IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.18, No.1, January 2024, pp. 25~36
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.85834

Ensemble Method for Anomaly Detection on the Internet of Things

Kurniabudi*[1], Eko Arip Winanto[2], Lola Yorita Astri[3] , Sharipuddin[4]
[1,4]Department of Information System, Faculty of Computer Science,
[2,3]Department of Computer System, Faculty of Computer Science,
Universitas Dinamika Bangsa, Jambi, Indonesia
e-mail: ***[1]kbudiz@yahoo.com**, [2] ekoaripwinanto@unama.ac.id , [3]lolayoritaastri@unama.ac.id,
[4]sharipuddin@unama.ac.id

Abstrak

Pertumbuhan jumlah aplikasi, peralatan dan protokol yang terhubung pada Internet of Things (IoT) menghasilkan data dengan heterogenitas yang signifikan dan volume lalu lintas serta ketidakseimbangan data terus meningkat. Di sisi lain, jenis serangan baru terhadap jaringan IoT dimungkinkan karena kemajuan teknologi dan pengetahuan. Mengingat besarnya volume lalu lintas data, mekanisme deteksi harus memiliki kemampuan untuk membedakan berbagai bentuk serangan. Idealnya, sistem deteksi serangan harus dapat diandalkan dalam distribusi data yang tidak seimbang. Pemilihan fitur chi-square dipilih untuk menangani dimensi data yang besar. Metode ansambel diusulkan dalam penelitian ini untuk meningkatkan kinerja deteksi anomali pada data yang tidak seimbang. Beberapa algoritma klasifikasi, termasuk Bayes Network (BN), Naive Bayes (NB), REPTree, dan J48, digabungkan untuk menghasilkan metode deteksi yang ideal. Penelitian ini menggunakan dataset CICIDS-2017 karena sudah teruji dan sering digunakan dalam penelitian IDS. Kesimpulan bahwa Ensemble-3 lebih unggul daripada pendekatan lain dan penelitian sebelumnya dapat diambil dengan mengevaluasi kinerjanya.

Kata kunci— *Anomaly Detection, CICIDS-2017, Feature selection, Chi-square, The ensemble method*

Abstract

The growth in the number of applications, equipment and protocols connected to the Internet of Things (IoT) generates data with significant heterogeneity and traffic volumes and data imbalances continue to increase. On the other hand, new kinds of attacks on IoT networks are made possible by advancements in technology and knowledge. Given the substantial volume of data traffic, a detection mechanism must be able to discern various forms of attacks. Ideally, the attack detection system must be reliable in unbalanced data distribution. Chi-square feature selection was chosen to deal with large data dimensions. In order to enhance intrusion detection on imbalanced data, an ensemble method is proposed in this study. The optimal detection approach is created by combining several classification methods, including Bayes Network (BN), Naive Bayes (NB), REPTree, and J48. This study used the CICIDS-2017 dataset because it has been tested and is frequently used in IDS research. Ensemble-3 is superior to other approaches and previous studies by evaluating its performance.

Keywords— *Anomaly detection, CICIDS-2017, Feature selection, Chi-square, The ensemble method*

1. INTRODUCTION

The transition from referring to the Internet as a general concept to the more specific term "Internet of Things" has significant implications for the network's continued growth of data

Received Juni 16th,2023; Revised July 28th, 2023; Accepted January 9th, 2024

traffic. In IoT various physical and electronic devices, sensors and other objects can communicate with each other without human intervention. Various objects in the IoT produce, send, and receive data across the network. IoT creates a very high data flow. Attackers can exploit IoT traffic to threaten user privacy [1]. Many academic studies have been conducted to establish a wide range of approaches and methods for dealing with the IoT and its associated security issues. Researchers frequently use machine learning and data mining techniques to identify attacks [2]. Although many attacks and mitigation approaches have been discussed, many attacks still need to be evaluated, and research is still needed to curate these attacks [3].

Conversely, each intelligent device and sensor within the IoT network produces a significant volume of data traffic characterized by a large data dimension. In the field of anomaly detection and intrusion detection systems (IDS), the problem of data dimensionality poses a significant barrier. The feature selection approach is a commonly employed technique for handling and evaluating extensive datasets [4]. In order to reduce the number of features required for the detection procedure, an algorithm is used to pick the features to be used. Feature selection techniques have been found to reduce computing burden while maintaining high detection accuracy significantly [5]. To get around IDS's data dimensionality, numerous feature selection methods have been studied and developed. IDS had to deal with unbalanced data as well. In [6] research, it was explained that in real-world traffic, the distribution of normal traffic and attack traffic has significant differences (unbalanced). Therefore, IDS must be tested with data that represents real network characteristics. Unbalanced data gave an impact that machine detection did not work properly, so the output will be a biased prediction. In the case of IDS, data imbalance can result in attacks not being detected or the possibility of attack traffic being detected as regular traffic (misclassified) [7]. The application of the ensemble method is one approach used to overcome unbalanced problems. For instance, research by [8] suggested information gain (IG) and principal component analysis (PCA) as feature selection techniques. It used support vector machine (SVM), instance-based learning methods (IBK), and multilayer perceptron (MLP) as ensemble classifiers. In research [9], CFS-BA, a dimension reduction technique, is combined with the Random Forest (RF), Forest by Penalizing Attributes (Forest PA), and C4.5 algorithms as an ensemble approach. In an alternative investigation, Spearman's rank correlation coefficient was employed for feature selection, while logistic regression and the Decision Tree (DT) technique were utilized for detection [10]. In the meantime, the Best First search algorithm and the One R algorithm for anomaly detection were utilized in research [11] to select relevant features. These studies' findings indicate that anomaly detection performance can still be improved.

