Received June 30, 2020, accepted July 12, 2020, date of publication July 16, 2020, date of current version July 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009843

CICIDS-2017 Dataset Feature Analysis With Information Gain for Anomaly Detection

KURNIABUDI[1,2], (Member, IEEE), DERIS STIAWAN[3], DARMAWIJOYO[4], MOHD YAZID BIN IDRIS[5], (Member, IEEE), ALWI M. BAMHDI[6], AND RAHMAT BUDIARTO[7]

[1]Faculty of Engineering, Universitas Sriwijaya, Palembang 30128, Indonesia
[2]Faculty of Computer Science, Universitas Dinamika Bangsa, Jambi 36138, Indonesia
[3]Faculty of Computer Science, Universitas Sriwijaya, Palembang 30128, Indonesia
[4]Faculty of Mathematics and Natural Sciences, Universitas Sriwijaya, Palembang 30128, Indonesia
[5]Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia
[6]Department of Computer Sciences, College of Computing, Umm Al-Qura University, Al Qunfudhah 21912, Saudi Arabia
[7]College of Computer Science and IT, Albaha University, Al Aqiq 65779-7738, Saudi Arabia

Corresponding author: Deris Stiawan (deris@unsri.ac.id)

This research was supported partially by the Ministry of Higher Education (MoHE), Malaysia under grant MRUN and Research Management Center (RMC) of Universiti Teknologi Malaysia, vote No: R.J130000.7851.4L872, and Ministry of Education and Culture of the Republic of Indonesia under research grand from Directorate of Research and Society Dedication, Universitas Sriwijaya.

ABSTRACT Feature selection (FS) is one of the important tasks of data preprocessing in data analytics. The data with a large number of features will affect the computational complexity, increase a huge amount of resource usage and time consumption for data analytics. The objective of this study is to analyze relevant and significant features of huge network traffic to be used to improve the accuracy of traffic anomaly detection and to decrease its execution time. Information Gain is the most feature selection technique used in Intrusion Detection System (IDS) research. This study uses Information Gain, ranking and grouping the features according to the minimum weight values to select relevant and significant features, and then implements Random Forest (RF), Bayes Net (BN), Random Tree (RT), Naive Bayes (NB) and J48 classifier algorithms in experiments on CICIDS-2017 dataset. The experiment results show that the number of relevant and significant features yielded by Information Gain affects significantly the improvement of detection accuracy and execution time. Specifically, the Random Forest algorithm has the highest accuracy of 99.86% using the relevant selected features of 22, whereas the J48 classifier algorithm provides an accuracy of 99.87% using 52 relevant selected features with longer execution time.

INDEX TERMS Feature selection, anomaly detection, information gain, CICIDS-2017 dataset, classifier algorithm.

I. INTRODUCTION

The anomaly-based intrusion detection is one of the techniques used to recognize zero-day attacks. Although various anomaly detection techniques have been developed, yet there are challenges and issues in the area, namely high dimensionality of data [1], impact on computational complexity [2], [3], and computational time [4].

One approach used by researchers to deal with the data dimensionality issue is feature selection technique. Feature selection technique eliminates features, helps in understanding data, reduces computing time, reduces ''curse of dimensionality'' effects, and improves predictive machine performance [5]. Feature selection is a part of dimensional reduction, known as a process of selecting an optimal feature subset that represents the entire dataset [6].

Many research works that use feature selection techniques to improve the accuracy of anomaly detection have been carried out such as works in [7]–[11]. Most of the works use the Network Security Laboratory-Knowledge Discovery and Data Mining (NSL-KDD) dataset, a refined version of its predecessor KDD Cup 99 dataset. Methods and measurements have been proposed that show the ability in improving detection accuracy including Chi-Square, Information Gain, Correlation Based with Naive Bayes and Decision Table Majority Classifier [12], Support Vector Machine (SVM) [13] and Random Forest [12]. Nevertheless, those methods were not tested on a large dataset with a large number of features.

The associate editor coordinating the review of this manuscript and approving it for publication was Hong-Mei Zhang.

As mentioned in [14], data with a large number of features can affect the learning model that tends to overfit and will decrease the performance, increasing memory use, and computational cost for analytic. In fact, very rare researchers which consider computational time in their works, especially in anomaly detection.

On the other hand, Information Gain has been widely used by researchers to analyze significant and relevant features. According to works in [15]–[21] the Information Gain is used to reduce dimensionality by selecting more relevant features through feature weight calculation. Eliminating irrelevant features may improve the performance of the detection system. Many research works implement Information Gain on the dataset with limited features to analyze. In this study, the CICIDS-2017 dataset with more complex features is used. The CICIDS-2017 dataset contains a high volume of traffic and a large number of features to be observed for anomalies detection.

