IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 37, No. 3, June, 2023, pp. 567-576
Journal homepage: http://iieta.org/journals/ria

A Comparative Study of Machine Learning Algorithms for Intrusion Detection in IoT Networks

Zahia Benamor*, Zianou Ahmed Seghir, Meriem Djezzar, Mounir Hemam

ICOSI Laboratory, Computer Science Department, Abbes Laghrour University, Khenchela 4000, Algeria

Corresponding Author Email: benamor.zahia@univ-khenchela.dz

https://doi.org/10.18280/ria.370305

Received: 1 May 2023
Accepted: 28 May 2023

Keywords:
feature selection, Internet of Things, intrusion detection system, Machine Learning, security

ABSTRACT

The pervasive threat of cyberattacks jeopardizes the security and privacy of the Internet of Things (IoT) landscape, spanning devices to networks. To counter these attacks, research has been directed towards the development of effective and appropriate countermeasures. Intrusion Detection Systems (IDSs), particularly those leveraging Machine Learning (ML) techniques for expedited attack detection, are currently recognized as some of the most potent solutions for preserving the integrity of the IoT environment. This study was conducted with the objective of evaluating the efficacy of supervised Machine Learning techniques, specifically, Random Forest (RF), Decision Trees (DT), and XGBoost classifiers, in detecting attacks within the IoT network. Chi-square (Chi2) and Mutual Information served as the employed Feature Selection Techniques. The research utilized two recent datasets for model evaluation. In pursuit of an optimal solution and high IDS model accuracy, a comparison of different techniques was undertaken across each stage of the ML workflow. The performance of the algorithms was assessed using the Edge-IIoT and BoTNeTIoT datasets, and the results from the two were compared. The impact of each workflow step on the model's accuracy was also examined. According to the performance metrics, the best results were achieved with the Mutual Information and XGBoost combination, outperforming both the Random Forest and Decision Tree classifiers. This study thus contributes to the ongoing efforts to strengthen IoT security through enhanced intrusion detection techniques.

1. INTRODUCTION

Intrusion Detection Systems (IDSs) in IoT are designed to detect and protect against unauthorized access or malicious activities on a network, system, or application. They can be used to detect and respond to any malicious activity related to the Internet of Things (IoT). They can also help identify attacks and potential breaches, allowing an organization to take appropriate action before damage is done. Machine Learning algorithms can analyze large amounts of data to identify patterns that could indicate attacks. These algorithms use various techniques such as supervised learning, unsupervised learning, and reinforcement learning to recognize trends in the data and to look for anomalies or malicious activity. This allows systems to be better equipped to understand and defend against malicious activity. Common Machine Learning algorithms used for this kind of implementation include Support Vector Machines (SVM), Artificial Neural Networks (ANN), Naive Bayesian Classifiers (NB), Decision Trees (DT), and Random Forests (RF). These algorithms enable systems to defend more accurately against malicious activities. Each ML algorithm operates differently from one another. SVM works by finding an optimal hyperplane that separates different classes of data points. In IDS, SVM can be applied to classify network traffic as normal or malicious based on the extracted features [1]. ANN is a neural network model that learns patterns from data, inspired by biological neural networks. It can be applied in IDS to identify anomalies and classify network traffic using the learned patterns [2].

This study focuses on studying the influence of various workflow steps on the performance of ML classifiers in IDS rather than comparing the classifiers themselves. Specifically, it examines the impact of Feature Selection techniques, test split ratio variations, and the choice of datasets for evaluating the experiments. By exploring different combinations of these techniques within the context of different IDS types, such as binary and multiclass classification, the study aims to gain insights into how each step affects the overall performance.

The IDS workflow must include the feature selection procedure. By selecting a small sample of highly significant features from a dataset, the Feature Selection speeds up model execution and improves accuracy. In other words, it identifies features that can distinguish between samples from various classes. There are several methods for choosing the most relevant features, including filter, wrapper, and embedding strategies [3]. The filter method employs statistical measures to select relevant attributes, whereas the wrapper method evaluates the performance of the learning algorithm using different feature subsets. The last method is a combination of both previous methods.

Our research work aims to evaluate existing Machine Learning classifiers and study the impact of each IDS workflow step on the model performance in terms of binary and multiclass classification. For further insights:

(1) Choosing two newest datasets that contain several new attacks.
(2) Selecting a significant features subset using two different Feature Selection techniques.
(3) Studying the effect of train/test data splitting on the model performance.
(4) Using three supervised ML classifiers and evaluating their performance in term of accuracy, precision, recall, and F1-score.

The remainder of this paper is organized as follows: Section 2 focuses on works related to Machine Learning for Intrusion Detection Systems (IDS) in the Internet of Things. Additionally, Section 3 covers various comparative studies on Intrusion Detection Systems using Machine Learning. Section 4 explains the proposed methods, while Sections 5 and 6 discuss experiments and results, respectively. Finally, Section 7 provides a conclusion and outlines future directions for research in this area.

2. RELATED WORKS

Several research works use Machine Learning for Intrusion Detection Systems (IDS) in IoT. These works aim to develop better algorithms and technologies to detect malicious activity on networks, systems, and applications connected to the Internet of Things. Research works have developed a range of techniques such as using supervised and unsupervised learning techniques, deep learning techniques, and statistical methods.

Imad et al. [2] compared the existing works in term of the dataset used, attack/intrusion category, and the obtained results. Subsequently, a proposed model was implemented, wherein the significant features from the Network Intrusion Detection dataset were extracted using Random Forest (RF). The purpose of this extraction was to identify and categorize the data instances into anomalous and normal based on the extracted features. They applied five Machine Learning approaches (NB, DT, KNN, ANN, and LR). The outcomes demonstrated how well the DT model performed, achieving the best accuracy of 100% among all the classifiers. The authors got a perfect model in binary classification that can detect the existence of the attack without any specification about the type of this attack. Additionally, they used one dataset to evaluate their experiment.

