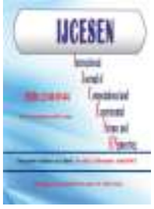
International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.3 (2025) pp. 6407-6417
http://www.ijcesen.com



ISSN: 2149-9144

Research Article

Leveraging Random Forest to Detect Botnet Attacks in IoT Environments

Hussien Alrakah[1], Yagoub Abbker Adam[2], Mohammed Abdalraheem[3], Phiros Mansur[4], Shaik Rizwan[5] and Ibrahim Al-Shourbaji[6]*

[1]Department of Electrical and Electronics Engineering , Jazan University, Jazan, 45142, Saudi Arabia
Email: hussei2n@gmail.com - **ORCID:** 0000-0002-5247-7805

[2]Department of Computer Science, Jazan University, Jazan, 45142, Saudi Arabia
Email: yagou2b@gmail.com - **ORCID:** 0000-0002-5247-7815

[3]Department of Computer Science, Jazan University, Jazan, 45142, Saudi Arabia
Email: mohamme2d@gmail.com - **ORCID:** 0000-0002-5247-7825

[4]Department of Electrical and Electronics Engineering , Jazan University, Jazan, 45142, Saudi Arabia
Email: phiro2s@gmail.com - **ORCID:** 0000-0002-5247-7835

[5]Department of Electrical and Electronics Engineering , Jazan University, Jazan, 45142, Saudi Arabia
Email: rizwa2n@gmail.com - **ORCID:** 0000-0002-5247-7845

[6]Department of Electrical and Electronics Engineering , Jazan University, Jazan, 45142, Saudi Arabia
* **Corresponding Author Email:** alshourbajiibrahim@gmail.com - **ORCID:** 0000-0002-5247-7865

Article Info:

DOI: 10.22399/ijcesen.3717
Received : 11 June 2025
Accepted : 18 August 2025

Keywords

Botnet, IoT
Detection
Machine learning
Cyber-security

Abstract:

Because of their secrecy and capacity to manage vast networks of hacked devices, botnet assaults have grown into a more serious and severe threat to Internet of Things (IoT) devices. The identification of botnet attacks is extremely difficult due to their spread nature and covert activity. IoT devices usually operate with insufficient security safeguards and are vulnerable to these types of assaults. In recent years, machine learning (ML) techniques have shown a lot of promise for identifying and stopping various kinds of cyberattacks. This study accurately detects botnet attacks in Internet of Things environments using a Random Forest (RF)-based approach. The RF model is evaluated on two publicly available datasets designed specifically for botnet discovery. Experimental results show that RF outperforms several other popular models in terms of F1-score, recall, accuracy, and precision. These outcomes show how resilient and effective the RF algorithm is as a practical and reliable method of enhancing IoT device security.

1. Introduction

In recent years, the Internet of Things (IoT) has improved everyday life in a sustainable way; hence, drawn increasing interest from researchers. IoT is a network of intelligent devices, or "things" that gather, process, analyze, and transmit data [1,2]. A growing number of internet-connected devices has spread IoT technologies across areas such as smart farming, smart cities, modern healthcare, and intelligent transportation [3,4]. IoT devices are limited in terms of memory and computing power, and they must adjust to a variety of situations and surroundings. An IoT device can be uniquely identified on the network, enabling interaction and communication between them and with people. This growing IoT network comprises the interconnection of wireless communication technologies, sensors, actuators, and smart devices [5].

IoT devices are vulnerable to hackers and attackers due to poor security and standards. The hacked IoT devices are used as part of a bigger IoT botnet. Using a hacked IoT device, hackers may attain total control of the network or attack other devices in the network. The development of IoT botnets is more dangerous than several criminal activities and is increasing rapidly [6]. In recent events, these botnets generated more than one terabit per second (Tbps) bandwidth and resulting in service disruptions by major internet services giants like CNN, Netflix, Guardian, and Twitter, and interruption of the Internet.

Over the past decade, machine learning (ML) models and datasets have investigated for IoT botnet detection [7-10]. For creating botnet detectors, the salient features are identified using a feature selection (FS) method. The salient features are used to build machine learning (ML) models based botnet detectors. Pavaiyarkarasi et al. [9] reported a unique feature selection (FS) metric using the wrapper method to effectively select the salient features. Nguyen et al. [11] investigated a rooted subgraph-based technique for IoT botnet detection.