Previous works which use the CICIDS-2017 dataset and also use Information Gain feature selection technique do not mention the basis on how to determine the score value used for feature selection. Each researcher uses different score value. In this paper, the authors investigate and analyze the ability of the Information Gain in determining relevant features for network traffic classification, especially for traffic with bigger number of features. The authors distribute the features into groups based on their minimum score values. Then each feature group is used as a filter for the five classifier algorithms; Random Forest, Bayes Network, Random Tree, Naive Bayes and J48 to perform anomaly/attack detection on the dataset. Then, the detection results are compared with the aim is to validate the significance and relevance of the selected feature groups. The more accurate the detection results the more significance and relevant the feature group. Thus, the authors analyze the effect of weighted features resulted from the Information Gain against the anomaly/attack detection performance as well as to find the most significant and relevant features to be used to increase the performance of anomaly/attack detection.

The rest of the paper is organized as follows. Section 2 presents the relevant researches. Section 3 briefly discusses the dataset and experimental setup used in this study. Section 4 explains more details on the experiments and results findings of this study. Finally, Section 5 provides a conclusion and potential future works.

II. RELEVANT RESEARCHES

Research on feature selection has been carried out especially in network attack detection. Wang *et al.* [22] analyze the features of large network traffic, by choosing the most significant features, using a combination of filtered-based and wrapper-based algorithms. The method produces 10 significant features and can increase the detection rate up to 99.8% and false alarm of 0.34%. Ambusaidi *et al.* [23] propose a supervised filtered-based features selection algorithm called Flexible Mutual Information Feature Selection (FMIFS). The algorithm contributes to the Least-squares support-vector machines (LS-SVM) IDS with a better accuracy and lower computational rates than the previous methods.

Authors in [24] propose a feature identification approach by combining filtered-based and wrapper-based methods with clustering method to provide weight for each feature. The proposed method is able to identify features that can improve the accuracy of attack detection. Chen *et al.* [25] introduce a tree-seed algorithm (TSA) that is used to extract effective features. The proposed algorithm reduces the dimension of data, by eliminating redundant features, which in turn improve the accuracy of the K-Nearest Neighbor (KNN) classifier. The work in [10] discusses a Discrete Differential Evolution (DDE) technique and the C4.5 Machine Learning algorithm. The proposed technique produces 16 relevant features with a classification accuracy of 99.92%. While Peng *et al.* [26] combine the Ant-Colony Optimization algorithm and feature selection, called FACO. The proposed work is able to produce features that improve the classification algorithm accuracy. Finally, researchers in [27] propose an IDS called FWP-SVM-GA, based on the genetic algorithm and SVM. The proposed algorithm increases detection rate, accuracy, true positive rate (TPR) and reduces false-positive rate (FPR) and SVM training time.

Having done reviewing previous works, the authors come up with a hypothesis that feature selection can improve the performance of classification algorithms by eliminating non-useful and redundant features. Even a small number of selected features may increase the detection accuracy. Up to now researchers mainly use the KDD CUP 99 dataset that only has 41 features as test data. The use of a large dataset still rare. Therefore, the reliability of the proposed methods have not been tested on larger dimension dataset (with more features and number of records). Table 1 summarizes feature selection research works on intrusion detection field for the last five (5) years.

Yulianto *et al.* [56] combine the Synthetic Minority Oversampling Technique (SMOTE), Principal Component Analysis (PCA), and Ensemble Feature Selection (EFS) to improve the performance of AdaBoost-based IDS on the CICIDS-2017 Dataset. The authors claim that the combined method outperforms the SVM-based method with regards to accuracy, precision, recall and F1 Score.

On the other hand, despite many researchers using Information Gain as a feature selection technique, there are very limited discussions on how to determine the minimum weight or rank score from the Information Gain result. This score determines how much the features are relevant to the class label. Researchers in [18] and in [21] use a score feature above 0.4 and a score above 0.001, respectively. Meanwhile, research work in [28] considers the minimum weight score of 0.8. In contrast, researchers in [29] remove features one by one and apply the classifier algorithm to find the best accuracy. Such work is very time-consuming especially with a large number of features in the dataset.

TABLE 1. Summary of related studies.

