Journal of Science Engineering Technology and Management Science
Volume 02, Issue 06, June 2025 *ISSN: 3049-0952*
www.jsetms.com *DOI:10.63590/jsetms.2025.v02.i06.pp114-124*

Behavioural Analysis for Security Threat Detection: Machine Learning Classifier Comparison

Dr. S. Venkata Achuta Rao[1*], G.Rahul[2], S.Purushotham[2], R.Rohith[2], A.Akhil[2]
[1,2]Department of Computer Science and Engineering(Cyber Security), Sree Dattha Group of Institutions, Sheriguda, Hyderabad, Telangana.
*Corresponding author: Dr. S. Venkata Achuta Rao

To Cite this Article
Dr. S. Venkata Achuta Rao, G.Rahul, S.Purushotham, R.Rohith, A.Akhil, "Behavioural Analysis for Security Threat Detection: Machine Learning Classifier Comparison", Journal of Science Engineering Technology and Management Science, Vol. 02, Issue 06, June 2025,pp:114-124, DOI: http://doi.org/10.63590/jsetms.2025.v02.i06.pp114-124
Submitted: 20-04-2025 *Accepted: 27-05-2025* *Published: 06-06-2025*

ABSTRACT
There are an increasing number of Internet of Things (IoT) devices connected to the network these days, and due to the advancement in technology, the security threads and cyberattacks, such as botnets, are emerging and evolving rapidly with high-risk attacks. IoT-based botnet attack is one of the most popular, spreads faster and create more impact than other attacks. In recent years, several works have been conducted to detect and avoid this kind of attacks by using novel approaches. Hence, a plethora of relevant of relevant models, methods, and etc. have been introduced over the past few years, with quite a reasonable number of studies reported in the research domain. Many studies are trying to protect against these botnet attacks on the IoT environment. However, there are many gaps still existing to develop an effective detection mechanism. These attacks disrupt IoT transition by disrupting networks and services for IoT devices. Many recent studies have proposed ML and DL techniques for detecting and classifying botnet attacks in the IoT environment. This work proposes machine learning methods for classifying binary classes i.e., Benign, or TCP attack. A complete machine learning pipeline is proposed, including exploratory data analysis, which provides detailed insights into the data, followed by preprocessing. During this process, the data passes through several fundamental steps. A random forest, k-nearest neighbour, support vector machines, and a logistic regression model are proposed, trained, tested, and evaluated on the dataset. In addition to model accuracy, F1-score, recall, and precision are also considered.

Keywords: Security Threat Detection, Behavioural Analysis, Binary Classification, Cyber Threats, Intrusion Detection System (IDS).

1. INTRODUCTION

The general idea of the Internet of Things (IoT) is to allow for communication between human-to-thing or thing-to-thing(s). Things denote sensors or devices, whilst human or an object is an entity that can request or deliver a service [1]. The interconnection amongst the entities is always complex. IoT is broadly acceptable and implemented in various domains, such as healthcare, smart home, and agriculture. However, IoT has a resource constraint and heterogeneous environments, such as low computational power and memory. These constraints create problems in providing and implementing a security solution in IoT devices. These constraints further escalate the existing challenges for IoT environment. Therefore, various kinds of attacks are possible due to the vulnerability of IoT devices. IoT-based botnet attack is one of the most popular, spreads faster and create more impact than other attacks. In recent years, several works have been conducted to detect and avoid this kind of attacks [2]–[3] by using novel approaches. Hence, a plethora of relevant of relevant models, methods, and etc. have been introduced over the past few years, with quite a reasonable number of studies reported in the research domain.

Many studies are trying to protect against these botnet attacks on the IoT environment. However, there are many gaps still existing to develop an effective detection mechanism. An intrusion detection system (IDS) is one of the efficient ways to deal with attacks. However, the traditional IDSs are often not able to be deployed for the IoT environments due to the resource constraint problem of these devices. The complex cryptographic mechanisms cannot be embedded in many IoT devices either for the same reason. There are mainly two kinds of IDSs: the anomaly and misuse approaches. The misuse-based, also called the signature-based, approach, is based on the attacks' signatures, and they can also be found in most public IDSs, specifically Suricata [4].