Another work compared the existing works based on the following parameters: dataset, classifiers, evaluation matrix, and findings. Additionally, to specify the best classifier for identifying intrusion detection, Almomani et al. [4] compared ten Machine Learning classifiers: LR, multinomial NB, Gaussian NB, Bernoulli NB, KNN, DT, Adaptative boosting, RF, MLP, and Gradient Boosting. The experimental results showed that the RF outperforms the other classifiers in terms of accuracy at 87% on the UNSW-NB15 dataset. The feature selection techniques are not utilized in this work, and there is also no implementation of multiclass classification.

In their performance analysis of Machine Learning algorithms in IDS, Saranya et al. [5] presented a survey of intrusion detection using ML algorithms. Next, a comparative analysis was conducted to evaluate the performance of various classifiers, including Linear Discriminant Analysis (LDA), Classification and Regression Trees (CART), RF, SVM, NB, DT, and Logistic Regression (LR). The experimentation was performed using the KDDcup99 dataset. The findings revealed that the RF classifier achieved the highest accuracy level, reaching 99.81%. This notable performance can be due to the utilization of a filter method that effectively reduced the number of features from 42 to 20. However, it is important to note that the proposed method was not specifically investigated for multiclass classification in this study.

Das et al. [6] mentioned the pros and cons, the used dataset, classification method, feature selection method, and preprocessing of each work in their comparative analysis. Besides this, they build their IDS model, where Ensemble Feature Selection, Ensemble Machine Learning classifiers, and three datasets are used to achieve higher accuracy with a lower False Positive Rate. They used majority voting to select the best features subset (EnFs) using ANOVA, Chi2, Lasso, LRL1, MutInfo, Pearson, RF, RFE, and SFPR techniques. They then applied LR, DT, NB, NN, and SVM classifiers to detect intrusions in NSL-KDD, UNSW-NB15, and CIC-IDS2017 datasets. The absence of the multiclass classification was notable in this study.

3. COMPARING VARIOUS COMPARATIVE STUDIES OF IDS USING ML

The privacy of users' information necessitates high-level networks and devices secure. This security is an obligatory and a big challenge for researchers nowadays. An Intrusion Detection System is required to complete this challenge, and many studies and experiments have been conducted in this field to develop an efficient technique. They compared several Machine and Deep Learning techniques to find the best model that can detect attacks with the highest accuracy, and lowest False Alarm Rate.

We divided these studies depending on the relied method in their comparative studies into two categories: In the first category, researchers aimed to compare the published work on IDS, highlight the advantages and disadvantages, and the limitations of each model.

The researchers, on the other hand, prefer to build their IDS model using various ML techniques and compare the results to find the most efficient one.

3.1 Comparing of existing work

Comparison of existing works is done in this category, where the researchers summarize, categorize and collect different recent researches in IDS using ML, and compare them in terms of pros and cons, advantages and disadvantages. Furthermore, the authors may compare the existing work based on the metrics of evaluation performance.

In their comparative study, Bhatia et al. [7] examined various research papers published between 2016 and 2020, demonstrating the benefits and drawbacks of available intrusion detection techniques that can be applied to large amounts of data. They also compared the accuracy of these research papers.

Kathiresan et al. [8] investigated various Machine Learning based classification methods to detect intrusions in network traffic. They compared various methods and focused on the advantages and disadvantages of each of them.

Boyanapalli and Shanthini [9] aimed to categorize the existing newest research papers on Intrusion Detection Systems in the Internet of Things environment. They are based on the dataset used in each method, as well as the authors mentioned the highlights of these IDSs. They also compared the IDS performance of each method.

Anushiya and Lavanya [10] discuss various attack detection techniques in this comparative study, where traditional methods, Machine Learning, and Deep Learning methods for intrusion detection in IoT are compared in their work. They concentrated on the advantages and disadvantages of the chosen papers.

In a recently published paper by Sangwan and Chhillar [11], the authors studied newly used ML algorithms in securing IoT devices, where they compared the accuracy of various ML classification techniques. Additionally, they highlighted the limitations of each research paper.

3.2 Comparing of proposed models

Building efficient IDSs is a significant challenge for researchers in general. Because the developed IDSs should be more accurate in detecting intrusions and attacks. Furthermore, IDSs must generate low False Positive Rates, and be able to handle large amounts of data for training with a large number of features.

To deal with this issue, researchers proposed several methods with different steps: collecting data, feature selection, and evaluating model, then comparing the results to select the best one.

3.2.1 Comparative study based on ML classifier with its parameters and types

In this category, all of the focus is on a single Machine Learning classifier, which compares its types and parameters to determine which one will provide the best performance of an intrusion detection system.

Agrawal and Singh [12] analyzed the different types and characteristics of the SVM classifier in order to determine its optimal performance for the anomaly detection system. On the NSL KDD dataset, multiple SVM kernels: linear, poly, RBF (Radial Basis Function kernel), and sigmoid are used to build the SVM classifier effectively. The experiment was conducted with different test split ratios of 0.20, 0.50, and 0.60. They investigated how each test ratio affected the effectiveness of the SVM kernel's classifier. Through their experiments, they found that the linear kernel was the best SVM kernel for detecting anomalies with 99.99% of accuracy.

3.2.2 Comparative study based on feature selection

Feature selection (FS) is a crucial problem in building a Machine Learning model. Decreasing the number of features by selecting the necessary one in the datasets can increase the accuracy and performance of the model. To achieve this issue, many Feature Selection (FS) techniques are available in the literature. Researchers try to compare different FS techniques to select the best one, which gives the best model's performance.

In order to address the security challenges of the Internet of Vehicles (IoV), Aswal et al. [13] studied six Machine Learning models. To categorize the BoTnet attack dataset, they employed the NB, KNN, LR, LDA, CART, and SVM models. Three steps are involved in feature selection: After eliminating useless features to reduce the total number of features to 71, 52 features are chosen using RF in the second step, and 35 features are selected using the correlation function in the last step. With 35 features, KNN and CART demonstrated the maximum accuracy, with 99.79% and 99.97%, respectively.