Alani [12] combined FS and explainable ML models to create realistic datasets and improve the accuracy of ML models trained on them for effective packet-based botnet detection. Joshi et al. [13] investigated a combination of fuzzy-based feature engineering and artificial neural network (ANN) for improving botnet detection. Fuzzy model identified the salient features in the CTU-13 dataset, and these features were used to train the ANN. Kalakoti et al. [14] investigated ML models aligned aligned with the different phases of the botnet life cycle. Six separate binary and multiclass ML models were determined by a combination of wrapper and filter-based FS methods. Jeelani et al. [15] explored the IoT-23 dataset and reported that ML models could identify anomalous network activity.

Soe et al. [16] investigated data resampling to alleviate dataset imbalance and improve the ANN-based botnet detection. In order to identify DDoS attacks in IoT systems, Aamir and Zaidi [17] investigated combinations of ML models and FS methods for enhancing the distributed denial-of-service attack detection in IoT. The k-Nearest Neighbors (kNN) outperformed other competing models, and FS reduced computational cost while having minimal impact on accuracy. Bahşi et al. [18] used FS with the decision tree (DT) model to identify botnet attacks. FS was reported to enhance the time efficiency and scalability of the model. Dietz et al [19] employed a honeypot-based ML model to characterise hackers' behavior by luring them to identify new malware attacks within the botnet.

Several studies have been reported for botnet detection, most of which combine FS with an ML model. To the best of our knowledge, the random forest model's potential for FS and building a robust classifier model has not been explored for botnet detection. RF extracts salient features from large datasets and minimizes overfitting. RF is selected for dotnet detection due to its accurate prediction, generalization capacity, scalability, ease of training, and the ability to deal with data with a wide variety of features. In the present study, the RF model is evaluated using two datasets for botnet detection, using a number of quantitative metrics to assess its performance. It is compared with kNN, support vector machine (SVM), ANN, and DT.

This paper's remaining sections are organized as follows: An overview of the RF models and the datasets description is provided in section 2. Section 3 presents the experimental data, assessment metrics, and comments. Section 4 presents the findings of this study together with recommendations for further research.

2. Methods And Materials

2.1Random Forest (RF)

Schonlau and Zou [20] presented RF, an ensemble model. Because of its quick training, capacity to handle complicated datasets with ease, and appropriateness for regression and prediction problems, it became well-liked in scientific and technical applications. Ma and Zhang [21], after generating a large number of non-pruned decision trees, RF uses majority voting to aggregate the findings. To improve the variety of the trees, each one is constructed using bootstrap data taken from the training data. Conversely, the samples that are not used in the building stages are referred to as "Out-Of-Bag" (OOB) data. During the training phase, the algorithm internally uses this OOB data as validation data. Boulesteix and associates [22] In each split of node of a decision tree, a small number of input variables (features) are randomly selected rather than selecting all the features (random feature selection). This procedure is repeated to create a large number of decision trees that result in a randomly generated forest. The primary disadvantage of RF is that many trees make it slow for real-time applications. RF prefers features with more categories and smaller correlated groups.

2.2 Datasets

2.2.1 Dataset 1

The N-BaIoT comprises both normal and botnet traffic labeled to differentiate between criminal and lawful activity and was developed at the UNSW Canberra cyber center to mimic several botnet situations in an IoT network [23]. To simulate several forms of malicious attacks, including DDoS, DoS, OS, data exfiltration, and keylogging attacks, the developers used a number of virtual computers on the internal network. The dataset also contained more DoS and DDoS attacks. With more than 72 million records, the N-BaIoT dataset captures a broad spectrum of harmful activity.

However, the BoT-IoT dataset, which is associated with this work, provides a realistic testbed environment and classifies the data according to attack types. A subset of the N-BaIoT dataset, including 999,610 records, was utilized in the particular study you referred to. Of these, 994,828 were samples of botnet activity, while the remaining records represented normal traffic. 115 real-valued characteristics are used to represent each traffic record in the collection. Bashlite and Mirai botnets are used to inject various assaults, creating the dataset. IoT devices running Linux can be infected with Bashlite to launch DDoS attacks. Additionally, Mirai, which uses IoT devices to launch massive assaults, was found in August 2016 and is currently open-source on https://www.kaggle.com/datasets/mkashifn/nbaiot-dataset/code . Over time, botnets have evolved and have become a greater threat [24].