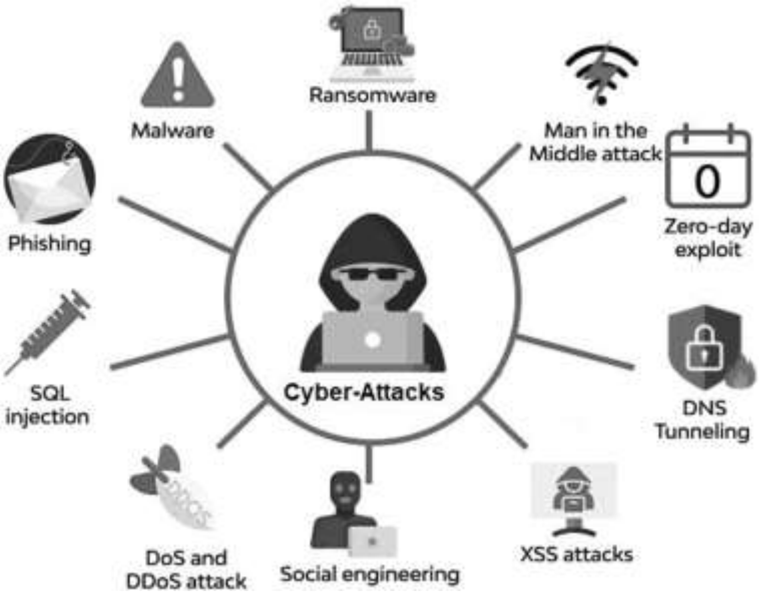


Fig. 1: Security Threat Detection.

Formally, the attacker can easily circumvent the signature-based approaches, and these mechanisms cannot guarantee to detect the unknown attacks and the variances of known attacks. The anomaly-based systems are based on normal data and can support to identify the unknown attacks. However, the different nature of IoT devices is being faced with the difficulty of collecting common normal data. The machine learning-based detection can guarantee detection of not only the known attacks and their variances. Therefore, we proposed a machine learning-based botnet attack detection architecture. We also adopted a feature selection method to reduce the demand for processing resources for performing the detection system on resource constraint devices. The experiment results indicate that the detection accuracy of our proposed system is high enough to detect the botnet attacks. Moreover, it can support the extension for detecting the new distinct kinds of attacks.

2. LITERATURE SURVEY

Soe et al. [5] adopted a lightweight detection system with a high performance. The overall detection performance achieves around 99% for the botnet attack detection using three different ML algorithms, including artificial neural network (ANN), J48 decision tree, and Naïve Bayes. The experiment result indicated that the proposed architecture can effectively detect botnet-based attacks, and also can be extended with corresponding sub-engines for new kinds of attacks.Ali et al. [6] outlined the existing proposed contributions, datasets utilised, network forensic methods utilised and research focus of the primary selected studies. The demographic characteristics of primary studies were also outlined. The result of this review revealed that research in this domain is gaining momentum, particularly in the last 3 years (2018-2020). Nine key contributions were also identified, with Evaluation, System, and Model being the most conducted. Irfan et al. [7] classified the incoming data in the IoT, contain a malware or not. In this research, this work under sample the dataset because the datasets contain imbalance class. After that, this work classified the sample using Random Forest. This work used Naive Bayes, K-Nearest Neighbor and Decision Tree too as a comparison. The dataset that has been used in this research are from UCI Machine Learning Depository's Website. The dataset showed the data traffic from the IoT Device in a normal condition and attacked by Mirai or Bashlite.Shah et al. [8] presented a concept called ‘login puzzle’ to prevent capture of IoT devices in a large scale. Login puzzle is a variant of client puzzle, which presented a puzzle to the remote device during the login process to prevent unrestricted log-in attempts. Login puzzle is a set of multiple mini puzzles with a variable complexity, which the remote device is required to solve before logging into any IoT device. Every unsuccessful log-in attempt increases the complexity of solving the login puzzle for the next attempt. This paper introduced a novel mechanism to change the complexity of puzzle after every unsuccessful login attempt. If each IoT device had used login puzzle, Mirai attack would have required almost two months to acquire devices, while it acquired them in 20 h.Tzagkarakis et al. [9] presented an IoT botnet attack detection method based on a sparsity representation framework using a reconstruction error thresholding rule for identifying malicious network traffic at the IoT edge coming from compromised IoT devices. The botnet attack detection is performed based on small-sized benign IoT network traffic data, and thus we have no prior knowledge about malicious IoT traffic data. We present our results on a real IoT-based network dataset and show the efficacy of proposed technique against a reconstruction error-based autoencoder approach.