Ibrahim and Thanon [14] employed two methods to choose the right set of features to get the best accuracy in an enormous dataset: the ANOVA F-test and a Recursive Feature Elimination (RFE). On the NSL-KDD dataset for the IDS, three Machine Learning techniques: KNN, RF, and SVM are applied. The performance of these techniques was compared using both all features set and the 13 best features set. The findings showed the effectiveness of RF in detecting abnormal behavior with all features as well as with the selected subset of the best features.

3.2.3 Comparative study based on ML classifiers

Choosing and selecting the best Machine Learning classifier is a crucial step in building an Intrusion Detection System. Researchers compared the performance of different ML classifiers to get the optimal one to detect anomalies.

Mondal and Singh [15] compared the performance of eight ML techniques, namely: LR, KNN, DT, RF, Gaussian NB, AdaBoost, Gradient Boosting, and LDA to select the best one for classifying Network packets as normal or some kind of malicious attack. The performance of classifiers is evaluated on the network logs data, and the results showed that the DT and the Gradient Boosting were the most accurate model comparing to other classification techniques.

Manvith et al. [16] demonstrated the ability of ML techniques to identify various attacks to ensure network security. They analyzed and compared SVM, LR, and RF techniques to determine which one can be used to identify network attacks. The experiment results revealed that the RF outperformed the other techniques on the KDDcup99 dataset.

3.2.4 Comparative study based on datasets

In this category of comparative studies, the datasets play a major role to deploy the Machine Learning models; the researchers compare the performance of the Machine Learning model with different datasets to discover the effect of choosing the dataset on their ML models.

Dwibedi et al. [17] concentrated on data contribution by performing and analyzing three recently published datasets: UNSW-NB15, Bot-IoT, and CSE-CIC-IDS2018. They used RF, SVM, Keras Deep Learning models, and XGBoost on the aforementioned datasets, selecting ten features from each dataset and comparing the performance of Machine Learning classifiers.

Kilincer et al. [18] employed various Machine Learning models and compared their performance using five widely used datasets to detect intrusion: CSE-CIC IDS-2018, UNSW-15, ISCX-2012, NSL-KDD, and CIDDS-001. The results of deploying the KNN, SVM, and DT classifiers revealed that the DT classifier outperformed the other classifiers in terms of classifier performance.

3.2.5 Comparative study based on splitting test ratio

This comparative study category focuses on investigating the effect of split ratio tests on classification performance, where researchers compare multiple ratios splitting data into training and testing sets.

To detect email phishing, Al Fayoumi et al. [19] applied three distinct Machine Learning classifiers: SVM, RF, and NB. They compared the outcomes of diverse experiments conducted on three benchmarking testing levels using the following testing ratios: 0.5, 0.4, and 0.3. The experiment results showed that the SVM classifier has higher performance and efficiency than the other classifiers with a 0.3 testing ratio split. Table 1 summarizes the works mentioned above.

Table 1. Comparative study of existing works

Ref	IoT (Yes/No)	Dataset		Feature Selection		ML Classifier		Ratio of (Training: Test) Split		IDS Type			Best Model
		Nbre	Name	Nbre	Name	Nbre	Name	Nbre	Type	Binary	Multiclass Attack Type	Attack Class	
Agrawal and Singh [12]	Yes	1	NSL-KDD	N/M	N/M	1	SVM (linear RBF, poly sigmoid)	3	0.8:0.2 0.5:0.5 0.4:0.6	✓	✓	X	Linear SVM 0.2 test split ratio
Aswal et al. [13]	Yes	1	BoTnet attack	2	RF+ Correlation	6	NB, KNN LDA LR CART, SVM	1	0.6:0.4	✓	X	X	KNN and CART with FS
Ibrahim et al. [14]	N/M	1	NSL-KDD	2	ANOVA RFE	3	KNN, RF SVM	N/M	N/M	X	✓	X	RF
Mondal and Singh [15]	Yes	1	Collected data from network	1	Correlation	8	LR, KNN, DT RF, LDA ADA boost Gaussian NB Gradient boost	1	0.75:0.25	✓	X	X	DT, XGBoost
Manvith et al. [16]	No	1	KDDcup99	1	PCA	3	SVM LR RF	1	0.7:0.3	✓	X	X	RF
Dwibedi et al. [17]	Yes	3	UNSW-NB15 Bot-IoT CSE-CIC-IDS2018	1	Select 10 features from each dataset	4	RF, SVM Keras DL XGBoost	1	0.75:0.25	✓	✓	X	XGBoost
Kilincer et al. [18]	No	5	CSE-CIC-IDS2018 UNSW-NB15 ISCX-2012 NSL-KDD CIDDS-001	N/M	N/M	3	KNN DT SVM	N/M	N/M	✓	X	X	DT
Al Fayoumi et al. [19]	N/M	1	Emails	N/M	N/M	3	SVM RF NB	3	0.7:0.3 0.6:0.4 0.5:0.5	✓	X	X	SVM with 0.3 test split ratio

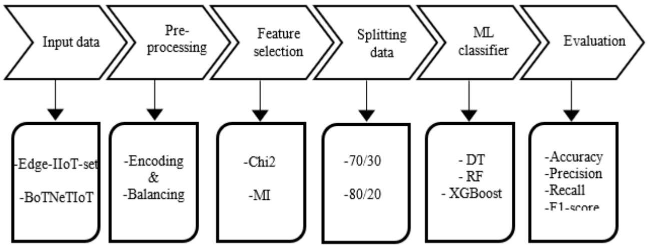

Figure 1. The workflow of the proposed model

4. PROPOSED MODEL

Data collection, Feature Selection (FS), Machine Learning (ML) classifier and model evaluation steps are typically the four primary stages of the IDS workflow. Each one is crucial and has an impact on how well the model works. In the studies related to this topic, researchers have used various workflow strategies to compare multiple Machine Learning (ML) classifiers and determine which model performs the best. These studies have followed one of two approaches. The first approach involves using a single Feature Selection (FS) technique alongside different ML techniques. The second approach focuses on using only one FS technique throughout the evaluation process. Moreover, it is worth noting that these studies did not investigate the influence of the data splitting test ratio on model performance. Additionally, there was no comparison of model performance between binary and multiclass classifications.