Ref. # (Year)	Dataset	# of Feat.	FS Algorithm	Result
[22] (2015)	KDD Cup 99	41	Information Gain (filtered) enhanced with Bayesian Network (wrapper) and C4.5 (wrapper).	Detection rate as high as 99.8% with false positive rate as 0.34%.
[23] (2016)	KDD Cup 99, NSL-KDD, Kyoto 2006+ dataset	41, 41 and 24	Flexible Mutual Information Feature Selection (FMIFS)	Improve the accuracy and lower computational cost compared to the state-of-the-art methods.
[24] (2016)]	KDD Cup 99	41	Multi Measure Multi Weight FS (Filtered-based &Wrapped-based)	Better detection accuracy with reduced detection time
[10] (2017)	NSL-KDD	41	DDE, C4.5 ML algorithm	16 relevant features with 99.92% of classification accuracy
[25] (2018)	KDD Cup 99	41	Tree-Seed+ Algorithm (TSA) with KNN classifier	Removal of redundant features, improve accuracy and efficiency of network intrusion detection
[26] (2018)	KDD Cup 99	41	Ant Colony	Improve classification efficiency with 98%. accuracy
[27] (2018)	KDD Cup 99	41	feature selection, weight, and parameter optimization (FWP) to support Genetic Algorithm and SVM	Increase detection rate, accuracy rate and true positive rate; decrease false positive rate; and reduces SVM training time
[56] (2018)	CICIDS - 2017	78	PCA,SMOTE, (EFS)	Improve IDS performance on CICIDS2017 dataset

III. METHODOLOGY

This section describes the dataset, experimental configuration, feature selection technique, classification algorithms, and experimental tools.

A. DATASET

This study uses MachineLearningCSV data, which is part of the CICIDS-2017 dataset from ISCX Consortium. MachineLearningCSV consists of eight (8) traffic monitoring sessions, each is in the form of a comma separated value (CSV) file. This file contains normal traffic defined as ''Benign'' traffic and anomaly traffic called as ''Attacks'' traffic. The attack traffics are detailed more as in the second column of Table 2. Other than normal traffic and benign traffic, there are 14 types of attacks in this dataset.

In this work, the authors consider complex features that represent sophisticated attacks on modern network based on its traffic attributes. For examples, features that exist in CICIDS-2017 but are not available in NSL-KDD include: *Subflow Fwd Bytes* and *Total Length Fwd Package* which are required to detect Infiltration and Bot attack types. The *Bwd Packet Lenght Std* feature is required to detect the types of DDoS, DoS Hulk, DoE GoldenEye, and Heartbleed attacks. The *Init Win Fwd Bytes* feature is required to detect the types of Web-Attack, SSH-Patator, and FTP-Patator attacks. Whereas the *Min Bwd Package Length* feature and *Fwd Average Package Length* features are required to recognize normal traffic [58].

TABLE 2. CICIDS-2017 dataset summary.

File Name	Type of Traffic	Number of Record
Monday-WorkingHours.pcap_ISCX.csv	Benign	529,918
Tuesday-WorkingHours.pcap_ISCX.csv	Benign	432,074
	SSH-Patator	5,897
	FTP-Patator	7,938
Wednesday-WorkingHours.pcap_ISCX.csv	Benign	440,031
	DoS Hulk	231,073
	DoS GoldenEye	10,293
	DoS Slowloris	5,796
	DoS Slowhttptest	5,499
	Heartbleed	11
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	Benign	168,186
	Web Attack-Brute Force	1,507
	Web Attack-Sql Injection	21
	Web Attack-XSS	652
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	Benign	288,566
	Infiltration	36
Friday-WorkingHours-Morning.pcap_ISCX.csv	Benign	189,067
	Bot	1,966
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	Benign	127,537
	Portscan	158,930
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	Benign	97,718
	DdoS	128,027
Total Instance/ Record		**2,830,743**

CICIDS-2017 has more complex types of attacks as presented in Table 2. The rational of choosing CICIDS-2017 dataset is to have a dataset that represents closely the current real world network traffic in the experiments.

B. EXPERIMENTAL SETUP

In general, there are four stages in the experimental settings shown in Fig. 1, which can be explained as follows.

1) Only 20% of MachineLearningCSV data from the CICIDS-2017 dataset are used in this experiment. Since the dataset has redundant features, it is needed to remove the redundant ones. Then relabeling process is performed. The 20% of MachineLearningCSV data are then split into 70% for training data and 30% for testing data.
2) Feature selection is performed on the training data using Information Gain. Then selected features are grouped according to their weights.

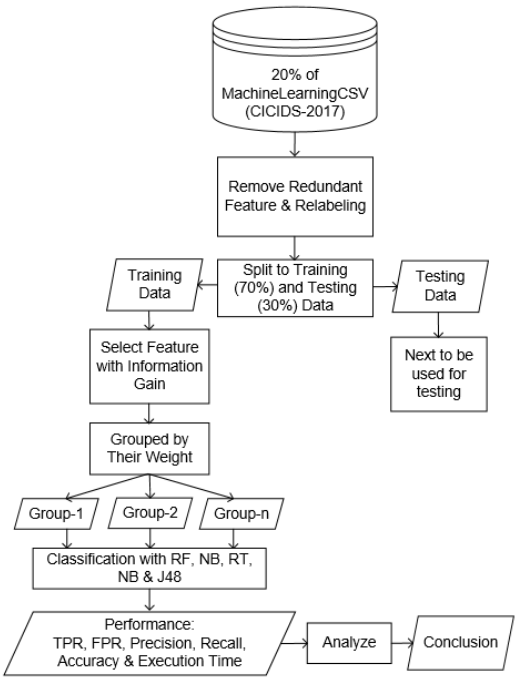


FIGURE 1. Experimental design.