Meidan et al. [10] proposed a novel network-based anomaly detection method for the IoT called N-BaIoT that extracts behavior snapshots of the network and uses deep autoencoders to detect anomalous network traffic from compromised IoT devices. To evaluate the method, this work infected nine commercial IoT devices in our lab with two widely known IoT-based botnets, Mirai and BASHLITE. The evaluation results demonstrated the proposed methods ability to detect the attacks accurately and instantly as they were being launched from the compromised IoT devices that were part of a botnet.Popoola et al. [11] proposed the federated DL (FDL) method for zero-day botnet attack detection to avoid data privacy leakage in IoT-edge devices. In this method, an optimal deep neural network (DNN) architecture is employed for network traffic classification. A model parameter server remotely coordinates the independent training of the DNN models in multiple IoT-edge devices, while the federated averaging (FedAvg) algorithm is used to aggregate local model updates. A global DNN model is produced after several communication rounds between the model parameter server and the IoT-edge devices. The zero-day botnet attack scenarios in IoT-edge devices are simulated with the Bot-IoT and N-BaIoT data sets.Hussain et al. [12] produced a generic scanning and DDoS attack dataset by generating 33 types of scans and 60 types of DDoS attacks. In addition, this work partially integrated the scan and DDoS attack samples from three publicly available datasets for maximum attack coverage to better train the machine learning algorithms. Afterwards, this work proposed a two-fold machine learning approach to prevent and detect IoT botnet attacks. In the first fold, this work trained a state-of-the-art deep learning model, i.e., ResNet-18 to detect the scanning activity in the premature attack stage to prevent IoT botnet attacks. While, in the second fold, this work trained another ResNet-18 model for DDoS attack identification to detect IoT botnet attacks.Abu et al. [13] proposed an ensemble learning model for botnet attack detection in IoT networks called ELBA-IoT that profiles behavior features of IoT networks and uses ensemble learning to identify anomalous network traffic from compromised IoT devices. In addition, this IoT-based botnet detection approach characterizes the evaluation of three different machine learning techniques that belong to decision tree techniques (AdaBoosted, RUSBoosted, and bagged). To evaluate ELBA-IoT, we used the N-BaIoT-2021 dataset, which comprises records of both normal IoT network traffic and botnet attack traffic of infected IoT devices.Alharbi et al. [14] proposed Gaussian distribution used in the population initialization. Furthermore, the local search mechanism was followed by the Gaussian density function and local-global best function to achieve better exploration during each generation. Enhanced BA was further employed for neural network hyperparameter tuning and weight optimization to classify ten different botnet attacks with an additional one benign target class. The proposed LGBA-NN algorithm was tested on an N-BaIoT data set with extensive real traffic data with benign and malicious target classes. The performance of LGBA-NN was compared with several recent advanced approaches such as weight optimization using Particle Swarm Optimization (PSO-NN) and BA-NN.Ahmed et al. [15] proposed a model for detecting botnets using deep learning to identify zero-day botnet attacks in real time. The proposed model is trained and evaluated on a CTU-13 dataset with multiple neural network designs and hidden layers. Results demonstrated that the deep-learning artificial neural network model can accurately and efficiently identify botnets.

3. PROPOSED METHODOLOGY

The main goal of this project is to develop a machine learning-based system capable of identifying botnet attacks within IoT device data. Botnets are networks of compromised devices controlled by malicious actors, and detecting their activities is crucial for network security.

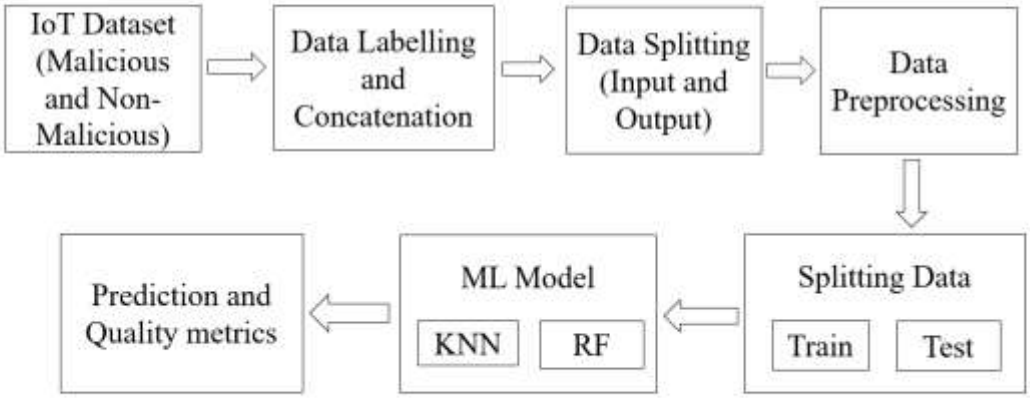


Fig. 2: Block diagram of proposed system.

The project focuses on detecting security threats in IoT environments using machine learning classifiers, with a particular emphasis on identifying malicious botnet behavior. Two datasets are used: "benign_traffic.csv" (representing normal activity) and "junk.csv" (representing malicious activity). Data is preprocessed by labeling instances as benign (0) or malicious (1) and normalizing features using Z-score standardization. Exploratory data analysis and descriptive statistics are performed to understand the datasets' characteristics. The project implements and compares three machine learning models—Support Vector Machine (SVM)**,** Random Forest (RF)**, and** K-Nearest Neighbors (KNN)—to classify network activity. The data is split into training and testing sets (80/20), and each model is evaluated using accuracy, precision, recall, F1-score, classification reports, and confusion matrices to assess performance.