This work suggests combining feature selection methods and classification algorithms to detect attacks on the IoT effectively. It accomplishes this by taking into account the capabilities of ensemble methods and feature selection techniques that previous studies have suggested. This study proposes feature selection with Chi-Square for dimensional reduction in unbalanced data. This technique will generate relevant features to be analyzed using a classification algorithm. The features resulting from the chi-square technique are expected to be used to identify attacks on the network. According to the research [12], chi-square can have relevant features, thereby increasing the performance of attack detection.

As mentioned in [13], more than one classification algorithm will be needed to deal with unbalanced data. Machine learning techniques include the ensemble method. This method combines several different classification techniques [11]. The survey's findings demonstrate that this approach can boost classification performance. Cases involving intrusion detection have frequently been resolved using the Ensemble method.

Based on the ensemble method, the anomaly detection system proposed in this study can handle unbalanced data following previous research. Combining classification algorithms such as BN, NB, REP Tree, and J48 is an ensemble detection method. This research implemented the Chi-Square technique and ensemble method on the CICIDS-2017 dataset for detecting Benign traffic, Infiltration, DDoS, Web Attack XSS, Port Scan, Web Attack Brute Force, Bot, DoS Slowloris, and Web Attack SQL Injection. The CICIDS-2017 was chosen as the dataset because this dataset is reliable [14], and has been widely used in IDS research. This research contributes to producing a reliable detection method that can detect attacks on high-dimensional and unbalanced data.

This study aims to develop an ensemble method to detect attacks on the IoT. For this reason, several steps were taken, first by implementing chi-square as a feature selection to deal with high-dimensional data problems. Second, to test and evaluate how well classification algorithms like J48, NB, REP Tree, and BN detect attacks on unbalanced data. Thirdly, to detect attacks and evaluate the proposed ensemble method's performance, make a proposal for an approach that combines a number of classification algorithms. Finally, a comparison of the proposed method's performance to that of previous studies was carried out in order to assess the method's dependability.

2. METHODS

This study proposes a reliable detection method for high-dimensional and unbalanced data. This research was conducted through 5 stages, as presented in Figure 1. These steps include Step 1, data preparation; Step 2, feature selection; Step 3, anomaly detection; Step 4, creating ensemble methods; and Step 5, comparing detection performance. Each step is explained in the next section.

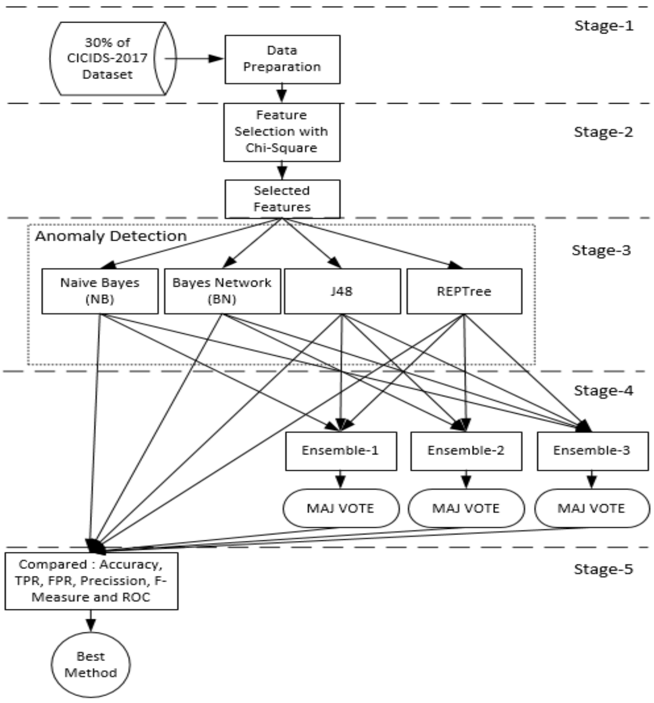

Figure 1 Research Framework

2.1 Data Preparation

Data preparation is done to eliminate unused features, redundant features and Overcoming missing values. The data used in this study is the CICIDS2017 from the ISCX UNB dataset UNB [15]. This dataset is used because it accurately depicts the complexity of actual network traffic. CICIDS 2017 consists of 2,830,743 data records collected over six (six) days during eight (eight) observation sessions in various scenarios. In addition, normal data (benign) and attack data (attack) have been added to this data. The CICIDS2017 dataset contains a variety of attack types, including Heartbleed, Brute Force, DoS, DDOS, Web attack, Infiltration, Bot, and Port Scan. This research only uses 30% of the data from the CICIDS-2017 dataset version of Machine Learning CSV. Thus the total data used is 849,223 records. The reason for using only 30% of the CICIDS-2017 dataset is due to the limited computational resources that researchers use. This test used a 2.70 GHz Intel Core i7 processor, 8 GB of RAM, and the Windows 10 operating system for testing purposes. The software for the analysis tool was WEKA 3.8.5, and the configuration for the heap size was 3072 MB. Nevertheless, the portion of the data that was utilized represented the requirements of the experiment.

