Comparison of Machine Learning Models for IoT Malware Classification

Piragash Maran[1(✉)], Timothy Tzen Vun Yap[1], Ji Jian Chin[1], Hu Ng[1], Vik Tor Goh[2], and Thiam Yong Kuek[3]

[1] Faculty of Computing and Informatics, Multimedia University, 63100 Cyberjaya, Malaysia
1181101448@student.mmu.edu.my
[2] Faculty of Engineering, Multimedia University, 63100 Cyberjaya, Malaysia
[3] Faculty of Business and Finance, Universiti Tunku Abdul Rahman, 31900 Kampar, Malaysia

Abstract. The Internet of Things (IoT) is a system where devices and sensors are interconnected to improve accuracy, efficiency, precision and consistency. It is being developed rapidly as more people are aware of this system. From farmers, all the way to the automotive engineers are all benefiting from the usage of IoT (Internet of Things). IoT transfers data in a very large amount without the help of a human, making the system very efficient and time saving. Since there is no assistance from humans, IoT can generate more data than ever. This paper focuses more on the security part of IoT devices or sensors. Machine learning (ML) algorithms are used to investigate and detect any malware in a dataset generated from an IoT device. The paper concludes which algorithm is more successful in detecting malware from the dataset and compares the result or the accuracy. The algorithms that this paper used are Random Forest (RF), Naive Bayes (NB), Artificial Neural Network (ANN), Decision Tree (DT) and K-Nearest Neighbours (KNN). The best results were achieved by the Random Forest algorithm with an accuracy score of 96%.

Keywords: Machine learning · Cybersecurity · Internet of Things · IoT-23 · Malware classification · Malware analysis

1 Introduction

The name, Internet of Things, is only 16 years old, but the whole development of these systems started since the 70's under the name of "embedded internet" or "pervasive computing". Only in 1999, the name IoT was used to describe the connection between the sensors, actuators, and devices through wired or wireless communication. Lately, even the governments have included the development of IoT in their yearly major national plan or the budget plan. For example, an announcement by the Chinese government state that they will include the Internet of Things as a main priority in their Five-Year-Plan. IoT made the communication between people and things possible through devices such as smart mobiles, kitchen appliances, smart refrigerators, smartwatches, cars, smart fire alarms, smart door locks, smart bicycles, medical sensors, thermostats, fitness trackers, baby monitors, smart security system, etc.

S.-C. Haw and K. Sonai Muthu (Eds.): CITIC 2022, 10, pp. 15–28, 2022.
https://doi.org/10.2991/978-94-6463-094-7_3

There also has been a rapid advancement in the usage of machine learning techniques so that the system can operate without the needs of absolute programming. There are many kinds of machine learning algorithms available right now, such as RF, NB, ANN, DT and KNN, Support Vector Machine (SVM), etc. There are also many approaches towards machine learning algorithms such as supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. The most common ones are supervised and unsupervised learning. This paper focuses on a supervised approach.

For now, the most challenging part is tackling the security issues surrounding the Internet of Things. There is now much research going on to improve the security measures of IoT to prevent attacks such as botnet attacks, eavesdropping, Mirai attack, hacking IoT devices, data breaches, etc. Without proper security, the attackers could easily have access to the important data of the users as IoT devices have the capability to siphon large amounts of data from the users.

This paper focuses on the classification and analysis of IoT malware using machine learning algorithms. This paper explores the IoT-23 dataset [1], a dataset that has data captured from IoT devices, both malicious and benign.

2 Literature Review

For now, researchers are still discovering ways to implement the machine learning techniques in the classification process. There are also new methods and frameworks being developed and formed by researchers using machine learning techniques. Nicolas-Alin Stoian, 2020, who has a similar framework as this project, focuses on the security aspect of IoT networks by investigating the usability of machine learning algorithms in the detection of anomalies found within, tested various machine learning algorithms on the IoT-23 dataset such as Random Forest (RF), Naive Bayes (NB), Multi-Layer Perceptron (MLP), a variant of the Artificial Neural Network class of algorithms, Support Vector Machine (SVM) and AdaBoost (ADA)] [1]. The researcher then concluded that the Random Forest algorithm is the best algorithm when it comes to classification of the IoT-23 dataset. However, Nicolas-Alin Stoian faced difficulties when handling the large dataset and fitting it into the algorithm.