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

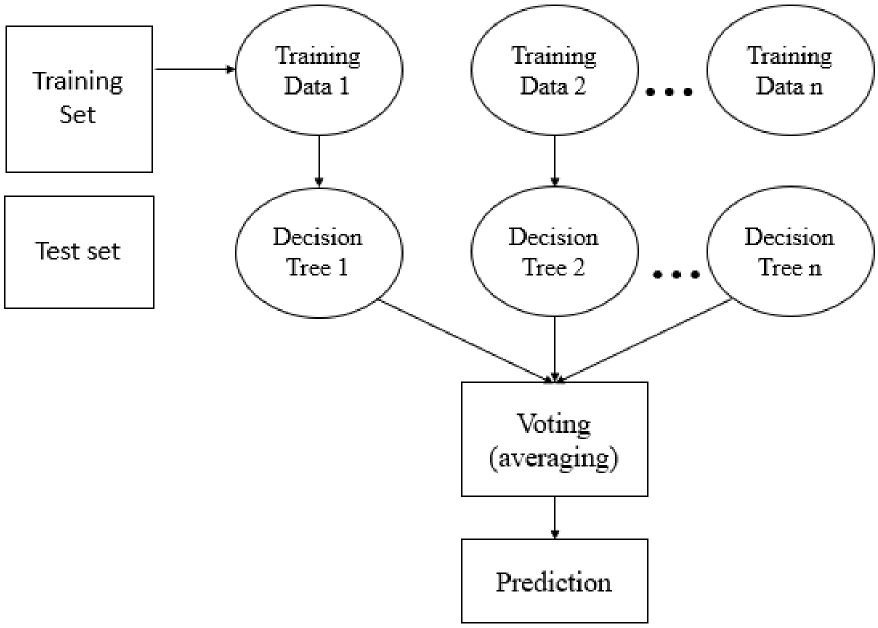


Fig. 3: Random Forest algorithm.

The Random Forest algorithm is highlighted for its ensemble-based approach using multiple decision trees trained on different subsets of the dataset. It uses majority voting to determine the final prediction, improving accuracy and reducing overfitting. Key benefits of Random Forest include its robustness to high-dimensional data, parallelization capability, resistance to overfitting, and stable performance even with missing data. As an ensemble method, Random Forest leverages bagging (bootstrap aggregation) by sampling data with replacement, training trees independently, and aggregating results through majority voting. These qualities make it a reliable and efficient classifier for detecting IoT-based cyber threats.

4. RESULTS AND DISCUSSION

Figure 4 shows a visual representation of data collected from Internet of Things (IoT) devices that are operating normally and not exhibiting any malicious behavior. It displays various features or attributes of the data points collected from these devices, such as network traffic patterns, communication protocols, and other relevant parameters. Each data point in this figure represents a non-malicious activity. Figure 5 illustrates the data collected from IoT devices that are exhibiting malicious or abnormal behavior. The visual representation highlights the anomalies or suspicious patterns in the data that indicate potential attacks or unauthorized activities. Each data point in this figure represents a malicious activity.

	MI_dir_L5_weight	MI_dir_L5_mean	MI_dir_L5_variance	MI_dir_L3_weight	MI_dir_L3_mean	MI_dir_L3_variance	MI_dir_L1_weight	MI_dir_L1_mean
0	1.000000	60.0	0.0	1.000000	60.0	0.0	1.000000	60.0
1	1.000000	60.0	0.0	1.000000	60.0	0.0	1.000000	60.0
2	1.000000	60.0	0.0	1.000000	60.0	0.0	1.000000	60.0
3	1.000000	590.0	0.0	1.000000	590.0	0.0	1.000000	590.0
4	1.927179	590.0	0.0	1.955648	590.0	0.0	1.984992	590.0

5 rows × 115 columns

MI_dir_L1_variance	MI_dir_L0.1_weight	...	HpHp_L0.1_radius	HpHp_L0.1_covariance	HpHp_L0.1_pcc	HpHp_L0.01_weight	HpHp_L0.01_mean	HpHp_L0.01_std
0.0	1.000000	...	0.000000	0.0	0.0	1.000000	60.000000	0.000000e+00
0.0	1.000000	...	0.000000	0.0	0.0	1.061357	60.000000	9.540000e-07
0.0	1.000000	...	0.000000	0.0	0.0	1.000000	60.000000	0.000000e+00
0.0	1.000000	...	9591.400096	0.0	0.0	5.832783	388.850426	9.199164e+01
0.0	1.998489	...	6354.393843	0.0	0.0	6.831901	418.293119	1.108120e+02

HpHp_L0.01_magnitude	HpHp_L0.01_radius	HpHp_L0.01_covariance	HpHp_L0.01_pcc
60.000000	0.000000e+00	0.0	0.0
60.000000	9.090000e-13	0.0	0.0
60.000000	0.000000e+00	0.0	0.0
388.850426	8.462461e+03	0.0	0.0
418.293119	1.227931e+04	0.0	0.0

Fig. 4: Illustration of sample non-malicious dataset obtained from IoT device.