2.2 Feature Selection

By eliminating irrelevant features, feature selection reduces data dimensions. It has been demonstrated that feature selection is an effective and efficient method for preparing high-dimensional data for various machine-learning problems [16]. High-dimensional data can have an impact on the computation of machine learning algorithms. Selecting features within datasets with many dimensions is an excellent way to eliminate redundant information and unnecessary details [17]. Feature selection is used to solve the “Curse of dimensionality” problem, by eliminating irrelevant features, thereby reducing the computing time of the detection system.

The feature selection approaches employed in this study were the attribute evaluator chi-square and the ranker-based search method. The feature selection approach employed in this study is the Chi-square method. The chi-square test is employed to eliminate variables that are deemed insignificant in the statistical model. This method measures the weight of dependency between features and classes [22]. The Chi-square is calculated by applying equation 1.

$$x^2(f,c) = \frac{N*(WZ-XY)^2}{(W+x)+(W+Z)+(W+X)+(Y+Z)} \qquad (1)$$

The feature 't' and label of class 'c' frequency of recurrence in the dataset is represented by W. The frequency of occurrence of "t" in the absence of "c" is denoted by the symbol "X," whereas the frequency of "c" in the absence of "t" is denoted by the symbol "Y." The letter Z denotes the frequency of occurrence of any other entity except 'c' or 't.' Last but not least, N denotes the total number of entries in the data set. This study uses the Chi-Square method to select features from the CICIDS 2017 dataset. The feature selection process with Chi-Square is shown in the following pseudocode.

Pseudecode Chi-Square

```
#get information dataset
matrix ← get_information()
#normalize the matrix
normal ← normal_matrix(matrix)
#compute weigth
sub_of_weight ← subjective_of_weight
ob_of_weight ← objective_of_weight(normal)
weight ← combine_weight(sub_of_weight. ob_of_weight)
#compute the weighted_normal_matrix
normal ← normal * weight
#find ideal value of weighted_normal_matrix
ideal ← fin_ideal(wnormal)
#compute chi-square distance between weighted_normal_matrix and
ideal
chisqr_dist ← chisqr_distance(wnormal. ideal)
#rank the chi-square distance
score ← rank(chisqr_dist. ascending)
```

2.3 Anomaly Detection

At this stage, anomaly detection (attack) testing is carried out using a classification algorithm. Based on the study about IDS, the researchers used Machine Learning to detect data traffic attacks. Some studies applied NB, BN, J48, and REPTree as classifier algorithms for anomaly detection. The following is a brief review of some of the classification algorithms used in this study:

- The Naive Bayes (NB) method uses the Bayes theorem to calculate the likelihood that a given data point will be classified into a particular group. The idea that the importance of each trait has no bearing on the class is referred to as "naive." Due to its simplicity and effectiveness, this method has found widespread application in various settings [13].
- The Bayes Network (BN) is a probabilistic graphical model employed to depict the interrelationships among variables of significance. This method's accuracy depends on a few presumptions regarding the target system model's fundamental behavior. The detection accuracy can be decreased if these assumptions are corrected [20].
- The "Reduced Error" concept refers to the reduction or minimization of errors in a given context or system. It involves the identification and implementation of strategies or techniques. The Pruning Tree algorithm, also known as REPTree, is a DT technique that utilizes the principles of a regression tree and iteratively constructs numerous trees. The algorithm chooses the tree in the set deemed most reflective of the data and selects it. The size employed for tree trimming corresponds to the mean squared error of the predictions generated by the tree[21].
- One popular machine learning algorithm, J48 or C4.5, is usually included in decision tree algorithms. Using the idea of entropy, it makes a DT from a training dataset[22]. One notable difference between this algorithm and IDE3 is how the DT is constructed, as J48 or C4.5 can process both continuous and categorical attributes.

2.4 Creating Ensemble Method

The Ensemble method proposed in this research is to use the majority voting technique, which consists of a combination of several classification algorithms. During the training process. the input data will be processed by each algorithm used. At the end of the process. a vote will be taken on the classification results. Of course. The results of the best classification will be shown in the output. This study employed and evaluated the NB algorithm, BN, J48, and REPTree as traffic classification techniques. In this experiment, Three ensembles were proposed, which were named Ensemble-1. Ensemble-2. and Ensemble-3. The following is its configuration:

- Ensemble-1 (E1): using Majority Vote with Naïve Bayes, J48, and REPTree algorithms.
- Ensemble-2 (E2): using Majority Vote with BN, J48, and REPTree algorithms.
- Ensemble-3 (E3): using Majority Vote with Naïve Bayes, BN, J48, and REPTree algorithms.

The training set mode has been used as an ensemble method test. This means all the data input will be analyzed for details. See the following pseudocode.