This paper proposed several IDS process step combinations to select the best one that can detect correctly and accurately all the attacks in the IoT network. First, two new IoT datasets were chosen, as depicted in Figure 1, to evaluate the model. Second, two Feature Selection techniques, namely Chi2 and Mutual Information classifier (MI), were employed to reduce the dimensionality of the datasets and select the most significant subset of features. Subsequently, the datasets were divided into train and test sets using the 80/20 and 70/30 ratio. Next, three key Machine Learning classifiers, namely XGBoost, Decision Trees (DT), and Random Forest (RF), were selected to classify the data into normal and abnormal categories in binary classification, as well as normal or one attack category and normal or one attack type in multiclass classification. Finally, the performance of all these combinations was assessed by computing scores (accuracy, precision, recall, and F1-score) and comparing them to determine the most effective approach.

5. EXPERIMENTS

The experiments are implemented on a PC with Intel(R) core (TM) i5-4210U CPU @ 1.70GHz 2.40GHz of processor and 12GB RAM, and 64-bits windows 7 operating system, we have used Google Colab to run the code. Python, as well as the Scikit-learn, Pandas, and matplotlib libraries were used to create the model.

5.1 Data collection

This proposed work utilized two new related IoT datasets for intrusion detection from different research organizations to train and test the model and they are Edge-IIoT set Ferrag et al. [20] and BoTNeTIoT datasets Alhowaide et al. [21]. Each dataset has three types of targets including binary (Label), category (Attack category), and subcategory (Attack type).

We utilized the EdgeIIoT dataset due to its recent release and extensive coverage of various attack types spanning 5 categories and 14 attacks type. On the other hand, we also used the BotnetIoT dataset, which, in contrast, encompasses only two categories of attacks. The rationale behind this approach was to demonstrate the robustness and effectiveness of the model across diverse datasets, displaying its ability to effectively detect and mitigate attacks in different contexts.

Edge-IIoT set: is a new cyber security dataset of IoT and IIoT applications. It has 61 features and 20 952 649 instances with 14 types of attack which were mapped to six attack categories, namely DDoS attacks, Information gathering attack, Man in the Middle attacks, Injection attacks, and Malware attacks. Table 2 shows the distribution percentage of instances in each category.

Table 2. Instance distribution in Edge-IIoT set dataset

Label	Category	Type	Instances (%)	Subtotal (%)	Total (%)
Normal	Normal	Normal	53.57	53.57	53.57
Attacks	DoS/DDoS attacks	DDoS TCP	9.64	39.92	46.43
		DDoS UDP	15.28		
		DDoS HTTP	1.09		
		DDoS ICMP	13.91		
	Information Gathering attack	Port Scanning	0.11	0.81	
		Fingerprinting	0.005		
		Vulnerability scanner	0.70		
	MITM attacks	MITM	0.01	0.01	
	Injection attacks	XSS	0.08	0.50	
		SQL Injection	0.24		
		Uploading	0.18		
	Malware attacks	Backdoor	0.12	5.20	
		Password	5.03		
		Ransomware	0.05		

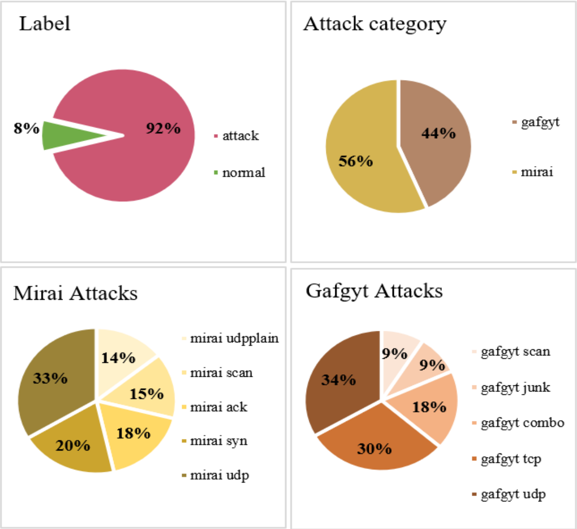

Figure 2. BoTNeTIoT dataset description

BoTNeTIoT: only two attack categories are included in the BotNetIoT dataset Mirai and Gafgyt botnet attacks. There are numerous new or recent IoT attacks on this dataset and it contains 23 features. Figure 2 shows how the instances are distributed on this dataset.

5.2 Preprocessing

Any Machine Learning classifier requires a cleaned, transformed, normalized, and feature reduced dataset as a feed to evaluate the model.

From the Edge-IIoT set, we used the ML-EdgeIIot-dataset.csv file, which contains a portion of the dataset and is used for evaluating Machine Learning models based Intrusion Detection Systems. For the multiclass classification model, we created a new target feature called the Attack_category that regroups the Attacks type into five classes namely DDoS, Injection Attack, Information Gathering Attack, MITM, and Malware Attack. We also used the get dummies package to convert the non-numeric data into numeric as follows: pandas.get_dummies (dataset).

Then we standardized features using MinMaxScaler technique:

$$New\_x \;=\; \frac{x - Min(x)}{Max(x) - Min(x)} \tag{1}$$

where, *New_x* is the new value of the feature that lies between 0 and 1. *Min(x)* and *Max(x)* are the minimum and the maximum feature value, respectively. Finally, to address the challenge of imbalanced data and mitigate the risk of overfitting or underfitting, we employed the SMOTE (Synthetic Minority Over-sampling) technique on our dataset. This approach generates synthetic samples of the minority class to rebalance the data distribution. We utilized the *SMOTETomek* technique, which combines the oversampling of the minority class with the undersampling of the majority class.