2.2.2 Dataset 2

The constraints of current datasets in terms of representing intricate network topologies with real IoT devices have been addressed. The goal of the CICIoT2023 dataset is to offer a real-time dataset created especially for extensive assaults in Internet of Things settings [25]. With 33 assaults carried out on 105 IoT devices, this dataset presents a substantial IoT attack dataset. Seven categories are used to classify the attacks: Mirai, Spoofing, Recon, Brute Force, DoS, DDoS, and Web-based. This dataset enables a more thorough investigation and assessment of security analytics tools in actual IoT operations by include a range of threat types. In the CICIoT2023 dataset, 47 real-valued attributes are used for describing each traffic record. These capabilities most likely record pertinent data on network traffic, IoT device activity, and other attributes that might help with security analysis tool design and evaluation.

A network tap was used to gather this dataset, and two traffic monitors were put up specifically to keep an eye on network traffic. Each packet delivered over the network is saved on a different computer. Two distinct interfaces on the network are linked to two additional monitoring ports that route incoming traffic to these PCs. As a result, the network traffic is recorded in p-cap format and viewed with Wireshark. Merge-cap is used to unify p-cap files for every experiment since two data streams are stored. A distinct experiment is conducted for every attack, aiming at all relevant devices. Invasive IoT devices carry out attacks on susceptible IoT devices in every case. For instance, web-based assaults target devices that enable web applications, while DDoS attacks are launched across all devices [26].

Table 1. *The characteristics of the datasets.*

Dataset	Source	No. of features	No. of samples
N-BaIoT	[19]	115	999,610
CICIoT2023	[20]	47	10,340,161

The fundamental features of the N-BaIoT and CICIoT2023 datasets are outlined in Table 1. Additionally, using iterative FS techniques like the MH algorithm might become computationally costly because to the enormous quantity of traffic records in the datasets. Therefore, only 10% of the dataset is utilized for FS assessment in order to reduce the computational overhead and provide a representative sample of both normal and botnet traffic for evaluation purposes.

3. Experimental Results

3.1 Experimental setup

Five distinct classifiers are used to create the botnet detection models. Table 2 lists the hyperparameter values used for the Python scikit-learn method, which is used to execute all of the models. An Ubuntu 22.04.1 LTS operating system, 32 GB of RAM, and a 3.13 GHz PC are used for all trials.

Table 2. *The hyperparameters of all models for botnet detection.*

Model	Hyperparameters
KNN	#Neighbors=50, metric=Euclidean distance, weights= uniform
SVM	Kernel= radial basic function, gamma= 0.01, regularization=10,
ANN	Batch_size = 200, activation= ReLU, Hidden layers = (20, 2), solver= Adam
DT	Max_features= 10, min_samples_split=2, impurity_criterion=gini
RF	#Estimators=200, criterion=gini, min_samples_split=5, max_features= 14

3.2Evaluation measures

Various measures can be used to evaluate the efficiency of different classifiers, and they are calculated as follows:

$$AC = \frac{TP + TN}{TP + TN + FN + FP} \qquad (1)$$

$$\text{Recall} \ = \frac{TP}{TP + FN} \qquad (2)$$

$$\text{Precision} \ = \frac{TP}{TP + FP} \qquad (3)$$

$$\text{F1-score} \ = \frac{2\,P\,R}{P + R} \qquad (\ 4)$$

Accuracy (AC): It is a ratio of correctly classified samples to the total samples being evaluated, i.e., the total of True Positives (TP) and True Negatives (TN) samples, to the total number of samples, i.e., the sum of TP, TN, False Negatives (FN), and False Positives (FP).

Recall: It quantifies the model's effectiveness in capturing the positive samples. Recall, sensitivity, or true positive rate, is a ratio of TP to the total number of real positive samples (sum of TP and FN).

Precision: It measures the reliability of positive predictions by calculating the ratio of TP to the total predicted positives, which is the sum of TP and FP. It indicates the likelihood that a positive prediction is accurate..