Pseudocode Ensemble method

```
df ← pd.read_csv("train_data_CICIDS2017.csv")
target ← df["target"]
train ← df.drop("target")
X_train. X_test. y_train. y_test = train_test_split(
    train. target. test_size=0.30)
model_J45 ← RandomForestRegressor()
model_REPtree ← DecisionTreeRegressor()
model_BN ← Bayesnet()
model_NB ← GaussianNB()
all_models ← [model_J45. model_REPtree. model_BN. model_NB]
```

```
s_train. s_test ← stacking(all_models. X_train. X_test.
                           y_train. regression=True. n_folds=4)
final_model ← model_1
final_model ← final_model.fit(s_train. y_train)
pred_final ← final_model.predict(X_test)
print(mean_squared_error(y_test. pred_final))
```

2.6 Comparing Detection Performance

This study will evaluate the efficacy of the suggested ensemble method for anomaly detection in the IoT context. The confusion matrix is a fundamental tool for evaluating IDS research performance. Referring to the definitions generated by the confusion matrix. IDS performance can be measured by : True Positive Rate (TPR), False Positive Rate (FPR), Precision, F-Measure or F1-Score, (Accuracy).

3. RESULTS AND DISCUSSION

This part is dedicated to presenting the outcomes derived from the executed experiments. The discussion revolves around the results of feature selection and the evaluation of the performance of machine learning algorithms. The topic of interest pertains to the evaluation of performance in ensemble methods.

3.1 Feature Selection Result

In this section, the results of feature selection testing are presented. The CICIDS-2017 dataset has 79 data traffic features on the network. Not all of these features are used to recognize attacks. Apart from reducing data dimensions, feature selection is carried out to select relevant features. As elucidated in the preceding session, the chi-square approach was employed for determining feature choice in this study. The chi-square test is employed to categorize traffic characteristics and identify elements that exhibit statistical significance to both benign and attack traffic. Table 1 displays the chi-square-selected features.

Table 1 The selected features using Chi-square Techniques

Feature Names
Packet_length_std, Total_length_of_bwd_packets, Subflow_bwd_bytes, Packet_length_variance, Max_packet_length, Bwd_packet_length_std, Total_length_of_fwd_packets, Subflow_fwd_bytes, Bwd_packet_length_max, Average_packet_size, Fwd_packet_length_max, Avg Bwd Segment Size, Bwd Packet Length Mean, Init_Win_bytes_backward, Packet_length_mean, Flow_IAT_max, Fwd_packet_length_std, Fwd_IAT_max, Flow_duration, Destination_port, Avg_fwd_segment_size, Fwd_packet_length_mean, Init_win_bytes_forward

By applying the chi-square feature selection technique, from 79 features, 23 features were selected. Selected features are presented in Table 3. These selected features will then be used to detect anomalies using the ensemble method. These selected features will then be used to detect anomalies using the ensemble method.

3.2 The Ensemble-1 performance (Majority Vote (NB+J48+REPTree))

Figure 2 presents the result of the Ensemble-1 test. Ensemble-1 can detect Benign traffic, DoS_GoldenEye, PortScan, and DDoS based on TPR, Precision, and F-Measure values. BoT, HeartBleed, FTP_Patator, SSH_Patator, Infiltration, DoS_Slowloris, DoS_httptest, DoS_Hulk, and Web_Attack_BruteForce. Only a few attacks have not been fully detected, namely the Web_Attack_Sql_Injection and Web_Attack_Xss attacks.

3.3 The Ensemble-2 Performance (Majority Vote (BN+J48+REPTree))

Figure 3 presents the result of the ensemble-2 test. Based on TPR, Precision and F-Measure values, Ensemble-2 is able to detect Benign traffic, DoS_GoldenEye, PortScan, DDoS, Bot, HeartBleed, FTP_Patator, SSH_Patator, Infiltration, DoS_Slowloris, DoS_httptest, DoS_Hulk, and Web_Attack_BruteForce. The test results also show the increased precision value of the Web_Attack_Sql_Injection and Web_Attack-XSS attacks.

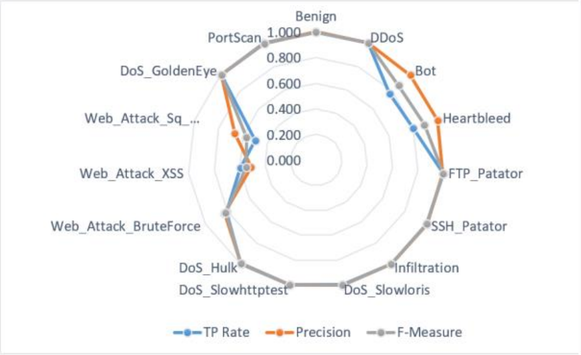

Figure 2 The Performance of Ensemble-1 (Majority Vote (NB+J48+REPTree))

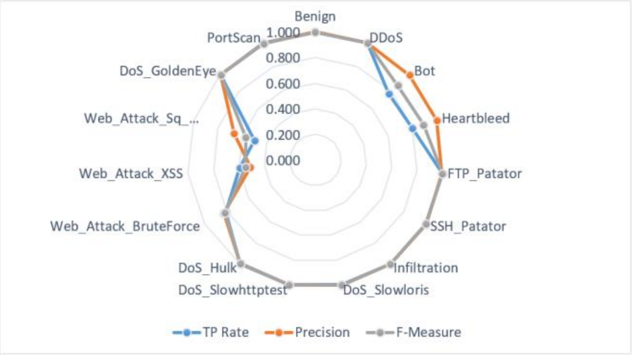

Figure 3 The Performance of Ensemble-2 (Majority Vote (BN+J48+REPTree))