On the other hand, from the BoTNeTIoT dataset we used BoTNeTIoT-L01-v2.csv part that contains just two attack categories Mirai and Gafgyt botnet, all its features are numeric and it does not need any transformation, whereas MinMaxScaler is applied to scale data, than we balanced the scaled data manually.

5.3 Feature selection

ML-EdgeIIot-dataset and BoTNeTIoT-L01-v2 contain numerous features (61 and 23 features, respectively), where some of which are unimportant, cannot affect the output label, produce over-fitting or under-fitting, increase the training time, and reduce the performance of the model. Due to these factors, we choose only the most crucial features using the Mutual Information classifier (MI) and Chi-square (Chi2) Feature Selection techniques. We employed the MI with fixed predefined threshold as a cutoff or a fixed number for feature selection in both the EdgeIIoT and BotnetIoT datasets. For the EdgeIIoT dataset, we utilized a threshold of 0.2, while for the BotnetIoT dataset; we applied a higher threshold of 0.4. Additionally, in the Chi2 feature selection method, we considered the number of features obtained from the MI (same number of features).

Figure 3 shows the rank of the most important and selected features subset using MI in ML-Edge-IIoT-dataset with three

IDS types. In Figure 4, the preprocessing outcomes are visually represented, providing insights into the number of features for the two datasets. While Table 3 demonstrates the impact of the SMOTETomek balancing technique on the EdgeIIoT dataset. It provides a comprehensive overview of the changes in sample distribution on each attack type resulting from the SMOTETomek technique, highlighting how it helps to handle the imbalances within the dataset.

5.4 Splitting data

For the performance examination of models and to investigate the influence of the splitting strategies of training and testing datasets used ML classifier to detect the intrusions, we utilized two different ratios 80/20 and 70/30 to split the datasets into the training and testing sets. These two split ratios were chosen due to their superior performance when compared to the ratios of 60/40 and 50/50, as indicated in Table 1. The training set helps in model learning, whereas the testing set allows the evaluation of model performance.

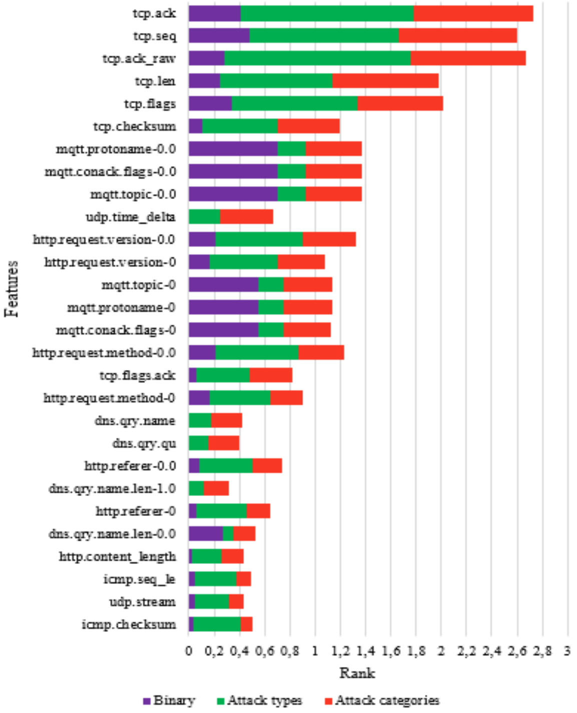

Figure 3. The selected features from ML-Edge-IIoT-dataset using MI

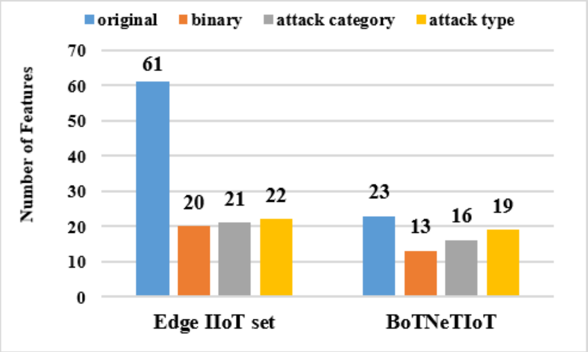

Figure 4. The selected features from ML-Edge-IIoT-dataset using MI

Table 3. Instance distribution in Edge-IIoT set dataset

Attack Type	Before Sampling	After Sampling
DDoS_UDP	14498	24124
DDoS_ICMP	13096	24108
DDoS_HTTP	10495	17418
SQL_injection	10282	17661
DDoS_TCP	10247	17537
Uploading	10214	17869
Vulnerability_scanner	10062	22928
Password	9972	15479
Backdoor	9865	23225
Ransomware	9689	22722
XSS	9552	21264
Port_Scanning	8924	16883
Fingerprinting	853	21177
MITM	358	24125

Table 4. Confusion matrix

Class		Predicted Class	
		Normal	Attack
Actual class	Normal	TP	FN
	Attack	FP	TN

True Positive (TP), True Negative (TN) False Positive (FP), False Negative (FN).

5.5 ML classifiers

Based on a comparative analysis of existing works presented in Table 1, we opted to utilize the top three supervised Machine Learning classifiers, namely Random Forest (RF), Decision Tree (DT), and eXtreme Gradient Boosting (XGBoost), with their default parameter settings, to categorize the data into normal, attack category, or attack type depending on the IDS type (binary or multiclass classification).