Precision: It measures the model's efficiency in predicting the positive samples. It is a ratio of TP to the total number of positive predictions (sum of TP and FP).

F1-score: It combines recall and precision into a single metric and is particularly useful when dealing with imbalanced datasets. F1-score is the harmonic mean of precision and recall.

These metrics help assess the performance of classification models and are commonly reported in the literature for botnet detection.

3.3 Experimental results and discussion

The five most frequently applied classifiers and two commonly used datasets in botnet detection literature are used for evaluation [27-31].

A comparison of five ML models, KNN, DT, SVM, RF, and ANN, for botnet detection using the BoTIoT and CICIoT2023 datasets is presented in Table 3. Classifiers were trained on a 10% subset of the datasets. Across all evaluated metrics, the results consistently outperform those of the baseline methods. For the BoTIoT dataset, the RF achieved an F-score of 0.9620 with FS, compared to 0.9262 when using FS in the baseline setup, indicating RF's strong capability in detecting botnets. Accuracy values for different models on the BoTIoT dataset range from 0.9532 to 0.9856. With an accuracy of 0.9856, RF outperformed the other classifiers, followed by ANN and KNN, DT, and SVM. RF has the highest recall of 0.9543, followed by DT, ANN, KNN, and SVM. The maximum precision of 0.9698 was attained by RF, followed by DT, KNN, ANN, and SVM. The greatest F1-score, 0.9620, was attained by RF, followed by DT, ANN, KNN, and SVM.

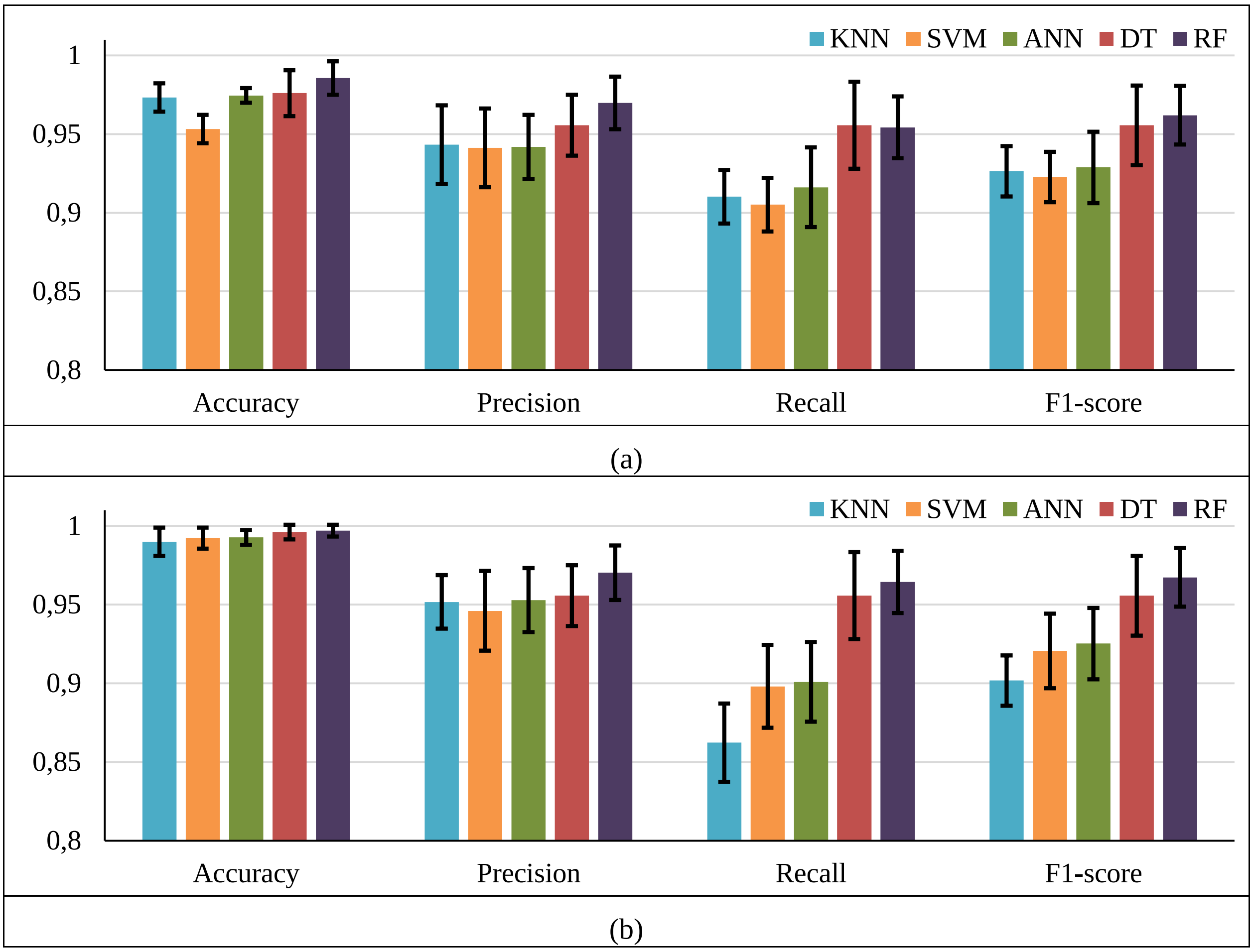
With an accuracy of 0.9970, RF outperformed DT (0.9960) and ANN (0.9927) for the CICIoT2023 dataset. SVM (0.9823) and KNN (0.9899) both did well, albeit their accuracy was a little bit lower. Similarly, with a precision of 0.9703, RF was the most accurate, followed by ANN (0.9529) and DT (0.9557). The accuracy ratings of SVM (0.9460) and KNN (0.9517) were comparatively high. With a recall of 0.9644, RF was the most successful, followed by DT (0.9556) and ANN (0.9008). The recall values of KNN (0.8622) and SVM (0.8980) were marginally lower. With an F1-score of 0.9673, RF was the highest, followed by ANN (0.9252) and DT (0.9556). The F1-scores of KNN (0.9017) and SVM (0.9206) were comparatively high. Given that it nearly works well with regard to of accuracy, precision, recall, and F1-score on both datasets, the overall findings demonstrate that RF is a useful technique for botnet identification.