3.4 The Ensemble-3 Performance (Majority Vote (NB+BN+J48+REPTree))

Figure 4 presents the result of the ensemble-3 test. Based on TPR, Precision, and F-Measure values, Ensemble-1 can detect Benign traffic, DoS_GoldenEye, PortScan, DDoS, and BoT. HeartBleed. FTP_Patator. SSH_Patator. Infiltration. DoS_Slowloris. DoS_httptest. DoS_Hulk, Web_Attack_XSS and Web_Attack_BruteForce. The test results also show increased TPR, Precision, and F-measure values in the Web_Attack_Sql_Injection attack. Based on the performance of each ensemble method. the performance of Ensemble-3 is better than Ensemble-1 and Ensemble-2.

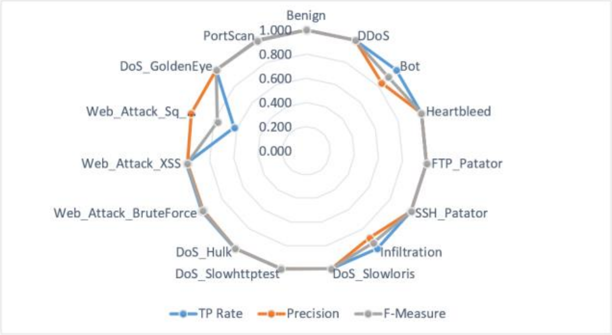

Figure 4 The Performance of Ensemble-3 (Majority Vote (NB+BN+J48+REPTree))

3.5 Performance Comparison

In addition, this study compared the proposed method's performance to that of more recent or traditional approaches. This study employs NB, BN, J48, and REPTree as classification methods. The proposed Ensemble method's performance is also compared to these methods. The objective of this comparative analysis is to assess the reliability of the suggested methodology. The TPR, FPR, Precision and the F-Measure values are employed to conduct comparisons.

The TPR values for each method are shown in Table 2. The TPR values for each classification method are shown in Table 4. The performance of each classification method in detecting attacks on the CICIDS 2017 dataset is shown by this TPR value. It can be deduced from these TPR values that the Ensemble-3 approach is superior to other approaches when detecting attack traffic. With a TPR value greater than 0.970, almost all types of traffic can be correctly identified, according to the TPR ensemble-3 value, except for traffic caused by Web

Attack Sql Injection, which has a TPR value of 0.625. However, this value is superior to other approaches.

Table 2 Comparison of TPR Values

Class	Classifier						
	NB	BN	J48	REPT.	E1	E2	E3
Benign	0.343	0.897	0.999	0.999	0.999	0.999	0.999
Bot	0.352	1.000	0.735	0.801	0.774	0.824	1.000
DDoS	0.782	0.998	0.999	0.999	0.999	0.999	1.000
Infiltration	1.000	0.875	0.625	1.000	1.000	0.875	1.000
PortScan	0.991	0.995	1.000	0.999	1.000	0.999	1.000
Web-Attack-SQL-Injection	0.500	0.500	0.500	0.000	0.500	0.500	0.625
Web-Attack-XSS	0.955	0.045	0.406	0.282	0.594	0.252	0.995
Web-Attack-Brute-Force	0.004	0.996	0.899	0.870	0.807	0.945	0.996
DoS-Hulk	0.597	0.990	1.000	0.999	1.000	1.000	1.000
DoS-Slowhttptest	0.189	0.990	0.993	0.994	0.994	0.996	0.998
DoS-GoldenEye	0.700	0.807	0.998	0.993	0.994	0.994	0.999
DoS-Slowloris	0.525	0.994	0.994	0.994	0.995	0.995	0.998
Heartbleed	1.000	1.000	0.800	0.800	0.800	0.800	1.000
FTP-Patator	0.998	0.998	1.000	0.997	1.000	1.000	1.000
SSH-Patator	0.515	0.998	0.999	0.996	0.997	0.999	0.999
Average	0.630	0.872	0.863	0.848	0.897	0.878	0.974

Description: E1=Vote(NB+J48+REPTree). E2=Vote(BN+J48+REPTree). E3=Vote(NB+BN+J48+REPTree)

In Table 3, the FPR values for each method are presented. The lowest average value of FPR is 0.000. Furthermore, the highest is 0.298. So, almost all algorithms have good FPR values.

Table 3 Comparison of FPR Values

Class	Classifier						
	NB	BN	J48	REPT.	E1	E2	E3
Benign	0.001	0.001	0.001	0.002	0.001	0.001	0.000
Bot	0.298	0.026	0.000	0.000	0.000	0.000	0.000
DDoS	0.002	0.000	0.000	0.000	0.000	0.000	0.000
Infiltration	0.011	0.002	0.000	0.000	0.000	0.000	0.000
PortScan	0.225	0.001	0.000	0.000	0.000	0.000	0.000
Web-Attack-SQL-Injection	0.000	0.001	0.000	0.000	0.000	0.000	0.000
Web-Attack-XSS	0.004	0.002	0.000	0.000	0.000	0.000	0.000
Web-Attack-Brute-Force	0.004	0.042	0.000	0.000	0.000	0.000	0.000
DoS-Hulk	0.007	0.000	0.000	0.000	0.000	0.000	0.000
DoS-Slowhttptest	0.011	0.009	0.000	0.000	0.000	0.000	0.000
DoS-GoldenEye	0.020	0.000	0.000	0.000	0.000	0.000	0.000
DoS-Slowloris	0.006	0.000	0.000	0.000	0.000	0.000	0.000
Heartbleed	0.000	0.001	0.000	0.000	0.000	0.000	0.000
FTP-Patator	0.001	0.000	0.000	0.000	0.000	0.000	0.000
SSH-Patator	0.002	0.000	0.000	0.000	0.000	0.000	0.000
Average	0.039	0.006	0.000	0.000	0.000	0.000	0.000