Zeadally and Tsikerdekis, 2019, stated that even though IoT devices are used everywhere due its large data handling, the security part is being ignored and demonstrated how machine learning techniques can improve the security much more [2]. Based on them, there are two types of methods to implement machine learning techniques that are network-based and host-based and this paper focuses on the network-based method. The paper also states that machine learning also has vulnerabilities that can be spotted by the attackers and cannot be fully depended on it. Zeadally and Tsikerdekis also stated that algorithm portability is one of the limitations due to the lack of standardized tools and libraries.

Imtiaz Ullah and Qusay H. Mahmoud, 2021, proposed a new framework which analyzes network traffic flow data to classify IoT devices [3]. The researchers developed the IoT23 pcap files from the dataset. At first, the network traffic is collected by a network management tool and the device behaviour is identified. This paper highlights the usage of sensor traffic analysis to classify IoT devices across the network. With this, more

malicious sensors can be detected. Imtiaz Ullah and Qusay H. Mahmoud stated that with sensor profiles, various other security measures could be implemented on different types of IoT devices. However, Imtiaz Ullah and Qusay H. Mahmoud faced limitations of accessing dataset for many devices and therefore not having enough devices to test this framework.

Vibekananda Dutta, Michal Choraś, Rafal Kozik, and Marek Pawlicki, 2020, proposed a hybrid model [4]. The hybrid model achieved higher scores of accuracy, precision, recall, f1-score, and false-positive rate compared to Random Forest (RF) and Deep Neural Network (DNN). The researchers stated that the proposed hybrid model improves the classification and identifying the behaviour of attacks. For future, the researchers plan to use more data samples and to study the results to improve the proposed hybrid model.

Anthi, Williams and Burnap, 2018, focused on a novel method called pulse [5]. Pulse is a network based real time malware detection system. It is a signature and also anomaly-based detection system. From the results of the research, Naive Bayes algorithm achieved the best performance results with the precision score of 81% to 97.7% according to various types of attack.

Booij et al. explains the role of heterogeneity and the need for standardization of features [6]. Datasets such as IoT Network Intrusion Dataset, IoT-23 (which is used in this paper) and N-BaIt were used in this research because these contain the nature of the current IoT networks with different protocols, standards and technologies. The researchers focus on showing the effectiveness of heterogeneity in improving the learning rate of machine learning detection algorithms. Three types of data were used in this experiment which were sensor data, raw data and log data. Booij and Tim M stated that the random forest algorithm achieved the best score of 99.688% for ToNIoT and 99.986% for IoT-23. However, the paper stated that the weakness of the sampled data is that training and testing sets are taken from the same set. The paper concluded that the ToNIoT dataset is one of the best dataset that is available now. Booij and Tim M showed the functionality of a cross-training method of combining different IoT network intrusion dataset. The future work of the paper is that the researchers hope that industry and academia collaborate on defining features and attack to create a better IoT network intrusion detection datasets.

Sudheera,Kalupahana Liyanage Kushan et al. proposed a security framework called Adept [7]. This framework overcomes the challenges of spatial dispersion and temporal dispersion. As stated by the researchers, the proposed model undergoes three phases which are processing IoT traffic locally, employ a data mining technique and usage of alert-level and pattern-level information and classifies the malicious activity.