Table 3. *Comparative performance analysis of the different measures with 10% subset and baseline classifiers for botnet detection using two real-world datasets.*

Dataset	Using a subset of 10%					
	Metric	KNN	SVM	ANN	DT	RF
BoTIoT	Accuracy	0.9733	0.9532	0.9745	0.9760	0.9856
	Recall	0.9102	0.9051	0.9162	0.9556	0.9543
	Precision	0.9432	0.9412	0.9418	0.9557	0.9698
	F1-score	0.9264	0.9228	0.9288	0.9556	0.9620
	Using baseline classifiers					
	Accuracy	0.9100	0.9234	0.9035	0.9240	0.9356
	Recall	0.8953	0.9020	0.8832	0.9276	0.9423
	Precision	0.9032	0.9236	0.9011	0.9247	0.9108
	F1-score	0.8992	0.9126	0.8920	0.9261	0.9262
CICIoT2023	Using a subset comprising 10%					
	Accuracy	0.9899	0.9823	0.9927	0.9960	0.9970
	Recall	0.8622	0.8980	0.9008	0.9556	0.9644
	Precision	0.9517	0.9460	0.9529	0.9557	0.9703

	F1-score	0.9017	0.9206	0.9252	0.9556	0.9673
	Using baseline classifiers					
	Accuracy	0.9579	0.9683	0.9748	0.9448	0.9362
	Recall	0.8673	0.8689	0.8808	0.9259	0.9354
	Precision	0.9117	0.9340	0.9424	0.9377	0.9503
	F1-score	0.8889	0.9003	0.9106	0.9318	0.9427

A subset of 10% of the BoTIoT and CICIoT2023 datasets is used in Figure 1 to compare the five ML models. All metrics' mean values are presented as vertical bars, while black marks indicate the standard deviation.