Description: E1=Vote(NB+J48+REPTree). E2=Vote(BN+J48+REPTree). E3=Vote(NB+BN+J48+REPTree)

The precision values for every classification algorithm are presented in Table 4. Based on the test findings, it can be observed that Ensemble-1 displays an average precision score of 0.931, while Ensemble-2 offers an average precision score of 0.956. Additionally, Ensemble-3 highlights an average precision score of 0.979. On the other hand, it can be observed that the J48 classifier demonstrates an average precision score of 0.961, which exceeds the precision scores of ensembles 1 and 2. Upon careful examination of the different methods, it becomes apparent that ensemble-3 exhibits the highest average precision value.

Table 4 Comparison of Precision Values

Class	Classifier						
	NB	BN	J48	REPT.	E1	E2	E3
Benign	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Bot	0.001	0.025	0.981	0.952	0.991	0.946	0.839
DDoS	0.954	0.997	1.000	1.000	1.000	1.000	1.000
Infiltration	0.001	0.004	1.000	1.000	1.000	1.000	0.889
PortScan	0.207	0.988	0.994	0.994	0.994	0.994	0.994
Web-Attack-SQL-Injection	0.013	0.006	1.000	NaN	0.667	1.000	1.000
Web-Attack-XSS	0.050	0.005	0.678	0.633	0.511	0.718	0.985
Web-Attack-Brute-Force	0.001	0.013	0.773	0.720	0.827	0.697	0.983
DoS-Hulk	0.889	0.995	0.996	0.997	0.996	0.996	0.998
DoS-Slowhttptest	0.032	0.175	0.995	0.988	0.993	0.991	0.996
DoS-GoldenEye	0.115	0.998	0.995	0.996	0.997	0.998	0.999
DoS-Slowloris	0.154	0.985	0.999	0.998	0.998	0.999	0.998
Heartbleed	0.556	0.005	1.000	1.000	1.000	1.000	1.000
FTP-Patator	0.827	0.949	1.000	0.996	0.997	1.000	1.000
SSH-Patator	0.398	0.820	0.999	0.997	1.000	0.999	1.000
Average	0.347	0.531	0.961	0.948	0.931	0.956	0.979

Description : E1=Vote(NB+J48+REPTree). E2=Vote(BN+J48+REPTree). E3=Vote(NB+BN+J48+REPTree)

In Table 5. The F-Measure values are presented as the output of each classification algorithm. Based on the F-measure value for each traffic class, Ensemble-3 has a better F-measure value when compared to other methods.

Table 5 Comparison of F-Measure Values

Class	Classifier						
	NB	BN	J48	REPT.	E1	E2	E3
Benign	0.510	0.946	0.999	0.999	0.999	0.999	1.000
Bot	0.002	0.049	0.841	0.870	0.869	0.881	0.913
DDoS	0.860	0.997	1.000	0.999	1.000	0.999	1.000
Infiltration	0.002	0.007	0.769	1.000	1.000	0.933	0.941
PortScan	0.343	0.991	0.997	0.997	0.997	0.997	0.997
Web-Attack-SQL-Injection	0.025	0.011	0.667	?	0.571	0.667	0.769
Web-Attack-XSS	0.094	0.008	0.508	0.390	0.549	0.374	0.990
Web-Attack-Brute-Force	0.001	0.025	0.831	0.788	0.816	0.802	0.989
DoS-Slowloris	0.238	0.989	0.997	0.996	0.997	0.997	0.998
DoS-Slowhttptest	0.055	0.297	0.994	0.991	0.994	0.994	0.997
DoS-Hulk	0.715	0.993	0.998	0.998	0.998	0.998	0.999

DoS-GoldenEye	0.198	0.892	0.996	0.994	0.996	0.996	0.999
Heartbleed	0.714	0.010	0.889	0.889	0.889	0.889	1.000
FTP-Patator	0.904	0.973	1.000	0.997	0.998	1.000	1.000
SSH-Patator	0.449	0.900	0.999	0.997	0.998	0.999	1.000

Description : E1=Vote(NB+J48+REPTree). E2=Vote(BN+J48+REPTree). E3=Vote(NB+BN+J48+REPTree)

3.6 Accuracy

The accuracy testing for anomaly detection in this study, which involved several classification algorithms and ensemble approaches, is presented in Figure 5. Based on the accuracy value, The findings suggest that the ensemble technique demonstrates superior performance in accuracy compared to the NB, BN, J48, and REPTree algorithms. The graph presented in Figure 4 shows an accuracy value of 99.88% achieved by the J48, E1, and E2 (E2) algorithms. This accuracy value is very good when compared with NB, Network Bayes, J48, and the REPTree algorithm. Nevertheless, the accuracy of E3 stands at an impressive 99.93%, surpassing the accuracy values of NB, BN, J48, RepTree, E1, and E2. Therefore, the suggested approach, mainly E3, exhibits enhanced performance in anomaly identification.