	MI_dir_L5_weight	MI_dir_L5_mean	MI_dir_L5_variance	MI_dir_L3_weight	MI_dir_L3_mean	MI_dir_L3_variance	MI_dir_L1_weight	MI_dir_L1_mean
0	1.000000	98.000000	0.000000	1.000000	98.000000	0.000000e+00	1.000000	98.000000
1	1.029191	98.000000	0.000000	1.119992	98.000000	1.818989e-12	1.493231	98.000000
2	1.077270	68.295282	68.180692	1.236877	72.128396	1.585514e+02	1.889672	81.065843
3	2.038100	71.094319	42.860737	2.209694	72.975393	8.766659e+01	2.875726	78.608779
4	3.000117	72.062842	30.450564	3.184892	73.297101	6.036696e+01	3.864926	77.416316

5 rows × 115 columns

MI_dir_L1_variance	MI_dir_L0.1_weight	...	HpHp_L0.1_radius	HpHp_L0.1_covariance	HpHp_L0.1_pcc	HpHp_L0.01_weight	HpHp_L0.01_mean	HpHp_L0.01_std
0.000000e+00	1.000000	...	0.0	0.0	0.0	1.000000	98.0	0.0
3.637979e-12	1.931762	...	0.0	0.0	0.0	1.992957	98.0	0.0
2.551274e+02	2.834273	...	0.0	0.0	0.0	1.000000	66.0	0.0
1.777339e+02	3.832174	...	0.0	0.0	0.0	1.000000	74.0	0.0
1.358213e+02	4.830732	...	0.0	0.0	0.0	1.000000	74.0	0.0

HpHp_L0.01_magnitude	HpHp_L0.01_radius	HpHp_L0.01_covariance	HpHp_L0.01_pcc
98.000000	0.0	0.0	0.0
138.592929	0.0	0.0	0.0
114.039467	0.0	0.0	0.0
74.000000	0.0	0.0	0.0
74.000000	0.0	0.0	0.0

Fig. 5: Illustration of sample malicious dataset obtained from IoT device.

```
There are 19528 records with 115 features in non-malicious dataset

There are 30898 records with 115 features in malicious dataset
Adding output column in the datasets with all 0 in pureData dataset and all 1 in maliciousDataset dataset
```

	MI_dir_L5_weight	MI_dir_L5_mean	MI_dir_L5_variance	MI_dir_L3_weight	MI_dir_L3_mean	MI_dir_L3_variance	MI_dir_L1_weight	MI_dir_L1_mean
0	1.000000	60.000000	0.000000	1.000000	60.000000	0.000000	1.000000	60.000000
1	1.000000	60.000000	0.000000	1.000000	60.000000	0.000000	1.000000	60.000000
2	1.000000	60.000000	0.000000	1.000000	60.000000	0.000000	1.000000	60.000000
3	1.000000	590.000000	0.000000	1.000000	590.000000	0.000000	1.000000	590.000000
4	1.927179	590.000000	0.000000	1.955648	590.000000	0.000000	1.984992	590.000000
...	...	...	...	...	...	...	...	...
30893	166.931803	74.005716	0.137159	277.615508	74.013930	0.334630	756.938339	74.033362
30894	162.133219	74.005681	0.136313	272.788653	74.013879	0.333404	752.605073	74.033318
30895	163.124243	74.005646	0.135477	273.779592	74.013828	0.332187	753.596740	74.033274
30896	164.123165	74.005612	0.134652	274.778506	74.013777	0.330979	754.595743	74.033230
30897	165.119774	74.005578	0.133837	275.775101	74.013728	0.329779	755.592626	74.033186

