International Scientific Journal of Engineering and Management
Volume: 03 Issue: 06 | June – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM02082

A Machine Learning Approach for Anomaly Detection of IoT Traffic

Pranay Chouksey
PG Scholar, Gyan Ganga Institute of Technology and Sciences, Jabalpur

Dr. Siddharth Bhalerao
Associate Professor, Gyan Ganga Institute of Technology and Sciences, Jabalpur

Abstract
The implementation of IoT technology has enhanced the quality of life by optimizing numerous practical applications. The Internet of Things encompasses several devices that produce substantial volumes of data, necessitating computationally demanding processing.Identifying vulnerabilities and ensuring security are primary priorities in the field of IoT. Network vulnerability detection is a highly promising and efficient method for enhancing network security. The machine learning model is trained to proficiently analyze traffic data. This document presents a comprehensive analysis of machine learning techniques used to identify vulnerabilities in Internet of Things (IoT) applications. This article discusses the primary concerns and obstacles involved in acquiring knowledge about the technology utilized for practical issues in the Internet of Things. The suggested solution utilises authentic IoT data derived from actual IoT traffic. The paper utilises many classification algorithms such as Regression, Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), k- Nearest Neighbor (KNN), and Multi-layer Perception (MLP). Results were achieved in traffic models that involved both dual and multi-class scenarios.The findings were evaluated based on different parameters, and it was determined that the proposed strategy is more effective.Keywords: IoT network attack detection, machine learning, multiclass classification, anomaly detection, Distributed Denial-of-Service (DDoS), Denial-of-Service (DoS), IoT Security

Keywords: IoT network attack detection, machine learning, anomaly detection, Distributed Denial-of-Service (DDoS), Denial-of-Service(DoS),IoT Security

INTRODUCTION

The Internet of Things (IoT) connects hundreds of sensors and actuators to send and receive data for an application [1]. Smart lighting IoT systems can monitor agricultural water availability and use, transportation expenses, hospital patient health, and traffic movement [2]. Thus, the IoT system manages hardware, software, and connectivity to cut costs and provide intelligent services in various conditions. Managing the vulnerabilities of various devices and hardware in the Internet of Things is tough [3]. Reliability helps IoT tackle numerous problems. Although some studies limit trust to security associated with a bad exit, there is no consensus on its definition [4].Malicious external attacks, and malfunctions, vulnerabilities, and bad behavior threaten IoT systems. Both reliability and reliability [5] are similar. It requires availability, reliability, capacity, control, efficiency, data quality, hardware, security, response time agility, and system and network adaptability [6]. The cloud offers real-time edge data analysis, which can help manage IoT trust requirements [7]. Fog computing goes beyond cloud servers for data analysis. At the network edge, Fog allows local data analysis[8],reducing network connections and cloud data processing [20], and enables edge decision-making [9]. Even cloud-based IoT systems have dependability concerns due to data flow, network, memory, and power management[10]. Security attacks can target IoT devices and network topologies during transmission. IoT has information, access, application, and application layers. Every layer of the IoT requires security solutions and vulnerability prevention, which is tough to address and safeguard. Using Edinax SP-2101W on iPad and reporting device scanning, brute force, spoofing, and firmware attacks, Ling et al. [11] demonstrated the vulnerability of IoT devices to attack mitigation.The authors also show that the IoT node

ISSN: 2583-6129
DOI: 10.55041/ISJEM02082

lacks resources for advanced encryption. IoT security and privacy management is one of the hardest.

This research uses machine learning to predict botnet detection IoT dataset anomalies in binary and multiclass.

I. PROPOSED SYSTEM

The suggested work is an approach for predicting the kind of anomaly in a dataset for botnet identification using machine learning techniques.

A. ATTACK DETECTION USING SCANNING

The proposed work trains a scanning attack detection model in the first fold using the five stages. The following sections describe these actions.

1. Data Gathering: The proposed scanning attack detection approach starts with data collection. We examined some of the ways attackers search IoT networks and devices for data at this stage.
2. Preprocessing Data: This is necessary after recording scanner traffic. In pre-processing, we extracted the dataset's characteristics, which are a five-tuple of source IP, destination IP, source port, destination port, and protocol. The result is unlabeled data. The IP addresses utilized for scanning were used to classify them as scan traffic, whereas the remainder of the network traffic was normal traffic.

Step 3: Select Features: Select features to help a machine learning model distinguish between regular and scan traffic. After selecting helpful features, each dataset was partitioned into train, validation, and test sets for ML model training for scan detection. We randomly selected 60% of the data for training, 20% for validation, and 20% for testing to avoid over fitting and train the ML model. Training uses both the training and validation sets. Machine learning model training uses training set. We validate the trained model on the validation set after each epoch, depending on which the optimizer method modifies the ML model weights, to better train and optimize it. A new data set is used to evaluate the ML model after training.