K-NN, RF and SVM are the algorithms used in identifying the attack. The paper stated that two sets of features are used that are alert based features and pattern based features. The dataset that is used in this paper is IoT-23. The dataset is used as an external dataset to boost the classification performance. Two mirai samples are used, CTU-IoT-Malware-Capture-34–1 and CTU-IoT-Malware-Capture- The researchers then concluded that the Adept approach is a better approach compared to other approaches as the F1-score and accuracy of Adept is very high. Rafał Kozik, Marek Pawlicki and Michał Choraś, 2021, proposed a new method which uses a transformer-based classification scheme for a time-window embedding solution [8]. The researchers stated that

the current and usual classification systems are over dependent on network features. The transformer architecture contains both encoding and decoding parts. This paper focuses on the encoding part. The paper stated that the data flows into the transformer using the proposed method which is the time windows embedding technique. This proposed method calculates the statistical properties of flows with IP addresses and allows the researchers to capture some short term malicious behaviour as it is one of the advantages of the method.

Based on the results, the proposed method, which is classification using a transformer, achieved the best F1-score compared to other methods when tested in all scenarios. This shows classification using transformers is very effective in detecting and classifying malicious behaviour. The researchers stated that using the IoT-23 dataset made handling various privacy issues easier. As usual, the dataset was split into two parts which is the testing and training part. The researchers then concluded that the proposed method using transformer classification is the best compared to other methods such as RF-500, RF-100, REPT, RF-10 and AdaRept. The experiment showed that the proposed method works the best in transformer-based classification after running on Aposemat IoT-23 dataset.

Sánchez and Pedro Miguel Sánchez, 2021, presented various methods where fingerprinting is used [9]. The main focus of the paper is to show that fingerprints can detect various devices and different models.

The paper stated that machine learning (ML) and DL are gaining importance and popularity because of their adaptability and effectiveness when large amounts of data is used. The paper then concluded that ML and DL methods are very efficient in all scenarios. The future work expected by the researchers is that they plan to create a dataset so that it can be fitted into more scenarios and it would be publicly attainable. The paper showed the implementation of behaviour based methods for IoT device classification and its effectiveness.

Sahu, Amiya Kumar et al. proposed a method which uses convolution neural network(CNN) and short-term memory models [10]. It is constructed as a hybrid deep learning. The researchers performed dynamic analysis to the flows from the samples that were recently published. The attacks were split into 8 different categories that is Command and Control (C&C), Distributed Denial of Server (DDoS), FileDownload, HeartBeat, PartOfAHorizontalPortScan, Mirai, Torii and Okiru. The researchers stated that deep learning(DL) techniques has advantages over machine learning(ML) ones. The dataset that was used in the experiment is Aposemat IoT-23.

Firstly, the CNN model studies the features from the flows of the sample and transfers them into the deep learning's long short term memory(LSTM) model. During the preprocessing phase, the labelled files are transformed into.csv file format. The dataset was also split into training and testing sets. After that, the model was trained and tested.

The researchers then stated that these kinds of models are much cheaper and more efficient compared to individual cryptographic systems and security. The proposed model achieved an accuracy score of 96%. The model also overcomes the under fitting or overfitting problems.

Al-Zewairi, Malek, Sufyan Almajali, and Moussa Ayyash, 2020, focused on shallow and deep ANN classifiers to classify security attacks [11]. The researchers are experimenting with machine learning techniques to detect unknown attacks. Based on the researchers, unknown attacks are one of the most common attacks in the past few years. Datasets such as UNSW-NB15 and the Bot-IoT are used by the authors in this research. UNSW-NB15 contains two kinds of labels which are category and label whereas Bot-IoT has attack, category and subcategory. It would have been easier for the researchers to work on the dataset since they are made from the same laboratory. The dataset underwent a pre-processing phase where the dataset was cleaned and converted to.csv file format. Then, the feature selection phase was carried out with the datasets. The features were generated, selected, converted and normalized. The paper states that the ANN model is better than machine learning(ML) models such as SVM, NB and RF. In this paper, the researchers studied both the shallow and deep ANN models and the paper stated deep ANN models always surpass shallow ANN models. Before the experiment begins, a testing, validation and training set are separated from the dataset. The training and validation set were split into 0.9 and 0.1. Also, the sets are split into Type-A which contains info of unknown attack and Type-B which contains info of known attack types.