50426 rows × 116 columns

MI_dir_L1_variance	MI_dir_L0.1_weight	...	HpHp_L0.1_covariance	HpHp_L0.1_pcc	HpHp_L0.01_weight	HpHp_L0.01_mean	HpHp_L0.01_std	HpHp_L0.01_magnitude	HpHp_L0.01_radius	HpHp_L0.01_covariance	HpHp_L0.01_pcc	output
0.000000	1.000000	...	0.0	0.0	1.000000	60.000000	0.000000e+00	60.000000	0.000000e+00	0.0	0.0	0
0.000000	1.000000	...	0.0	0.0	1.061357	60.000000	9.540000e-07	60.000000	9.090000e-13	0.0	0.0	0
0.000000	1.000000	...	0.0	0.0	1.000000	60.000000	0.000000e+00	60.000000	0.000000e+00	0.0	0.0	0
0.000000	1.000000	...	0.0	0.0	5.832783	388.850426	9.199164e+01	388.850426	8.462461e+03	0.0	0.0	0
0.000000	1.998489	...	0.0	0.0	6.831901	418.293119	1.108120e+02	418.293119	1.227931e+04	0.0	0.0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
4.856612	6712.484233	...	0.0	0.0	1.682482	74.000000	9.536743e-07	95.268043	9.094947e-13	0.0	0.0	1
4.850160	6708.739662	...	0.0	0.0	1.682456	74.000000	0.000000e+00	95.268043	0.000000e+00	0.0	0.0	1
4.843725	6709.732234	...	0.0	0.0	1.682464	74.000000	0.000000e+00	95.268043	0.000000e+00	0.0	0.0	1
4.837308	6710.731347	...	0.0	0.0	1.682464	74.000000	9.536743e-07	95.268043	9.094947e-13	0.0	0.0	1
4.830907	6711.728574	...	0.0	0.0	1.682464	74.000000	0.000000e+00	95.268043	0.000000e+00	0.0	0.0	1

Fig. 6: Illustration of dataset after combining both malicious and non-malicious data with labelling as benign = 0, and attack = 1.

Figure 6 depicts the dataset created by combining the non-malicious data (from Figure 4) and the malicious data (from Figure 2). This shows how the data points have been labeled, with the "benign" data points labeled as 0 and the "attack" data points labeled as 1. This labeled dataset is essential for training machine learning models to distinguish between normal and malicious activities.

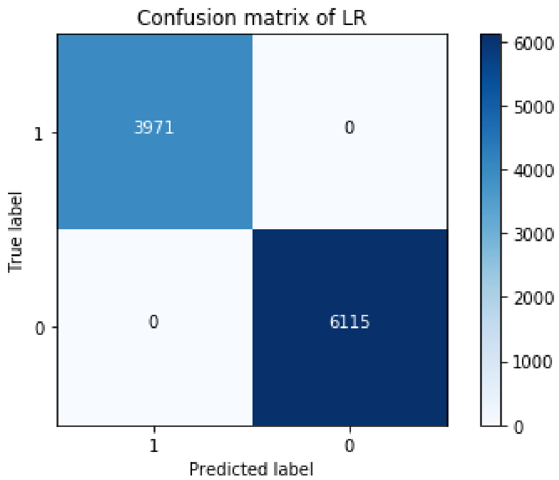
```
The accuracy score of Logistic Regression is 1.000000
Classification report :-
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3971
           1       1.00      1.00      1.00      6115

    accuracy                           1.00     10086
   macro avg       1.00      1.00      1.00     10086
weighted avg       1.00      1.00      1.00     10086

The accuracy score of SVM is 0.999802
Classification report :-
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3971
           1       1.00      1.00      1.00      6115

    accuracy                           1.00     10086
   macro avg       1.00      1.00      1.00     10086
weighted avg       1.00      1.00      1.00     10086
```

Fig. 7: Classification report of LR model, and SVM classifier for detection of botnet attack in IoT network traffic data.

```
The accuracy score of Random Forest Classifier is 1.000000
Classification report :-
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3971
           1       1.00      1.00      1.00      6115

    accuracy                           1.00     10086
   macro avg       1.00      1.00      1.00     10086
weighted avg       1.00      1.00      1.00     10086

The accuracy score of K-nearest neighbour is 0.999901
Classification report :-
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3971
           1       1.00      1.00      1.00      6115

    accuracy                           1.00     10086
   macro avg       1.00      1.00      1.00     10086
weighted avg       1.00      1.00      1.00     10086
```

Fig. 8: Classification report of RF, and KNN classifiers for detection of botnet attack in IoT network traffic data.

Figure 7 and Figure 8 presents the classification performance report of two machine learning models, namely LR model, SVM classifier, RF model, and KNN classifier for the specific task of detecting botnet attacks in IoT network traffic data. The report includes metrics such as accuracy, precision, recall, and F1-score, showing how well each model performed in identifying botnet attacks. Figure 6 demonstrate the obtained confusion matrices using various ML models. Confusion matrices are used to assess the performance of classification algorithms. Each matrix shows the number of true positives, true negatives, false positives, and false negatives for a given model's predictions. By comparing these matrices, we can evaluate which model is performing better in terms of correctly identifying normal and malicious activities.