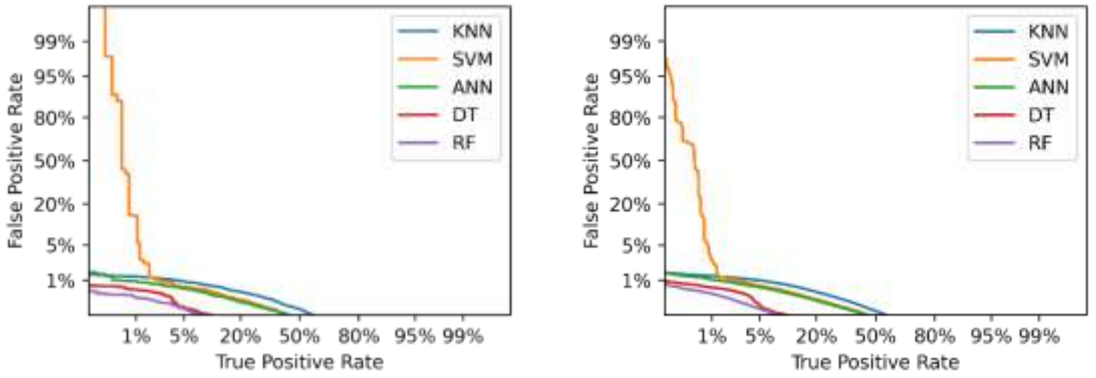
(a)

(b)

Figure 1. *Quantitative comparison of different measures of five classifiers for botnet detection using two real-world (a) BoTIoT and (b) CICIoT2023 datasets.*

All of the measures had greater values for the RF than the remaining comparison models, as shown in Figure 1. Additionally, RF has the least standard deviation, demonstrating the model's dependability. These findings validate the RF's capabilities and suggest that it may be a useful model for detecting botnets.

The performance tradeoff between a classification system's TPR and false positive rate (FPR) is visually represented by DET (Detection Error Tradeoff) curves. The DET curves for the methods KNN, DT, SVM, RF, and ANN are compared in Figure 2. The DET curves for each of the five classifiers for the BoTIoT dataset are displayed in Figure 2(a). The performance trade-off between the FPR and FNR for various threshold settings is depicted by the DET curve. You may visually evaluate these methods' respective performance in terms of the FPR and FNR tradeoff by comparing their DET curves. The precise form and arrangement of the curves might help you choose the best strategy for your classification problem by revealing the advantages and disadvantages of each method. As may be shown, RF and KNN have the lowest equal error rates among the classifiers. DT, ANN, SVM and RF has the lowest equal error rate for the



(a) (b)

Figure. 2. *DET curves of KNN, SVM, ANN, DT, and RF for botnet detection using subset of 10% for (a) BoTIoT and (b) CICIoT2023 datasets.*

CICIoT2023 dataset as well, while the performance of the other classifiers is ranked in the same order as for the earlier dataset.

Figure 3 calculates and displays the computational time in seconds required to train and test the models employed in this study. Figure 3 shows that, in comparison to alternative approaches, the RF model needed a lower average execution time with a 10% subset on the BoTIoT and CICIoT2023 datasets. The ANN model took longer on the BoTIoT dataset on both the 10% subset and as a baseline approach, but the SVM took longer on the CICIoT2023 dataset. This suggests that the RF model is appropriate for detecting botnets.

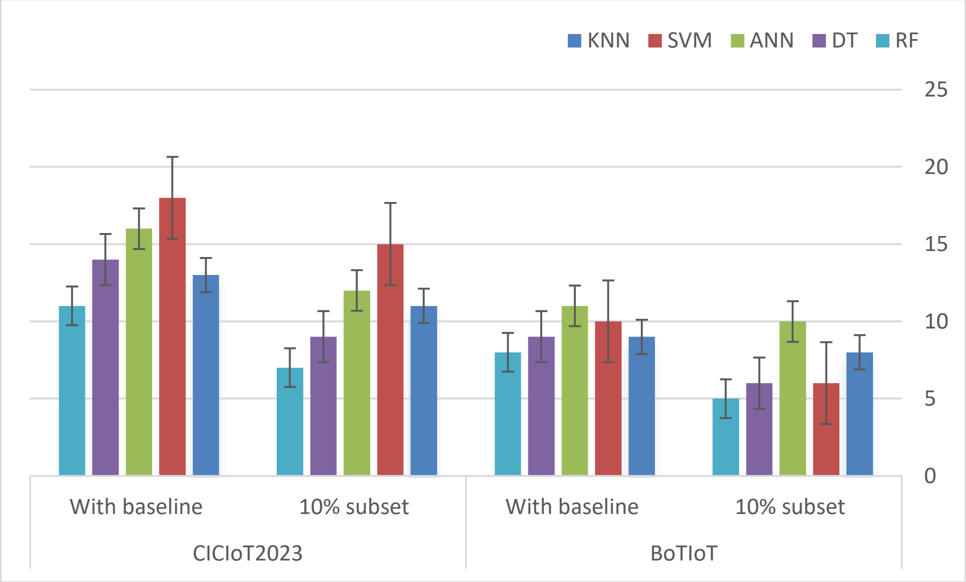


Figure 3*. Computational complexity of five classifiers for botnet detection using two real-world BoTIoT and CICIoT2023 datasets*