5.6 Performance and evaluation

The confusion matrix (Table 4) is the basis on which we build the performance of the model and calculate the various evaluation metrics such as:

• Accuracy: is used to calculate the ratio of correct classifications to all samples, defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

• Precision: is the percentage of correct attack classes to all predicted attack classes, and it can be calculated as follows:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

• Recall: indicates the percentage of correctly classified attacks compared to the total number of samples that should have been identified as attacks, it is calculated as follows:

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

• F1 Score: the harmonic average of Precision and Recall, it gives a balance between both metrics Precision and Recall, which is given by:

$$F1\ score = 2*\frac{Precision * Recall}{Precision + Recall} \tag{5}$$

6. RESULTS AND DISCUSSION

In this section, the findings from our experiments are presented and discussed. The outcomes of our experiments are summarized in Tables 5 and 6, where they contain the weighted average Precision, Recall, and F1-score values obtained after applying training and testing the model for RF, DT, and XGBoost on the ML-Edge-IIoT-dataset and BoTNeTIoT-L01-v2 datasets respectively. Utilizing the features subset picked by MI and Chi2 techniques based on 80/20 and 70/30 splitting ratios. On the other hand, Figures 5 and 6 show the accuracy of the three ML classifiers in ML-Edge-IIoT-dataset and BoTNeTIoT-L01-v2 dataset, respectively.

Table 5. Classification model evaluation on the Ml-Edge-IIoT dataset

			ML Classifier/Metrics								
IDS Type	FS Techniques	Test Split Ratio	DT			RF			XGBoost		
			Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
Binary	MI	0.2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		0.3	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	Chi2	0.2	0.87	0.87	0.87	0.88	0.87	0.87	0.94	0.94	0.94
		0.2	0.87	0.87	0.87	0.88	0.87	0.87	0.94	0.94	0.94
Attack type	MI	0.2	0.92	0.92	0.92	0.92	0.9	0.9	0.93	0.93	0.93
		0.3	0.92	0.92	0.92	0.92	0.9	0.9	0.93	0.93	0.93
	Chi2	0.2	0.91	0.91	0.91	0.91	0.88	0.89	0.92	0.91	0.91
		0.3	0.91	0.91	0.91	0.91	0.89	0.89	0.92	0.91	0.91
Attack category	MI	0.2	0.95	0.95	0.95	0.95	0.94	0.94	0.97	0.96	0.96
		0.3	0.94	0.94	0.94	0.95	0.94	0.94	0.97	0.96	0.96
	Chi2	0.2	0.93	0.93	0.93	0.95	0.94	0.94	0.95	0.94	0.94
		0.3	0.93	0.93	0.93	0.95	0.94	0.94	0.96	0.95	0.95

Mutual Information (MI), Decision Tree (DT), Random Forest (RF),eXtreme GradientBoosting (XGBoost), Precision (Pr), Recall (Re), F1-score (F1).

Table 6. Classification model evaluation on the BoTNeTIoT-L01-V2 dataset

			ML Classifier/Metrics								
IDS Type	FS Techniques	Test Split Ratio	DT			RF			XGBoost		
			Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
Binary	MI	0.2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		0.3	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	Chi2	0.2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		0.2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Attack type	MI	0.2	0.90	0.84	0.82	0.93	0.90	0.89	1.00	1.00	1.00
		0.3	0.90	0.84	0.82	0.93	0.90	0.89	1.00	1.00	1.00
	Chi2	0.2	0.90	0.84	0.82	0.93	0.91	0.91	1.00	1.00	1.00
		0.3	0.90	0.84	0.82	0.93	0.91	0.90	1.00	1.00	1.00
Attack category	MI	0.2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		0.3	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	Chi2	0.2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		0.3	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Mutual Information (MI), Decision Tree (DT), Random Forest (RF),eXtreme GradientBoosting (XGBoost), Precision (Pr), Recall (Re), F1-score (F1).

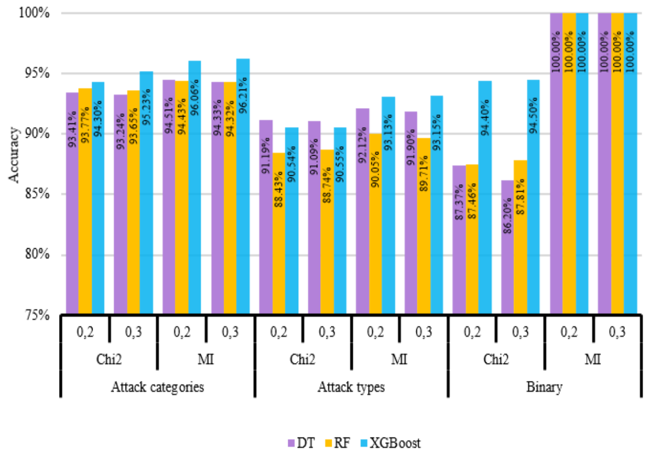

Figure 5. Accuracy results in ML-Edge-IIoT-dataset

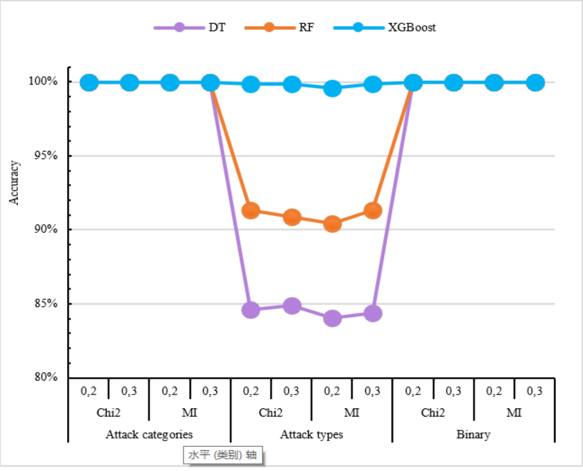

Figure 6. Accuracy results in BoTNeTIoT-L01-v2 dataset

6.1 ML-Edge-IIot-dataset

The experiment results using ML-Edge-IIoT-dataset are discussed below.

6.1.1 Binary classification

In this type of classification, the MI feature selection technique succeeded in selecting the most suitable feature subset from ML-Edge-IIoT-dataset features that helped the RF, DT, and XGBoost to achieve the highest accuracy (100%), whatever the splitting ratio. Additionally, all of the tested ML classifiers that used the same features subset had the highest Precision, Recall, and F1-score.