The experiment was carried out by the researchers. The researchers then concluded that the ANN model was incapable of detecting unknown attack and reached 50% of error rate and then stated that a more innovative approach is needed to detect more unknown attacks. The future work is examining hybrid supervised and unsupervised machine learning(ML) models.

Anagnostopoulos, Marios, et al. proposed a system called Ghost [12]. Ghost is a smart-home security system. It classified network and traffic flows through Wi-Fi, Bluetooth, Ethernet, ZigBee, LPWANs, Cellular (3G/4G/5G) or RFID. The researchers stated that the Ghost system is able to capture in and out of the network flow of the smart-home system. The paper stated that the Ghost system consists of 5 layers which are:

(1) Gateway layer
(2) Data Interception and Inspection layer
(3) Contextual profiling layer
(4) Risk assessment layer
(5) Control and monitoring layer.

The NDFA which is, Network and Data Flow Analysis, is responsible for monitoring the traffic flow and takes the right amount of data. Another function of NDFA is to examine the pcap files. It is also responsible for the main data flow of the Ghost. The researchers have used the dpkt Python library as a tool for the IP packet examining phase. Then, the system undergoes IP Flows Analysis, Non-IP Protocols Work-Flow, Packet Analysis and Batches Analysis. The researchers have concluded that the Ghost system is a very smart-home security scheme and system. It is able to detect multiple protocols and classify important network flow information from the traffic flow. The dataset used in the research was GHOST-IoT-data-set.

Vibekananda Dutta, Michał Choraś, Marek Pawlicki and Rafał Kozik, 2020, focused on a deep learning ensemble for classification and analysis [13]. Datasets such as IoT-23,

LITNET-2020, and NetML-2020 was used in this research. This paper focuses on deep learning techniques for improving cybersecurity. The steps in this model as follows:

(1) Dataset Selection
(2) Data Pre-processing
(3) Data Output
(4) Data Splitting
(5) Classification and Analysis.

In the data pre-processing phase, there will be feature selection where information which serves no purpose will be removed and eliminated from the datasets. The proposed DNN is a four-layer neural network. The results showed that the proposed method has achieved the highest accuracy score of 0.997 compared to Random Forest, DNN and LSTM. The researchers then concluded that when DNN and LSTM are combined, the model works very efficiently and has higher accuracy. The future work of the researchers is to extend the implementation strategy so that the model can be tested using more dataset and hence, more results can be obtained to be studied.

Vibekananda Dutta, Michał Choraś, Marek Pawlicki and Rafał Kozik, (2020), focused on using DAE, Multi-Layer Perceptron (MLP/DNN) and LSTM [14]. The steps involved in this approach are:

(1) Data-set selection.
(2) Feature engineering.
(3) Classification process.
(4) Results.

The results showed that the proposed mLSTM method has achieved the highest average accuracy score with 99**.**98% compared to DNN with 99**.**68% and LSTM with 99**.**84%. The researchers then concluded that when DAE and mLSTM collaborated, it was very efficient when it comes to handling big datasets such as IoT-23. The future work is to work more on deep learning algorithms for classification and analysis.

Kira Bobrovnikova, Sergii Lysenko, Piotr Gaj, Valeriy Martynyuk and Dmytro Denysiuk, 2020, proposed DNS traffic analysis for attack identification [15]. Steps of the proposed model as follows:

(1) Collecting the DNS traffic.
(2) List checking.
(3) Features extraction.
(4) Features analysis.
(5) Stopping infected IoT devices.

Based on the results, SVM produced better results, from 96.06% to 98.01% and false positives of 0.015% to 0.31%, compared to Artificial Immune system, from 95.19% and false positives of 1.9%, and Semi-supervised fuzzy c-means clustering, from 93.8% to 95.9% and false positives from 0.01% to 0.48%. The researchers then concluded that DNS traffic analysis is very efficient in classifying and analysing unknown attacks.