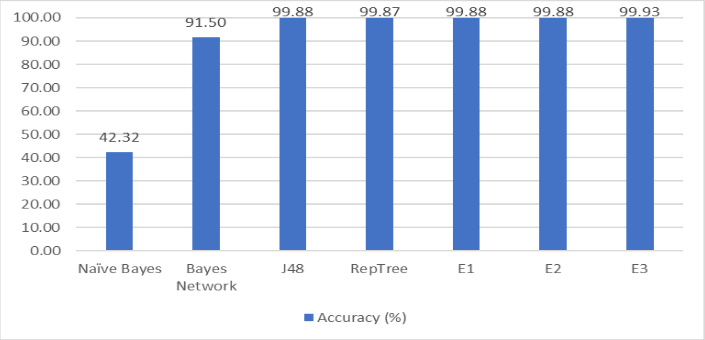

Figure 5. Accuracy of the Proposed Method

A confusion matrix is utilized, as was previously mentioned, to evaluate the detection method's efficacy. The TPR, FPR, Precision, and F-Measure and Accuracy values are determined using the confusion matrix. Ensemble-3 outperforms the other methods used in this study in terms of performance based on the test results for each method, taking into account the TPR, FPR, F-Measure, and Accuracy values.

3.7 Comparing with previous work

In order to establish the reliability of the proposed methodology, a comparison with the previous study is done. Table 6 provides a comparative analysis of the ensemble method's efficacy, as prior scholars suggested, concerning accuracy metrics. In contrast to prior studies, the proposed methodology exhibits a higher level of efficacy.

Table 6. Comparison with previous research

Authors	Feature Selection	Detection Method	Accuracy (%)
[8]	Principle Component Analysis (PCA) and Information Gain (IG)	a combination of SVM, IBK and MPL	98.95
[9]	Correlation-based feature selection combined with Bat Algorithm	Voting (C4.5, RF, ForestPA)	99.89
[10]	Spearman's rank correlation coefficient	logistic regression and a DT	98.80
[11]	Best First search algorithm	Jrip, PART, and OneR	82.97
[23]	Not applied	SVM and ExtraTree	99.90
Proposed method	Chi-Square	Majority Vote with NB, BN, J48, and REPTree algorithms	99.93

4. CONCLUSIONS

This research aims to improve attack detection on IoT networks characterized by large data traffic volumes. The chi-square feature is used as a selection approach to overcome the challenges of solving high-dimensional data. By providing recommendations regarding essential features and their relevance through weight ranking, the dimensionality of the data can be reduced. In this study, a total of 23 features were selected through feature selection to differentiate between normal network traffic and malicious behavior effectively. An ensemble method is proposed to improve attack detection on high-dimensional data with imbalanced data. The proposed ensemble method combines state-of-the-art classification algorithms, namely Bayesian Network, Naïve Bayes, J48, and REPTree. The proposed ensemble methods, especially ensemble-3, show performance that outperforms other classification algorithms used in this study. The comparison results with previous research show that the accuracy value of the proposed method is superior.

Although this research has produced an ensemble method with outstanding performance, several weaknesses must be corrected in future research. This research uses the WEKA tool to test, utilizing the Use Training Set mode. In the future, it is necessary to test with various test modes, such as Fold and Split Data cross-validation. This research can still be developed to optimize the detection of Web Attack SQL Injection attacks, Web Attack XSS, and Bot attacks by using more effective feature selection techniques and other classification algorithms. Because this is initial research, this research only uses 30% of the CICIDS2017 dataset. Future research will be tested with 100% CICIDS2017 dataset and will test the method using the relevant and latest IDS dataset.

REFERENCES

[1] H. Mliki, A. Kaceam, and L. Chaari, “A Comprehensive Survey on Intrusion Detection based Machine Learning for IoT Networks,” *ICST Transactions on Security and Safety*, vol. 8, no. 29, p. 171246, Nov. 2021, doi: 10.4108/eai.6-10-2021.171246.

[2] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, “A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security,” *IEEE Commun. Surv. Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020, doi: 10.1109/COMST.2020.2988293.

[3] S. Choudhary and N. Kesswani, “A Survey: Intrusion Detection Techniques for Internet of Things,” *International Journal of Information Security and Privacy*, vol. 13, no. 1, pp. 86–105, Jan. 2019, doi: 10.4018/IJISP.2019010107.

[4] S. Maza and M. Touahria, “Feature selection for intrusion detection using new multi-objective estimation of distribution algorithms,” *Appl Intell*, vol. 49, no. 12, pp. 4237–4257, Dec. 2019, doi: 10.1007/s10489-019-01503-7.