B. Detecting DDOS Attacks

The authors trained a model in the second fold of the described process to detect DDoS attacks. As mentioned, the DDoS attack detection model detects DDoS attacks if an attacker breaks into the scanning attack detection model, installs malware on IoT devices, and then strikes. The suggested study built a DDoS detection model using five procedures. The following sections describe these actions.

Step 1: Collect data Like scanning attacks, we explored DDoS attack tactics. already in use that are often employed by attackers to carry out DDoS attacks.

1. Data Pre-processing: Following the DDoS traffic collection, we extracted the dataset's characteristics from the collected files. Therefore, we classified the DDoS attack's IP addresses as DDoS and the rest of the network's traffic as normal traffic based on their use in the attack.

2. Features Selection: After features were extracted and the dataset was labelled, we used an algorithm on the labelled dataset to choose the features that would be most helpful in assisting a machine learning model in differentiating between regular and DDoS traffic.

3. Training an ML model for DDoS detection: Following the selection of the pertinent features, the dataset was split into train, validation, and test sets. To prevent overfitting and efficiently train the ML model, we randomly picked 60% of the data for training, 20% for validation, and 20% for testing.

4. Testing and Affirmation: After the model has been trained, we assess its effectiveness using data from the test set. Since the trained model's test set is unknown, we evaluated the trained model's performance using four widely used performance metrics in order to check how well it performed.

Figure 1 depicts the system's general architecture that represents basic working components of the system like data set, pre-processing, dataset splitting, model training, prediction & evaluation.The data consist of some unwanted information. Data pre-processing is used to remove incomplete records, noise values and inconsistent data.

The suggested system would use two categorization types and several ML models.

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 06 | June – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM02082

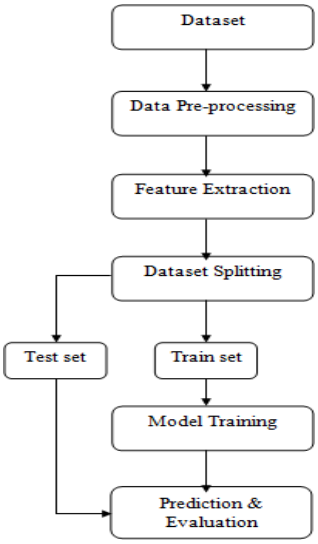

Figure 1. Proposed architecture

The binary classification technique involves establishing two classifications for the input data. It guesses which of two categories the object belongs to. Imagine receiving two emails: one from an insurance company selling to you, and one from your bank regarding your credit card payment. The email service provider will put one in the spam folder and one in the main one. This method is called binary classification since it has two classes—spam and primary.

2. Multi-class classification: Classification tasks using multiple class labels. Face categorization, plant species classification, and visual character recognition are examples. Binary classification has normal and abnormal outcomes, whereas multi-class classification does not. Instead, instances are classified into numerous popular groups. Class labels may be numerous in some problems. A model in a facial recognition system may predict that a snapshot belongs to one of thousands or tens of thousands of faces. Multi-class classification may apply to text translation models and other word prediction challenges. Multi-class classification is needed for each word in the prediction sequence, where the vocabulary size is tens of thousands to hundreds of thousands of words can be predicted as classes. The suggested multi-class classification system uses the Multi Layer Perceptron model (see picture 2). The Multi Layer Perceptron model is recommended for multi-class categorization. The traditional and simple MLP, a Feed Forward ANN, design is below. Multilayer perceptron is a feed-forward artificial neural network that outputs from inputs. MLPs are defined by a directed graph connecting input nodes in multiple levels between input and output layers. MLP trains networks with back propagation. A deep learning method is MLP.

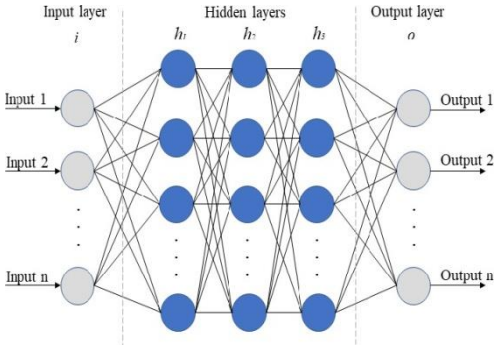

Figure 2. General MLP System

For binary classification ensemble method like random forest is proposed. Its working is shown in figure 3:

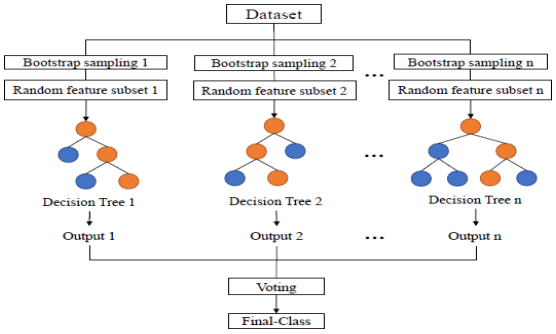

Figure 3. Random forest *method*

Supervised learning uses Random Forest. It can be used for classification and regression ML problems. Ensemble learning integrates numerous classifiers to solve challenging problems and improve model performance. As its name shows, "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." The random forest uses forecasts from each decision tree and guesses the conclusion based on the majority of predictions.