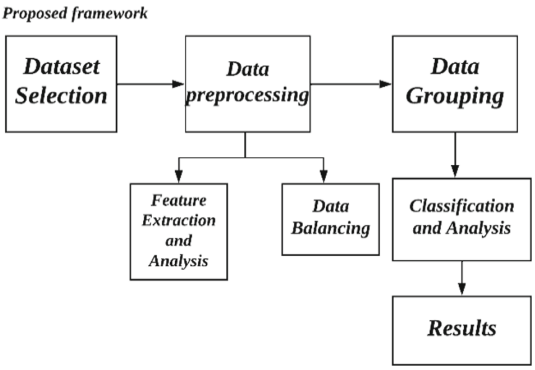

Fig. 1. Overview of methodology process.

Future work of the researchers is to continue to develop and train machine learning(ML) algorithms to classify IoT attacks.

After doing a deep research on these similar papers relating to machine learning techniques, an innovative idea is built to carry out the experiment in this paper. Most of the steps from this paper are inspired by the framework of the similar research papers that are mentioned above. Moreover, papers more than 5-year-old were not able to be studied due to the recent developed machine learning techniques.

3 Research Methodology

This part of the paper explains the method that is implemented in this project. Methods such as data pre-processing, algorithm implementation and results recording are all explained in this part of the paper. The first step of this project is data pre-processing. This step includes cleaning, transforming, combining and data splitting. This method was necessary because the dataset had to be fitted into the respective algorithm. Figure 1 shows the methodology process.

3.1 General Overview

This section explains the proposed method of IoT malware classification in general.

The proposed framework consists of (1) dataset selection, (2) data preprocessing which consists of feature extraction and analysis, data balancing and then (3) data grouping, (4) classification and analysis using the algorithms which are Random Forest, Naive Bayes, Artificial Neural Network, Decision Tree and K-Nearest Neighbors(KNN), and finally the (5) Results. Tools such as python was used in the process.

3.2 Datase

The dataset selected and used in this paper is Aposemat IoT-23 (will be called IoT-23 from now onwards), by Avast AIC laboratory. Czech Republic is the country where the IoT-23 dataset was created and developed. The data was collected from the year 2018

to 2019. The dataset contains 20 malware captures and labels and rest are benign. Other than that, it carries 21 feature attributes. The dataset also carries pcap files, labelled (conn.log. Labelled) files. In this research, only labelled files were used as it was easier to work on with. The pcap files were not important in this project as they were difficult to handle and were mostly ignored throughout the project. The dataset was downloaded from the website called Stratosphereips.org. Two methods were available to download the dataset which was by downloading the whole folder in a compressed zip folder format or download the files separately, such as conn.log. Labelled and pcap. A total of 325,307,990 captures can be found in this dataset making the size of the dataset very large. The types of attack from the capture are shown in Table 1.

3.3 Data Pre-processing

In this stage, the paper explains how the data was preprocessed. Data pre-processing has two sub-phases that are feature extraction and analysis and data balancing. After the data was downloaded, the conn.log. Labelled files were then converted into.csv file format. The reason for this step is because the files need to be read by python and python is only able to read.csv files since the features are separated by commas. However, due to technical limitations, some of the capture or files have to be omitted due to very large space. Only certain files were able to be converted into.csv file format. The conn.log. Labelled files or captures are as follows:

(1) CTU-IoT-Malware-Capture-3-1
(2) CTU-IoT-Malware-Capture-8-1
(3) CTU-IoT-Malware-Capture-34-1
(4) CTU-IoT-Malware-Capture-42-1
(5) CTU-IoT-Malware-Capture-1-1
(6) CTU-IoT-Malware-Capture-20-1
(7) CTU-IoT-Malware-Capture-21-1
(8) CTU-IoT-Malware-Capture-44-1
(9) CTU-IoT-Malware-Capture-60-1.

After converting conn.log. labelled files into .csv files, the next steps are feature extraction and analysis.