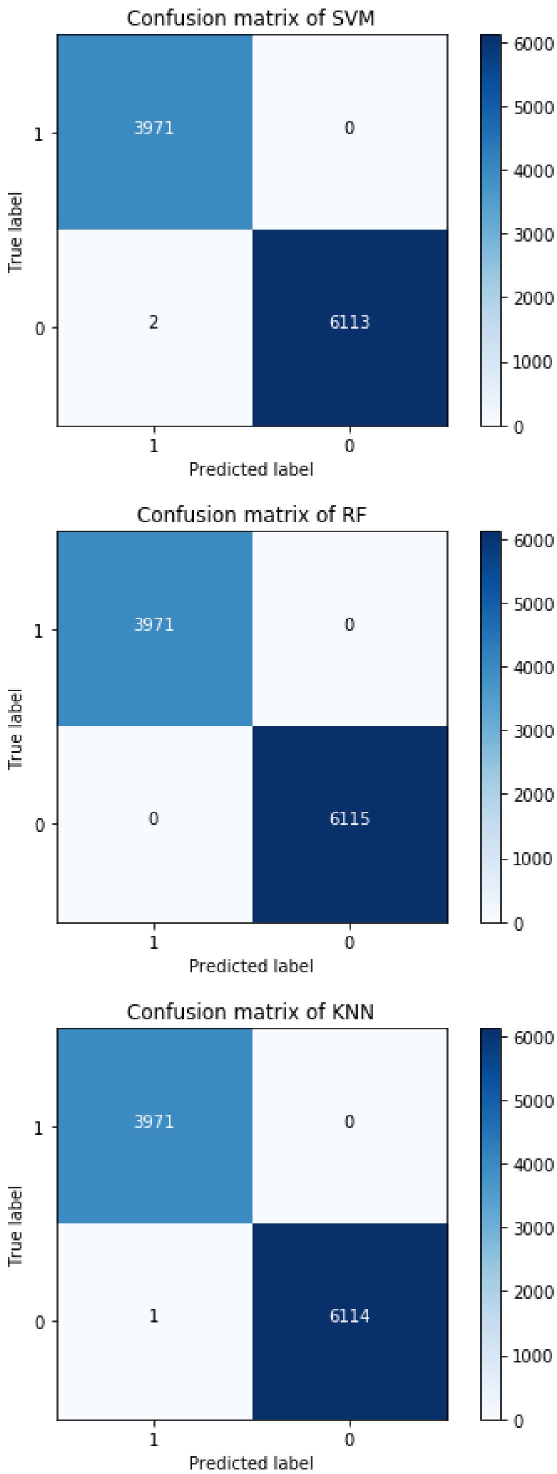


Fig. 9: Confusion matrices obtained using various ML models.

Table 1. Performance comparison of various ML models for predicting the botnet attack in IoT device data.

Model/Metric	LR model	SVM classifier	RF model	KNN classifier
Accuracy (%)	100	99.98	100	99.99
Precision (%)	100	100	100	100
Recall (%)	100	100	100	100
F1-score (%)	100	100	100	100

Table 1 presents a comprehensive performance comparison of different machine learning (ML) models employed for the prediction of botnet attacks in data obtained from IoT devices. The table showcases four ML models: LR model, SVM classifier, RF model, and KNN classifier. The metrics assessed for each model include Accuracy, Precision, Recall, and F1-score.

The LR model achieved exceptional results across all metrics. It displayed a remarkable accuracy rate of 100%, indicating that it accurately classified all instances, both botnet attacks and benign activities. This suggests that the LR model has a comprehensive understanding of the underlying patterns in the data, enabling it to make precise predictions. Additionally, the LR model demonstrated 100% precision, recall, and F1-score. This implies that the model not only made accurate positive predictions (precision) but also correctly identified all true positive instances (recall), leading to a harmonic balance between precision and recall, as represented by the F1-score. The SVM classifier performed impressively as well, with an accuracy of 99.98%. This indicates that the model almost perfectly classified instances into their respective categories. Similar to the LR model, the SVM classifier attained 100% precision, recall, and F1-score, signifying its strong capability to make accurate predictions and correctly identify botnet attacks. The RF model achieved a perfect accuracy of 100%, matching the performance of the LR model. This suggests that the RF model was able to effectively capture the complexities and variations in the data. The precision, recall, and F1-score also reached 100%, indicating consistent and reliable performance across these evaluation metrics. The KNN classifier, while slightly behind the other models in terms of accuracy with 99.99%, still demonstrated outstanding predictive capability. Like the other models, the KNN classifier achieved perfect precision, recall, and F1-score. This showcases the model's ability to make accurate and consistent predictions.

Finally, Table 1 showcases the remarkable performance of all the evaluated ML models for predicting botnet attacks in IoT device data. Each model exhibited near-perfect accuracy, precision, recall, and F1-score, implying their proficiency in accurately identifying both botnet attacks and benign activities. These findings suggest that the models are well-suited for detecting and preventing malicious activities within IoT networks, enhancing the overall security and reliability of IoT devices and systems.