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 06 | June – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM02082

II. DATA SET & MATERIAL

S. No:	Attribute Name	S. No:	Attribute Name
1	srcip	26	res_bdy_len
2	sport	27	Sjit
3	dstip	28	Djit
4	dsport	29	Stime
5	proto	30	Ltime
6	state	31	Sintpkt
7	dur	32	Dintpkt
8	sbytes	33	tcprtt
9	dbytes	34	synack
10	sttl	35	ackdat
11	dttl	36	is_sm_ips_ports
12	sloss	37	ct_state_ttl
13	dloss	38	ct_flw_http_mthd
14	service	39	is_ftp_login
15	Sload	40	ct_ftp_cmd
16	Dload	41	ct_srv_src
17	Spkts	42	ct_srv_dst
18	Dpkts	43	ct_dst_ltm
19	swin	44	ct_src_ ltm
20	dwin	45	ct_src_dport_ltm
21	stcpb	46	ct_dst_sport_ltm
22	dtcpb	47	ct_dst_src_ltm
22	smeansz	48	attack_cat
24	dmeansz	49	Label
25	trans_depth		

Figure 3: Features in dataset

For this project, the BoT-IoT dataset from UNSW-NB15 was used. Traffic on the network was a mix of real users and botnets. There are different types of source files for the dataset, such as the original pcap files, the created argus files, and csv files. The files were split up by attack type and group to make labeling easier.

The following features are part of the dataset:

Instances can be put into two groups by a binary classifier: positive and negative. Positive: The case is found to belong to the class the classifier is trying to recognize. For example, if you tell a classifier to look for pictures of cats, it will find them and mark them as positive. It has been decided that the instance does not belong to the class we are trying to find. In this case, a classifier that is looking for pictures of cats should mark as negative any pictures of dogs but not any pictures of cats.

Accuracy, memory, and F1-Score are all based on the ideas of True Positive, True Negative, False Positive, and False Negative. You can balance the classifier predictions with different metrics, such as Accuracy, Precision, Recall, and F1-score, since different types of errors have different levels of severity in different use cases.

Accuracy: It measures how many of the guesses were right, and is often used as the main way to judge the performance of a model.

Precision: It is a metric for how many correctly predicted positive outcomes (true positives) there were overall. Its formula is as follows:

Recall/ Sensitivity: Recall is a metric for how many out of all the positive cases in the data that the classifier correctly predicted. It is also known as sensitivity at times. Its formula is as follows:

Specificity: The number of correctly predicted negative outcomes (true negatives) is a measure of specificity. Its formula is:

F-1 Score: The F1-Score is a measurement that combines recall and accuracy. Generally speaking

it is referred to as the harmonic mean of the two. Another method of determining a "average" of values is the harmonic mean, which is typically seen as more suited for ratios than the conventional arithmetic mean (such as recall and accuracy). In this instance, the F1-score calculation is as follows:

The goal is to create a single measure that equally weights the accuracy and recall ratios, needing both to increase in value in order for the F1-score value to increase.

Some benefits of F1-score include: A lower total score will be the result of very low accuracy or recall. So it contributes to balancing the two metrics. F1-score can assist in balancing the metric between positive and negative samples if you chose your positive class as the one with the fewest examples.

For binary Classification:

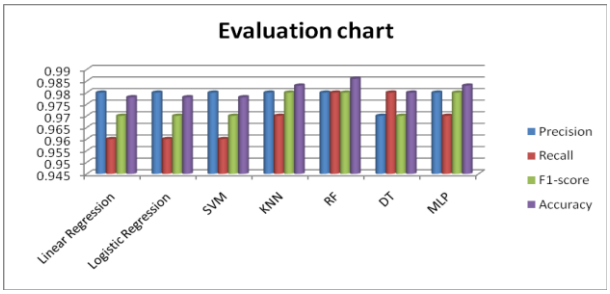

Figure 4: Result of binary classification

It is clear that Random Forest Classifieroutperforms the other classifiers in terms of MAE, MSE, RMSE and R2 Score.