6.1.2 Multiclass classification

In this type of IDS model, we tested the ability of our model to detect not only the normal and attack classes, but also the category of attack in addition to attack type.

Attack category: with ML-Edge-IIoT-dataset, DT and RF achieved almost similar accuracy with selected features subset using the Mutual Information technique, no matter the splitting ratio. Whereas the XGBoost classifier achieved the highest intrusion detection performance in accuracy (96.21%) with this type of classification. Additionally, XGBoost reached the best values of Precision, Recall, and F1-score in the range of 93%-97%.

Attack type: the accuracy values in this experiment are reduced significantly with all three ML classifiers in ML-Edge-IIoT-dataset. RF achieved the lowest value with the Chi2 technique and 0.2 splitting test ratio, while the highest accuracy was obtained from the XGBoost classifier with MI and 0.3 splitting test ratio. The results of the weighted averages are similar in 0.2 and 0.3 splitting test ratio with almost all Machine Learning classifiers.

6.2 BoTNeTIoT-L01-v2 dataset

We use the BoTNeTIoT-L01-v2 dataset to compare the performance of these models. Figure 4 demonstrates that all the ML classifiers' performance was similar and achieved the height accuracy, when the MI and Chi2 Features Selection techniques were used in binary and multiclass classification (i.e., Attack category). Where they can correctly detect the normal and abnormal classes as well as the category of attack, as the performance metrics shown in Table 6.

On the other hand, the performance of the proposed model decreases when detecting attack types in multiclass classification. The accuracy drops by 15% with the decision tree (DT) classifier and about 10% with the Random Forest (RF) classifier. In contrast, the XGBoost classifier consistently performs well, reaching an accuracy of 99.86%. Moreover, the evaluation results show that for the binary and attack category IDS, all classifiers consistently achieved perfect precision, recall, and F1 scores of 100% using different FS techniques (MI and Chi2) and test split ratios (0.2 and 0.3), indicating accurate classification of instances as normal/attack, or normal/attack category. In the case of the Attack Type IDS, the classifiers achieved high precision, recall, and F1 scores ranging from 82% to 93% when using the MI and Chi2 FS techniques.

6.3 Comparison with other methods

The introducer of the Edge-IIoT set dataset executed five ML techniques DT, RF, SVM, KNN, and DNN in binary and multiclass classification. Table 7 depicts the comparative accuracy results of similar detection classifiers. For binary classification results, our proposed DT and RF methods act better than the existing DT and RF methods. On the other hand, the results obtained by RF and DT are higher than the other work by 11.53% and 16.61%, respectively, in detecting the attack categories (multiclass classification). The difference is likewise observable with the detection of attack types (multiclass classification), where our model has 25.01% high accuracy with DT and 9.22% with RF. These observable distinctions are due to deploying MI Features selection techniques in our model. That confirms the effect of the feature selection step on the model performance.

Table 7. Accuracy comparison of our model and existing works

			Ferrag et al. [20]	**Our Model**
Binary		XGBoost	-	100%
		RF	99.99%	100%
		DT	99.98%	100%
Multiclass classification	Attack category	XGBoost	-	96.21%
		RF	82.90%	94.43%
		DT	77.90%	94.51%
	Attack type	XGBoost	-	93.15%
		RF	80.83%	90.05%
		DT	67.11%	92.12%

In order to identify the optimal solution and achieve high accuracy in the IDS model, we compared various combinations of ML techniques. We selected MI and Chi2 as FS techniques, then used the DT, RF, and XGBoost classifiers for the classification task. The evaluation of our model was done on the Edge-IIoT and BoTNeTIoT datasets, with both binary and multiclass classification scenarios.

7. CONCLUSION AND FUTURE WORK

The evaluation of various Feature Selection strategies, Machine Learning classifiers, and test split ratios and their effects on detecting intrusions in the IoT environment, categorizing attacks, and identifying their types are the main focuses of this research. The accuracy of the model is evidently impacted by the Features Selection method chosen. The Mutual Information technique improves model performance and produces comparatively better results than the Chi2 technique. Additionally, selecting the type of Machine Learning classifiers to use in the IDS model is a crucial workflow step. As we discovered in our study, the XGBoost classifier continues to outperform DT and RF in terms of accuracy for binary and multiclass classification. Furthermore, the performance of the model with a 0.2 test split ratio is approximately the same as the 0.3 test split ratio, which shows this step had a negligible impact on the model's performance. Finally, the dataset also plays an essential role and affects the model performance, where the well preprocessing data gives the best performance.

According to the results of our study, it is observed that the choice of feature selection method has a noticeable impact on the accuracy of the model. Specifically, the MI technique yielded better results and improved the overall performance compared to the Chi2 technique. Moreover, the selection of appropriate ML classifiers is a crucial step in the workflow of intrusion detection models. The study found that the XGBoost classifier consistently outperformed the DT and RF classifiers in terms of accuracy, both for binary and multiclass classification tasks. Additionally, the research explored the influence of test split ratios on the model's performance. Interestingly, it is discovered that there was only a negligible difference in performance between a 0.2 test split ratio and 0.3. This indicates that the choice of test split ratio had minimal impact on the model's accuracy. Lastly, the dataset used in the research was found play a significant role in affecting the model's performance. Well-preprocessed data was found to result in the best performance, highlighting the importance of proper data preprocessing techniques.

Based on the results obtained, the combination MI-XGBoost shows superior performance as the most effective model for detecting attacks in IoT networks in both binary and multiclass classifications.

Future research will encompass further experimentation involving additional datasets, such as the CICIoT2023 dataset.

Feature selection strategies like Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) will be explored, alongside the utilization of Deep Learning approaches, including CNN and LSTM classifiers.

