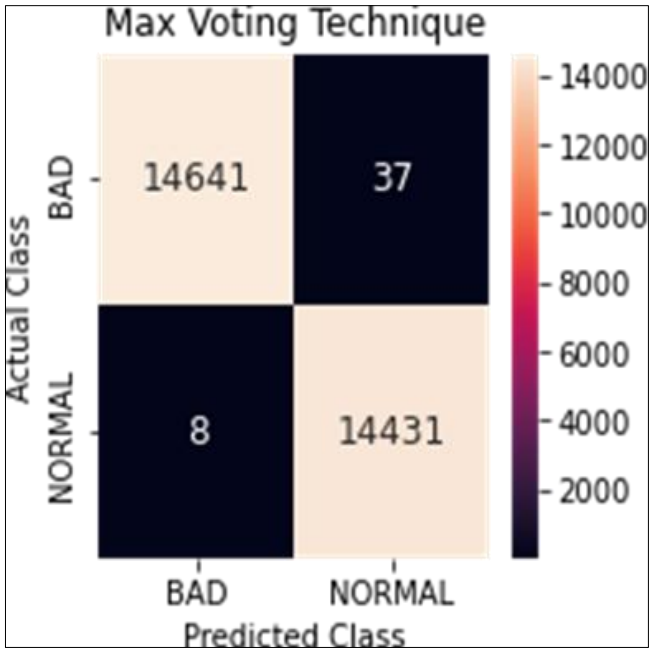

Figure 3: Confusion Matrix.

Table 1: Comparison of Different Techniques

Technique	Accuracy	False Positives
Gaussian Naive Bayes	96.03	462
Decision Tree	99.71	54
XGBoost	99.86	25
Max Voting Technique	99.85	37

4. CONCLUSION

This paper explored the application of ensemble learning with max voting for intrusion detection in IoT networks. By combining the predictions of DT, GNB, and XGBoost, the approach aimed to achieve more accurate and reliable intrusion detection compared to using individual models. The strengths of different models are considered, and ensemble learning captures complex relationships within the data, leading to better intrusion detection accuracy of 99.85%. Combining diverse models can help mitigate the biases inherent in individual models, resulting in more robust and generalizable intrusion detection. The approach can be extended to incorporate additional learning models or adapt to evolving network threats in real-time IoT applications. A Hybrid Ensemble Framework for Intrusion Detection in Internet of Things Networks.

REFERENCES

1. Musleh D, Alotaibi M, Alhaidari F, Rahman A, Mohammad RM. Intrusion detection system using feature extraction with machine learning algorithms in IoT. Journal of Sensor and Actuator Networks 2023;12(2):29.
2. Abiodun OI, Abiodun EO, Alawida M, Alkhawaldeh RS, Arshad H. A review on the security of the Internet of Things: Challenges and solutions. Wireless Personal Communications 2021;119:2603–2637.
3. Heidari A, Jabraeil Jamali MA. Internet of Things intrusion detection systems: A comprehensive review and future directions. Cluster Computing 2023;26(6):3753–3780.
4. Abdullah A, Hamad R, Abdulrahman M, Moala H, Elkhediri S. Cybersecurity: A review of Internet of Things (IoT) security issues, challenges, and techniques. 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS) 2019:1–6. IEEE.
5. Boussard M, Thai Bui D, Douville R, Justen P, Le Sauze N, Peloso P, Vandeputte F, Verdot V. Future spaces: Reinventing the home network for better security and automation in the IoT era. Sensors 2018;18(9):2986.
6. Tabassum A, Erbad A, Guizani M. A survey on recent approaches in intrusion detection system in IoTs. 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC) 2019:1190–1197. IEEE.
7. Eskandari M, Janjua ZH, Vecchio M, Antonelli F. Passban IDS: An intelligent anomaly-based intrusion detection system for IoT edge devices. IEEE Internet of Things Journal 2020;7(8):6882–6897.
8. Otoum Y, Liu D, Nayak A. DL-IDS: A deep learning–based intrusion detection framework for securing IoT. Transactions on Emerging Telecommunications Technologies 2022;33(3):3803.
9. Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Al-Nemrat A, Venkatraman S. Deep learning approach for intelligent intrusion detection system. IEEE Access 2019;7:41525–41550.
10. Özgür A, Erdem H. A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015. 2016.
11. Ali Z, Hussain F, Ghazanfar S, Husnain M, Zahid S, Shah GA. A generic machine learning approach for IoT device identification. 2021 International Conference on Cyber Warfare and Security (ICCWS) 2021:118–123. IEEE.
12. Sapre S, Ahmadi P, Islam K. A robust comparison of the KDDCup99 and NSL-KDD IoT network intrusion detection datasets through various machine learning algorithms. arXiv preprint arXiv:1912.13204 (2019).
13. Zhang J, Zulkernine M, Haque A. Random-forests-based network intrusion detection systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 2008;38(5):649–659.
14. Huda S, Abawajy J, Alazab M, Abdollalihian M, Islam R, Yearwood J. Hybrids of support vector machine wrapper and filter-based framework for malware detection. Future Generation Computer Systems 2016;55:376–390.