4. Conclusion And Future Works

Because IoT infrastructure currently lacks strong security, the worldwide existence of IoT has given hackers the chance to take advantage of the network's security and privacy by exploiting anomalous entities like botnets. However, because ML models can uncover hidden patterns in data and determine links between them, they can make IoT devices smarter and far more efficient at making choices. In this work, RF was utilized to detect botnets in an Internet of Things context. Two open source datasets and a set of assessment measures are used to assess the RF. Its performance is also contrasted with that of other well-known ML models. The outcomes demonstrated that the RF model outperforms the other techniques in detecting botnets. The datasets contain a number of drawbacks, such as being large and unbalanced, among other problems that must be fixed for botnet identification. Big data, signal processing, and intrusion detection are just a few of the prospective uses for the RF model. Exploring further metaheuristic techniques to be used as feature selection in Botnet detection is another avenue that may be pursued, since these optimization algorithms have demonstrated significant promise in other fields. Last but not least, the RF model may be used and evaluated on a sizable IoT-based online tracking system to identify botnets using real-time data provided from IoT devices, gauge its effectiveness, and precisely address various attacks.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Lampropoulos, G., Siakas, K., & Anastasiadis, T. (2018). Internet of Things (IoT) in industry: Contemporary application domains, innovative technologies and intelligent manufacturing. *People, 6*(7).

[2] Wójcicki, K., Biegańska, M., Paliwoda, B., & Górna, J. (2022). Internet of Things in industry: Research profiling, application, challenges and opportunities—A review. *Energies, 15*(5), 1806.

[3] Asghari, P., Rahmani, A. M., & Javadi, H. H. S. (2019). Internet of Things applications: A systematic review. *Computer Networks, 148*, 241–261.

[4] Gulati, K., Boddu, R. S. K., Kapila, D., Bangare, S. L., Chandnani, N., & Saravanan, G. (2022). A review paper on wireless sensor network techniques in Internet of Things (IoT). *Materials Today: Proceedings, 51*, 161–165.

[5] Kumar, R., Rani, S., & Awadh, M. A. (2022). Exploring the application sphere of the Internet of Things in Industry 4.0: A review, bibliometric and content analysis. *Sensors, 22*(11), 4276.

[6] Wazzan, M., Algazzawi, D., Bamasaq, O., Albeshri, A., & Cheng, L. (2021). Internet of Things botnet detection approaches: Analysis and recommendations for future research. *Applied Sciences, 11*(12), 5713.

[7] Pour, M. S., Mangino, A., Friday, K., Rathbun, M., Bou-Harb, E., Iqbal, F., ... & Erradi, A. (2019). Data-driven curation, learning and analysis for inferring evolving IoT botnets in the wild. In *Proceedings of the 14th International Conference on Availability, Reliability and Security* (pp. 1–10).

[8] Alqahtani, M., Mathkour, H., & Ben Ismail, M. M. (2020). IoT botnet attack detection based on optimized extreme gradient boosting and feature selection. *Sensors, 20*(21), 6336.

[9] Pavaiyarkarasi, R., Manimegalai, T., Satheeshkumar, S., Dhivya, K., & Ramkumar, G. (2022). A productive feature selection criterion for Bot-IoT recognition based on random forest algorithm. In *2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT)* (pp. 539–545).

[10] Elsayed, N., ElSayed, Z., & Bayoumi, M. (2023). IoT botnet detection using an economic deep learning model. *arXiv preprint arXiv:2302.02013*.

[11] Nguyen, H. T., Ngo, Q. D., Nguyen, D. H., & Le, V. H. (2020). PSI-rooted subgraph: A novel feature for IoT botnet detection using classifier algorithms. *ICT Express, 6*(2), 128–138.