REFERENCES

[1] Jing, D., Chen, H.B. (2019). SVM based network intrusion detection for the UNSW-NB15 dataset. In 2019 IEEE 13th International Conference on ASIC (ASICON), IEEE, pp. 1-4. https://doi.org/10.1109/ASICON47005.2019.8983598

[2] Imad, M., Abul Hassan, M., Hussain Bangash, S., Naimullah. (2022). A comparative analysis of intrusion detection in iot network using machine learning. In Big Data Analytics and Computational Intelligence for Cybersecurity, Cham: Springer International Publishing, 149-163. https://doi.org/10.1007/978-3-031-05752-6_10

[3] Fenanir, S., Semchedine, F., Baadache, A. (2019). A machine learning-based lightweight intrusion detection system for the internet of things. Revue d'Intelligence Artificielle, 33(3): 203-211. https://doi.org/10.18280/ria.330306

[4] Almomani, O., Almaiah, M.A., Alsaaidah, A., Smadi, S., Mohammad, A.H., Althunibat, A. (2021). Machine learning classifiers for network intrusion detection system: Comparative study. In 2021 International Conference on Information Technology (ICIT), IEEE, pp. 440-445. https://doi.org/10.1109/ICIT52682.2021.9491770

[5] Saranya, T., Sridevi, S., Deisy, C., Chung, T.D., Khan, M.A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review. Procedia Computer Science, 171: 1251-1260. https://doi.org/10.1016/j.procs.2020.04.133

[6] Das, S., Saha, S., Priyoti, A.T., Roy, E.K., Sheldon, F.T., Haque, A., Shiva, S. (2021). Network intrusion detection and comparative analysis using ensemble machine learning and feature selection. IEEE Transactions on Network and Service Management, 19(4): 4821-4833. https://doi.org/10.1109/TNSM.2021.3138457

[7] Bhatia, V., Choudhary, S., Ramkumar, K.R. (2020). A comparative study on various intrusion detection techniques using machine learning and neural network. In 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), IEEE, pp. 232-236. https://doi.org/10.1109/ICRITO48877.2020.9198008

[8] Kathiresan, V., Karthik, S., Divya, P., Rajan, D.P. (2022). A comparative study of diverse intrusion detection methods using machine learning techniques. In 2022 International Conference on Computer Communication and Informatics (ICCCI), IEEE, pp. 1-6. https://doi.org/10.1109/ICCCI54379.2022.9740744

[9] Boyanapalli, A., Shanthini, A. (2021). A comparative study of techniques, datasets and performances for intrusion detection systems in IoT. In Artificial Intelligence Techniques for Advanced Computing Applications: Proceedings of ICACT 2020, Springer Singapore, pp. 225-236. https://doi.org/10.1007/978-981-15-5329-5_22

[10] Anushiya, R., Lavanya, V.S. (2021). A comparative study on intrusion detection systems for secured communication in internet of things. ICTACT Journal on Communication Technology, 6948: 2527-2537. https://doi.org/10.21917/ijct.2021.0373

[11] Sangwan, U., Chhillar, R.S. (2022). Comparison of various classification techniques in cyber security using Iot. International Journal of Intelligent Systems and Applications in Engineering, 10(3): 334-339.

[12] Agrawal, A.P., Singh, N. (2021). Comparative analysis of SVM kernels and parameters for efficient anomaly detection in IoT. In 2021 5th International Conference on Information Systems and Computer Networks (ISCON), IEEE, pp. 1-6. https://doi.org/10.1109/ISCON52037.2021.9702398

[13] Aswal, K., Dobhal, D.C., Pathak, H. (2020). Comparative analysis of machine learning algorithms for identification of BOT attack on the internet of vehicles (IoV). In 2020 International Conference on Inventive Computation Technologies (ICICT), IEEE, pp. 312-317. https://doi.org/10.1109/ICICT48043.2020.9112422

[14] Ibrahim, Z.K., Thanon, M.Y. (2021). Performance comparison of intrusion detection system using three different machine learning algorithms. In 2021 6th International Conference on Inventive Computation Technologies (ICICT), IEEE, pp. 1116-1124. https://doi.org/10.1109/ICICT50816.2021.9358775

[15] Mondal, B., Singh, S.K. (2022). A comparative analysis of network intrusion detection system for iot using machine learning. In Internet of Things and Its Applications: Select Proceedings of ICIA 2020, Singapore: Springer Nature Singapore, 825: 211-221. https://doi.org/10.1007/978-981-16-7637-6_19

[16] Manvith, V.S., Saraswathi, R.V., Vasavi, R. (2021). A performance comparison of machine learning approaches on intrusion detection dataset. In 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), IEEE, pp. 782-788. https://doi.org/10.1109/ICICV50876.2021.9388502

[17] Dwibedi, S., Pujari, M., Sun, W.Q. (2020). A comparative study on contemporary intrusion detection datasets for machine learning research. In 2020 IEEE International Conference on Intelligence and Security Informatics (ISI), IEEE, pp. 1-6. https://doi.org/10.1109/ISI49825.2020.9280519

[18] Kilincer, I.F., Ertam, F., Sengur, A. (2021). Machine learning methods for cyber security intrusion detection: Datasets and comparative study. Computer Networks, 188: 107840. https://doi.org/10.1016/j.comnet.2021.107840

[19] Al Fayoumi, M., Odeh, A., Keshta, I., Aboshgifa, A., AlHajahjeh, T., Abdulraheem, R. (2022). Email phishing detection based on naïve Bayes, random forests, and SVM classifications: A comparative study. In 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), IEEE, pp. 0007-0011. https://doi.org/10.1109/CCWC54503.2022.9720757

[20] Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H. (2022). Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access, 10: 40281-40306. https://doi.org/10.1109/ACCESS.2022.3165809

[21] Alhowaide, A., Alsmadi, I., Tang, J. (2019). Features quality impact on cyber physical security systems. In 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), IEEE, pp. 0332-0339. https://doi.org/10.1109/IEMCON.2019.8936280