5. CONCLUSION

The detection and classification of BotNet attacks using machine learning (ML) have shown promising results in enhancing cybersecurity measures. By leveraging various ML algorithms such as Random Forest, Support Vector Machines, and Neural Networks, researchers have been able to identify patterns and anomalies indicative of BotNet activities. These models can effectively distinguish between normal network traffic and malicious traffic, reducing the false positive rates and improving the overall accuracy of detection systems. Furthermore, the integration of feature engineering techniques and advanced data preprocessing methods has significantly contributed to the robustness and reliability of these ML-based detection systems. As cyber threats evolve, ML models must continuously adapt and learn from new data to maintain their efficacy in identifying sophisticated BotNet attacks.Looking ahead, the future of BotNet attack detection and classification using ML lies in the development of more adaptive and scalable models. The rapid advancement of ML techniques, such as deep learning and reinforcement learning, offers new avenues for creating more sophisticated detection systems capable of handling large-scale network data in real-time.

REFERENCES

[1] S. Dange and M. Chatterjee, "IoT botnet: The largest threat to the iot network" in Data Communication and Networks, Cham, Switzerland: Springer, pp. 137-157, 2020.

[2] J. Ceron, K. Steding-Jessen, C. Hoepers, L. Granville and C. Margi, "Improving IoT botnet investigation using an adaptive network layer", Sensors, vol. 19, no. 3, pp. 727, Feb. 2019.

[3] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, et al., "N-baiot-network-based detection of iot botnet attacks using deep autoencoders", IEEE Pervas. Comput., vol. 17, no. 3, pp. 12-22, 2018.

[4] Shah, S.A.R.; Issac, B. Performance comparison of intrusion detection systems and application of machine learning to Snort system. Futur. Gener. Comput. Syst. 2018, 80, 157–170.

[5] Soe YN, Feng Y, Santosa PI, Hartanto R, Sakurai K. Machine Learning-Based IoT-Botnet Attack Detection with Sequential Architecture. Sensors. 2020; 20(16):4372. https://doi.org/10.3390/s20164372

[6] I. Ali et al., "Systematic Literature Review on IoT-Based Botnet Attack," in IEEE Access, vol. 8, pp. 212220-212232, 2020, doi: 10.1109/ACCESS.2020.3039985.

[7] Irfan, I. M. Wildani and I. N. Yulita, "Classifying botnet attack on Internet of Things device using random forest", IOP Conf. Ser. Earth Environ. Sci., vol. 248, Apr. 2019.

[8] Shah, T., Venkatesan, S. (2019). A Method to Secure IoT Devices Against Botnet Attacks. In: Issarny, V., Palanisamy, B., Zhang, LJ. (eds) Internet of Things – ICIOT 2019. ICIOT 2019. Lecture Notes in Computer Science(), vol 11519. Springer, Cham. https://doi.org/10.1007/978-3-030-23357-0_3

[9] C. Tzagkarakis, N. Petroulakis and S. Ioannidis, "Botnet Attack Detection at the IoT Edge Based on Sparse Representation," 2019 Global IoT Summit (GIoTS), Aarhus, Denmark, 2019, pp. 1-6, doi: 10.1109/GIOTS.2019.8766388.

[10] Y. Meidan et al., "N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders," in IEEE Pervasive Computing, vol. 17, no. 3, pp. 12-22, Jul.-Sep. 2018, doi: 10.1109/MPRV.2018.03367731.

[11] S. I. Popoola, R. Ande, B. Adebisi, G. Gui, M. Hammoudeh and O. Jogunola, "Federated Deep Learning for Zero-Day Botnet Attack Detection in IoT-Edge Devices," in IEEE Internet of Things Journal, vol. 9, no. 5, pp. 3930-3944, 1 March1, 2022, doi: 10.1109/JIOT.2021.3100755.

[12] F. Hussain et al., "A Two-Fold Machine Learning Approach to Prevent and Detect IoT Botnet Attacks," in IEEE Access, vol. 9, pp. 163412-163430, 2021, doi: 10.1109/ACCESS.2021.3131014.

[13] Abu Al-Haija Q, Al-Dala'ien M. ELBA-IoT: An Ensemble Learning Model for Botnet Attack Detection in IoT Networks. Journal of Sensor and Actuator Networks. 2022; 11(1):18. https://doi.org/10.3390/jsan11010018.

[14] Alharbi A, Alosaimi W, Alyami H, Rauf HT, Damaševičius R. Botnet Attack Detection Using Local Global Best Bat Algorithm for Industrial Internet of Things. Electronics. 2021; 10(11):1341. https://doi.org/10.3390/electronics10111341

[15] Ahmed, A.A., Jabbar, W.A., Sadiq, A.S. et al. Deep learning-based classification model for botnet attack detection. J Ambient Intell Human Comput 13, 3457–3466 (2022). https://doi.org/10.1007/s12652-020-01848-9.