For Multi-class Classification:

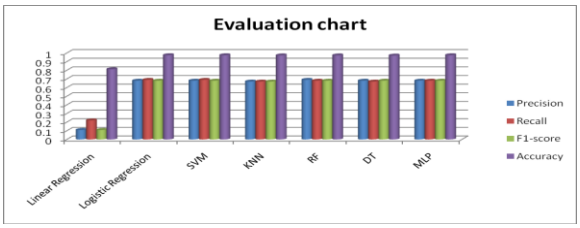

Figure 5: Result of multiclass classification

ISSN: 2583-6129
DOI: 10.55041/ISJEM02082

It is clear that Random Forest Classifier performs better as compared to other classifier in terms of parameters.

III. RESULTS AND CONCLUSION

The suggested MLP method works well for classifying things into more than two groups, and random forest works well for classifying things into two groups. Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks are the most common types of cyberattacks against IoT networks and devices, and they are happening more often and for longer periods of time. Because it works so well, the random forest model is important for any kind of classification. However, these models don't work well with small datasets because they were made to find patterns in big datasets. In this paper, we suggested a way to divide network data into two groups so that random forest models can be used to their fullest. Next, we looked at a cutting-edge RF model, Analyze forest of random cluster tree datasets and how well they work to find recent DoS and DDoS attacks. The suggested way is better at finding DoS and DDoS attacks. The multilayer perceptron model works better for classifying things into more than one group.

REFRENCES

[1]. L. Atzori, A. Lera, and G. Morabito, ‘‘The Internet of Things: A survey,’’ Comput. Netw. vol.54, no. 15, pp. 2787–2805, Oct. 2010.

[2]. E. Park, A. Del Pobil, and S. Kwon, ‘‘therole of Internet of Things (IoT) in smart cities: Technology roadmap-oriented approaches,’’ Sustainability, vol. 10, no. 5, p. 1388, May 2018. [3]. P. P. Ray, ‘‘A survey on Internet of Thingsarchitectures,’’ J. King Saud Univ., Comput. Inf. Sci., vol. 30, no. 3, pp. 291–319, 2018.

[4]. N. S. Labib, M. R. Brust, G. Danoy, and P. Bouvry, ‘‘Trustworthiness in IoT—A standards gap analysis on security, data protection and privacy,’’ in Proc. IEEE Conf. Standards Commun. Netw. (CSCN), Granada, Spain, Oct. 2019, pp. 1–7.

[5]. A. Avizienis, J.-C. Laprie, B. Randell, and B. Landwehr, ‘‘Basic concepts and taxonomy of dependable and secure computing,’’ IEEE Trans. Depend. Sec. Comput., vol. 1, no. 1, pp. 11–33, Jan. /Mar. 2004.

[6]. J.-H. Cho, S. Xu, P. M. Hurley, M. Mackay, T. Benjamin, and M. Beaumont, ‘‘STRAM: Measuring the trustworthiness of computer based systems,’’ ACM Comput. Surv., vol. 51, no. 6, pp. 1–47, Feb. 2019.

[7]. R. Mahmud, R. Kotagiri, and R. Buyya, ‘‘Fog computing: A taxonomy, survey and future directions,’’ in Internet of Everything (Internet of Things: Technology, Communications and Computing), B. Di Martino, K. C. Li, L. Yang, and A. Esposito, Eds. Singapore: Springer, 2018.

[8]. M. Aazam, S. Zeadally, and K. A. Harras, ‘‘Fog computing architecture, evaluation, and future research directions,’’ IEEE Commun. Mag., vol. 56, no. 5, pp. 46–52, May 2018.

[9]. C. Mouradian, D. Naboulsi, S. Yangui, R. H.Glitho, M. J. Morrow, and P. A. Polakos, ‘‘A comprehensive survey on fog computing: State-of- theart and research challenges,’’ IEEE Commun. Surveys Tuts. vol. 20, no. 1, pp. 416–464, 1st Quart.2018.

[10]. W. Masri, I. A. Ridhawi, N. Mostafa, and P. Pourghomi, ‘‘Minimizing delay in IoT systems through collaborative fog-to-fog (F2F) communication,’’ in Proc. 9th Int. Conf. Ubiquitous Future Netw. (ICUFN), Milan, Italy, Jul. 2017, pp. 1005–1010.

[11]. Z. Ling, J. Luo, Y. Xu, C. Gao, K. Wu, andX. Fu, ‘‘Security vulnerabilities of Internet of Things: A case study of the smart plug system,’’ IEEE Internet Things J., vol. 4, no. 6, pp. 1899– 1909, Dec. 2017.

[12]. R. Harada *et al*., "Quick Suppression of DDoS Attacks by Frame Priority Control in IoT Backhaul With Construction of Mirai-Based Attacks," in *IEEE Access*, vol. 10, pp. 22392-22399, 2022, doi: 10.1109/ACCESS.2022.3153067