3.3.1 Feature Extraction and Analysis

In this step or phase, the dataset undergoes feature selection. According to Nicolas-Alin Stoian, 2020, columns such as ‘ts’, ‘uid’, ‘id.orig_h’, ‘local_orig’, ‘local_resp’, ‘missed_bytes’, ‘tunnel_parents’ are completely unnecessary for IoT malware classification and is advised to be removed. Other reasons are missing so much data and unrelated things were also stated by Stoian. After further examination, some columns were also decided to be removed or omitted since they serve no purpose in this research.

Also, the empty values in the columns were changed to zero so that the results will be more accurate. After this step is completed, data balancing step is followed by.

Table 1. Types of attack.

	Type
1	Attack
2	Benign
3	C&C
4	C&C-FileDownload
5	C&C-Mirai
6	C&C-Torii
7	DDoS
8	C&C-HeartBeat
9	C&C-HeartBeat-Attack
10	C&C-HeartBeat-FileDownload
11	C&C-Part Of A Horizontal Port Scan
12	Okiru
13	Okiru Attack
14	Part Of A Horizontal Port Scan
15	Part Of A Horizontal Port Scan Attack

3.3.2 Data Balancing

Before any algorithms to be implemented on the dataset, the dataset had to balanced out so that the algorithms can run smoothly and the results can be equally justified. So, a library from python called RandomOverSampler was used. Initially, there was a choice of two samplers to use, SMOTE or RandomOverSampler. After multiple trials and runs, RandomOverSampler was more suitable for the IoT-23 dataset. The sampler increases overfitting and computational effort.

3.4 Data Grouping

Due to technical limitations, all the samples in the dataset cannot be used. It is almost impossible to use all of them due to very large space. So, a few samples were only used. The samples were grouped together and they are:

- CTU-IoT-Malware-Capture-3-1
- CTU-IoT-Malware-Capture-8-1
- CTU-IoT-Malware-Capture-34-1
- CTU-IoT-Malware-Capture-42-1
- CTU-IoT-Malware-Capture-1-1
- CTU-IoT-Malware-Capture-20-1
- CTU-IoT-Malware-Capture-21-1
- CTU-IoT-Malware-Capture-44-1
- CTU-IoT-Malware-Capture-60-1.

Table 2. Label encoding.

Detailed name	Encoded label
'C&C'	2
'C&C-FileDownload'	2
'C&C-HeartBeat'	3
'C&C-HeartBeat-Attack'	3
'C&C-HeartBeat-FileDownload'	3
'C&C-Mirai'	4
'C&C-Torii'	5
'DDoS'	6
'FileDownload'	7
'Okiru'	8
'Okiru-Attack'	8
'PartOfAHorizontalPortScan'	9
'PartOfAHorizontalPortScan-Attack'	9
'C&C-PartOfAHorizontalPortScan'	9
'Attack'	10

After that, the data was split into a training and testing set. The training set was used as a sample for the project while the training set was used for qualified performance. It was split into 70% training and 30% testing sets. Before the classification process begun, the data was encoded with each attack having its own number as a label. This process takes place so that the algorithm and python can read the values and to give a specific label to the detailed malicious type. Table 2 shows the encoded labels.

3.5 Classification and Analysis

In this part, the paper explains the part where the dataset will be fitted into the algorithm to get the classification and analysis results. The algorithms that were used in this research are Random Forest, Naive Bayes, Artificial Neural Network, Decision Tree and K-Nearest Neighbors (KNN).

3.5.1 Decision Tree

This algorithm is a very basic one compared to Support Vector Machine, Naive Bayes, etc. When it comes to large datasets such as IoT-23, decision trees have the ability to work faster and smoother than other algorithms. Due to its functional limitation, other advanced algorithms are preferably used in machine learning classification research. However, decision trees can still depend on making good classifying results.

3.5.2 K-Nearest Neighbors (KNN)

KNN is a simpler and uncomplicated machine learning algorithm. It also has the ability to do classification in a faster way. It depends on the discrete values for the output. Of course, for advanced research on machine learning classification, KNN is not preferred compared to random forest, naive bayes, SVM, etc. It is very computationally light, as identical as a decision tree. Very simple and basic machine learning algorithm.