[12] Alani, M. M. (2022). BotStop: Packet-based efficient and explainable IoT botnet detection using machine learning. *Computer Communications, 193*, 53–62.

[13] Joshi, C., Ranjan, R. K., & Bharti, V. (2022). A fuzzy logic-based feature engineering approach for botnet detection using ANN. *Journal of King Saud University–Computer and Information Sciences, 34*(9), 6872–6882.

[14] Kalakoti, R., Nõmm, S., & Bahsi, H. (2022). In-depth feature selection for the statistical machine learning-based botnet detection in IoT networks. *IEEE Access, 10*, 94518–94535.

[15] Jeelani, F., Rai, D. S., Maithani, A., & Gupta, S. (2022). The detection of IoT botnet using machine learning on IoT-23 dataset. In *2022 2nd IEEE International Conference on Innovative Practices in Technology and Management (ICIPTM)* (pp. 634–639).

[16] Soe, Y. N., Santosa, P. I., & Hartanto, R. (2019). DDoS attack detection based on simple ANN with SMOTE for IoT environment. In *2019 IEEE 4th International Conference on Informatics and Computing (ICIC)* (pp. 1–5).

[17] Aamir, M., & Zaidi, S. M. A. (2019). DDoS attack detection with feature engineering and machine learning: The framework and performance evaluation. *International Journal of Information Security, 18*, 761–785.

[18] Bahşi, H., Nõmm, S., & La Torre, F. B. (2018). Dimensionality reduction for machine learning-based IoT botnet detection. In *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)* (pp. 1857–1862).

[19] Dietz, C., Castro, R. L., Steinberger, J., Wilczak, C., Antzek, M., Sperotto, A., & Pras, A. (2018). IoT-botnet detection and isolation by access routers. In *2018 9th IEEE International Conference on the Network of the Future (NOF)* (pp. 88–95).

[20] Schonlau, M., & Zou, R. Y. (2020). The random forest algorithm for statistical learning. *The Stata Journal, 20*(1), 3–29.

[21] Zhang, C., & Ma, Y. (Eds.). (2012). *Ensemble machine learning: Methods and applications*. Springer.

[22] Boulesteix, A. L., Janitza, S., Kruppa, J., & König, I. R. (2012). Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2*(6), 493–507.

[23] Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems, 100*, 779–796.

[24] Abbasi, F., Naderan, M., & Alavi, S. E. (2021). Anomaly detection in Internet of Things using feature selection and classification based on logistic regression and artificial neural network on N-BaIoT dataset. In *2021 5th International Conference on Internet of Things and Applications (IoT)* (pp. 1–7).

[25] Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A. A. (2023). CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. *Preprints.org, 2023050443*.

[26] Thereza, N., & Ramli, K. (2023). Development of intrusion detection models for IoT networks utilizing CICIoT2023 dataset. In *2023 3rd International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS)* (pp. 66–72).

[27] Arshad, A., Jabeen, M., Ubaid, S., Raza, A., Abualigah, L., Aldiabat, K., & Jia, H. (2023). A novel ensemble method for enhancing Internet of Things device security against botnet attacks. *Decision Analytics Journal, 8*, 100307.

[28] Saied, M., Guirguis, S., & Madbouly, M. (2023). A comparative analysis of using ensemble trees for botnet detection and classification in IoT. *Scientific Reports, 13*(1), 21632.

[29] Akash, N. S., Rouf, S., Jahan, S., Chowdhury, A., & Uddin, J. (2022). Botnet detection in IoT devices using random forest classifier with independent component analysis. *Journal of Information and Communication Technology, 21*(2), 201–232.

[30] Wang, Z., Chen, H., Yang, S., Luo, X., Li, D., & Wang, J. (2023). A lightweight intrusion detection method for IoT based on deep learning and dynamic quantization. *PeerJ Computer Science, 9*, e1569.

[31] Sakthipriya, N., Govindasamy, V., & Akila, V. (2023). A comparative analysis of various dimensionality reduction techniques on N-BaIoT dataset for IoT botnet detection. In *2023 2nd International Conference on Paradigm Shifts in Communications, Embedded Systems, Machine Learning and Signal Processing (PCEMS)* (pp. 1–6).