[5] R. A. Ghazy, E.-S. M. El-Rabaie, M. I. Dessouky, N. A. El-Fishawy, and F. E. A. El-Samie, “Feature Selection Ranking and Subset-Based Techniques with Different Classifiers for Intrusion Detection,” *Wireless Pers Commun*, vol. 111, no. 1, pp. 375–393, Mar. 2020, doi: 10.1007/s11277-019-06864-3.

[6] P. Bedi, N. Gupta, and V. Jindal, “I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems,” *Appl Intell*, vol. 51, no. 2, pp. 1133–1151, Feb. 2021, doi: 10.1007/s10489-020-01886-y.

[7] N. Gupta, V. Jindal, and P. Bedi, “LIO-IDS: Handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system,” *Computer Networks*, vol. 192, p. 108076, Jun. 2021, doi: 10.1016/j.comnet.2021.108076.

[8] F. Salo, A. B. Nassif, and A. Essex, “Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection,” *Computer Networks*, vol. 148, pp. 164–175, Jan. 2019, doi: 10.1016/j.comnet.2018.11.010.

[9] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, “Building an efficient intrusion detection system based on feature selection and ensemble classifier,” *Computer Networks*, vol. 174, p. 107247, Jun. 2020, doi: 10.1016/j.comnet.2020.107247.

[10] Q. R. S. Fitni and K. Ramli, “Implementation of Ensemble Learning and Feature Selection for Performance Improvements in Anomaly-Based Intrusion Detection Systems,” in *2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, Bali, Indonesia: IEEE, Jul. 2020, pp. 118–124. doi: 10.1109/IAICT50021.2020.9172014.

[11] D. P. Gaikwad, “Intrusion Detection System Using Ensemble of Rule Learners and First Search Algorithm as Feature Selectors,” *IJCNIS*, vol. 13, no. 4, pp. 26–34, Aug. 2021, doi: 10.5815/ijcnis.2021.04.03.

[12] I. S. Thaseen, Ch. A. Kumar, and A. Ahmad, “Integrated Intrusion Detection Model Using Chi-Square Feature Selection and Ensemble of Classifiers,” *Arab J Sci Eng*, vol. 44, no. 4, pp. 3357–3368, Apr. 2019, doi: 10.1007/s13369-018-3507-5.

[13] A. Abbas, M. A. Khan, S. Latif, M. Ajaz, A. A. Shah, and J. Ahmad, “A New Ensemble-Based Intrusion Detection System for Internet of Things,” *Arab J Sci Eng*, vol. 47, no. 2, pp. 1805–1819, Feb. 2022, doi: 10.1007/s13369-021-06086-5.

[14] I. Sharafaldin *et al.*, “Towards a Reliable Intrusion Detection Benchmark Dataset,” *JSN*, vol. 2017, no. 1, pp. 177–200, 2017, doi: 10.13052/jsn2445-9739.2017.009.

[15] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization:,” in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Funchal, Madeira, Portugal: SCITEPRESS - Science and Technology Publications, 2018, pp. 108–116. doi: 10.5220/0006639801080116.

[16] J. Li *et al.*, “Feature Selection: A Data Perspective,” *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–45, Nov. 2018, doi: 10.1145/3136625.

[17] J. Cai, J. Luo, S. Wang, and S. Yang, “Feature selection in machine learning: A new perspective,” *Neurocomputing*, vol. 300, pp. 70–79, Jul. 2018, doi: 10.1016/j.neucom.2017.11.077.

[18] I. S. Thaseen and Ch. A. Kumar, “Intrusion Detection Model Using Chi Square Feature Selection and Modified Naïve Bayes Classifier,” in *Proceedings of the 3rd International Symposium on Big Data and Cloud Computing Challenges (ISBCC – 16’)*, vol. 49, V. Vijayakumar and V. Neelanarayanan, Eds., in Smart Innovation, Systems and Technologies, vol. 49. , Cham: Springer International Publishing, 2016, pp. 81–91. doi: 10.1007/978-3-319-30348-2_7.

[19] I. Sumaiya Thaseen and C. Aswani Kumar, “Intrusion detection model using fusion of chi-square feature selection and multi class SVM,” *Journal of King Saud University - Computer and Information Sciences*, vol. 29, no. 4, pp. 462–472, Oct. 2017, doi: 10.1016/j.jksuci.2015.12.004.

[20] M. Reazul, A. Rahman, and T. Samad, “A Network Intrusion Detection Framework based on Bayesian Network using Wrapper Approach,” *IJCA*, vol. 166, no. 4, pp. 13–17, May 2017, doi: 10.5120/ijca2017913992.

[21] T. Ait Tchakoucht and M. Ezziyyani, “Building A Fast Intrusion Detection System For High-Speed-Networks: Probe and DoS Attacks Detection,” *Procedia Computer Science*, vol. 127, pp. 521–530, 2018, doi: 10.1016/j.procs.2018.01.151.

[22] S. Aljawarneh, M. B. Yassein, and M. Aljundi, “An enhanced J48 classification algorithm for the anomaly intrusion detection systems,” *Cluster Comput*, vol. 22, no. S5, pp. 10549–10565, Sep. 2019, doi: 10.1007/s10586-017-1109-8.

[23] N. S. Bhati and M. Khari, “A new ensemble based approach for intrusion detection system using voting,” *IFS*, vol. 42, no. 2, pp. 969–979, Jan. 2022, doi: 10.3233/JIFS-189764.