3.5.3 Random Forest

This algorithm uses many trees. It uses multiple and many features to perform analysis. It is considered one the best algorithm to be used for machine learning analysis. Due to its simplicity and flexibility, it is used widely and frequently when comes to analysis and classification.

3.5.4 Naive Bayes

Naïve Bayes is commonly used for very large dataset. This algorithm can perform analysis very fast and easily. Its assumptions of an independent value of the features are high. However, they are limitations for this algorithm are that is assumes that all the features are independent. Also, it will assume that the values that are not in the training data are none and therefore considers it as zero probability. So, this algorithm might not be suitable for this IoT-23 dataset.

3.5.5 Artificial Neural Network (ANN)

This is a deep learning base algorithm. The structure of this algorithm is inspired by a human brain. It is made up of layers which are input layer, output layer and hidden layers. It mainly solves complex and large analysis and classifications. In this research, three layers was used to run the classification process.

3.5.6 Performance Metrics

The classification process gives an output for precision, accuracy, recall score, support score and F-1 score.

The formula for calculating precision is by calculating the percentage of accurate positives:

$$\frac{TP}{TP + FP} \tag{1}$$

The formula for recall score is by computing the percentage of true positives properly identified:

$$\frac{TP}{TP + FN} \tag{2}$$

The formula for accuracy is calculated by dividing the number of correct predictions by the total number of forecasts and knows as ratio of classified classes to total classes:

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Table 3. Accuracy.

Algorithms	Accuracy score
Decision Tree	0.76
K-Nearest Neighbors (KNN)	0.65
Naive Bayes	0.63
Random Forest	0.96
Artificial Neural Network	0.46

F-1 score is basically the weighted average of the score of precision and the score of recall with having both false positives and false negatives. It calculates the model's capability to classify classes. The formula as follows:

$$2 * \frac{precision * recall}{precision + recall} \tag{4}$$

The difference between macro and micro averages is that macro takes all classes equally while micro takes the frequency of each class.

$$F1 - micro = \sum_{i=1}^{n} \left(\mathrm{F1(i)} * \frac{support\ score\ i}{set\ size} \right) \tag{5}$$

$$F1 - macro = \sum_{i=1}^{n} \left(\mathrm{F1(i)} * \frac{1}{n} \right) \tag{6}$$

4 Results and Discussion

The algorithms used in this research were run in Python version 3.9.7 64-bit. Pandas library was used to combine, drop and clean the dataset while sklearn (scikit-learn) library was used to run the algorithms.

After the data was grouped and balanced with RandomOverSampler, the accuracy score was obtained and recorded in Table 3.

As the data was equally and randomly grouped, it was fitted into the 5 algorithms. Based on the results, Random Forest appeared to be the better algorithm when compared to Decision Tree, K-Nearest Neighbors, Naive Bayes and Artificial Neural Network. Random forest has the highest accuracy score of 96% and the lowest score was obtained by Artificial Neural Network, 46%. The reason why Naïve Bayes achieve poor results despite their capability of handling large dataset is because the data is not independent. So, clearly we can say that the assumptions made by the algorithm is definitely wrong. It is also because the columns for this dataset are not independent. Surprisingly, not even Random Forest could detect Mirai attacks due to the number of occurrences. Also, the Random Forest algorithm has the capability of overcoming overfitting problem as mentioned earlier.

5 Conclusion

In conclusion, of the five algorithms, Random Forest performs best, and Naïve Bayes tend to work better with data that is independent. In addition, in this particular case, balancing the data also affects the results and the original imbalanced dataset may cause bias in the classification. For future work, the efficacy of deep neural nets and suitable architectures will be investigated.

Authors' Contributions. Piragash Maran, Roles: Data Curation, Formal Analysis, Investigation, Resources, Writing–Original Draft Preparation.

Timothy Tzen Vun Yap, Roles: Corresponding author, Conceptualization, Project Administration, Supervision, Validation, Visualization, Writing–Review & Editing.

Ji Jian Chin, Roles: Conceptualization, Project Administration, Supervision, Validation, Writing–Review & Editing.

Hu Ng, Roles: Investigation, Validation, Writing–Review & Editing.

Vik Tor Goh, Roles: Investigation, Validation, Writing–Review & Editing.

Thiam Yong Kuek, Roles: Investigation, Writing–Review & Editing.

References

1. Stoian, N.A. (2020) Machine Learning for anomaly detection in IoT networks: Malware analysis on the IoT-23 data set.
2. Zeadally, S, Tsikerdekis, M. Securing Internet ofThings (IoT) with machine learning. Int J Commun Syst. 2020; 33:e4169. https://doi.org/10.1002/dac.4169
3. I. Ullah and Q. H. Mahmoud, "Design and Development of a Deep Learning-Based Model for Anomaly Detection in IoT Networks," in IEEE Access, vol. 9, pp. 103906-103926, 2021, doi: https://doi.org/10.1109/ACCESS.2021.3094024.
4. Dutta, V., Choraś, M., Kozik, R., Pawlicki, M. (2021). Hybrid Model for Improving the Classification Effectiveness of Network Intrusion Detection. In: Herrero, Á., Cambra, C., Urda, D., Sedano, J., Quintián, H., Corchado, E. (eds) 13th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2020). CISIS 2019. Advances in Intelligent Systems and Computing, vol 1267. Springer, Cham. https://doi.org/10.1007/978-3-030-57805-3_38
5. Anthi, E., Williams, L., and Burnap, P. Pulse: An adaptive intrusion detection for the internet of things. IET Conference Publications (2018).
6. T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa and F. T. H. d. Hartog, "ToN_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets," in IEEE Internet of Things Journal, vol. 9, no. 1, pp. 485–496, 1 Jan.1, 2022, doi: https://doi.org/10.1109/JIOT.2021.3085194.
7. Sudheera, Kalupahana Liyanage Kushan, et al. "ADEPT: Detection and Identification of Correlated Attack Stages in IoT Networks." IEEE Internet of Things Journal 8.8 (2021): 6591–6607.
8. Kozik, Rafał, Marek Pawlicki, and Michał Choraś. "A new method of hybrid time window embedding with transformer-based traffic data classification in IoT-networked environment." Pattern Analysis and Applications (2021): 1-9.
9. Sánchez, Pedro Miguel Sánchez, et al. "A Survey on Device Behavior Fingerprinting: Data Sources, Techniques, Application Scenarios, and Datasets." IEEE Communications Surveys & Tutorials (2021).

10. Sahu, Amiya Kumar, et al. “Internet of Things attack detection using hybrid Deep Learning Model.“ Computer Communications (2021)
11. Al-Zewairi, Malek, Sufyan Almajali, and Moussa Ayyash. “Unknown Security Attack Detection Using Shallow and Deep ANN Classifiers.“ Electronics 9.12 (2020): 2006.
12. Anagnostopoulos, Marios, et al. “Tracing Your Smart-Home Devices Conversations: A Real World IoT Traffic Data-Set.“ Sensors 20.22 (2020): 6600.
13. Dutta V, Choraś M, Pawlicki M, Kozik R. A Deep Learning Ensemble for Network Anomaly and Cyber-Attack Detection. Sensors. 2020; 20(16):4583. https://doi.org/10.3390/s20164583
14. Dutta, V., Choras, M., Pawlicki, M., & Kozik, R. (2020). Detection of Cyberattacks Traces in IoT Data. J. Univers. Comput. Sci., 26(11), 1422-1434.
15. Bobrovnikova, K., Lysenko, S., Gaj, P., Martynyuk, V., & Denysiuk, D. (2020). Technique for IoT Cyberattacks Detection Based on DNS Traffic Analysis. In IntelITSIS (pp. 208–218).