3) Then each feature group or feature subset is classified using Random Forest (RF), Bayes Net (BN), Random Tree (RT), Naive Bayes (NB), and J48 classifiers. The analysis considers the following parameters: TPR, FPR, Precision, Recall, Accuracy, percentage of incorrectly classified, and execution time for the analysis. 10-fold cross-validation is used in this stage.
4) Next, compare and analyze the TPR, FPR, Precision, Recall, Accuracy, percentage of incorrectly classified, and execution time of each classifier algorithm. All learning and testing steps are executed with 10-fold cross-validation. Lastly, draw conclusions.

C. INFORMATION GAIN

Information Gain is the most used feature selection technique. It is a filter-based feature selection [28], [30]. Information Gain uses a simple attribute rank and reduces noise that caused by irrelevant features then detects a feature that have most of information base in specific class. The best feature is determined by calculating feature's entropy. Entropy is a measure of uncertainty that can be used to infer the distribution of features in a concise form [31]. The entropy can be calculated using (1).

$$Entropy\,(S) = \sum_{i}^{c} -P_i log_2 P_i \tag{1}$$

With c is the number of values in the classification class and P_i is the number of samples for class i. After getting the entropy value, the Information Gain value is calculated using (2).

$$Gain\,(S, A) = Entropy\,(s) - \sum_{Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{2}$$

where S is sample, A is an attribute, v is a possible value for attribute A, *Values(A)* are a set of possible values for A. $|S_v|$ is the number of samples for value v. $|S|$ is the number of samples for all data samples and *Entropy* (S_v) is entropy for sample that have a value of v.

This work chooses Information Gain as feature selection since it is a filtered-based technique which provides more stable sets of selected features due to its robust nature against overfitting. Overall, computational complexity of filter-based technique is $O(m \cdot n^2)$, where m is the number of training data, and n is number the of attributes/features. It is less as compared to embedded and wrapper-based techniques [55]. The complex nature of wrapper-based techniques creates the high risk of overfitting. Thus, using feature selection technique that produces significant, relevant, less number of features and less computational complexity will reduce the execution time of classification algorithms used in the anomaly/attack detection process.

The features are given IDs from 1 to 77. The Information Gain ranks the features based on their weight values and the minimum weight is determined manually using try and error approach. In this work, the researchers propose to rank and group the features according to the minimum weight values. Thus, groups of features are obtained and each feature group will be having different number of features as shown later in Table 6. Further, all feature groups will be validated by using the five classifier algorithms, so we can determine which feature groups are effective enough to be used for attacks' types classification.

D. CLASSIFICATION ALGORITHM

The main consideration on parameters for selecting classifier algorithms in this work is good performance in term of accuracy, learning ability, scalability, and speed. Having done some researches on several previous works that support the consideration, five algorithms are considered, they are: Random Forests, Bayesian Network, Random Trees, Naive Bayes and J48 classifiers to be experimented in this work. Research work by Hadi [20] states that random forest trees are strong learners and have good performance in detecting attacks based on the features resulted by Information Gain feature selection. Niranjan *et al.* [39] reveals that the ability of Bayesian Network in classifying attacks outperforms other algorithms. According to Sindhu *et al.* [57], Random Tree is an algorithm that has scalability and efficiency. Naive Bayes is a classification algorithm that is able to identify class labels faster than other algorithms because it has a low complexity of the model [55]. Sahu and Mehtre [15] conclude that J48 algorithm has good accuracy in classifying attacks. Thus, the five classification algorithms are used to validate the significance of the selected features resulted during feature selection stage.

1) RANDOM FOREST (RF)

Random Forest is one of the ensemble classifier methods. If a classifier in an ensemble is a decision tree classifier,

then the collection of classifiers is a ''forest''. Each decision tree is created through a random selection of attributes at each node for separation [32]. The random forest algorithm was proposed by Breich in 2001 [33]. Some anomaly detection studies that use random forest include research conducted by [20], [34] and [35].

2) BAYES NETWORK (BN)

Bayesian Network (BN) is a model that encodes probabilistic relationships between variables of interest. The accuracy of this method depends on assumptions which are usually based on the model behavior of the target system. So any significant deviation from the assumption will cause a decrease in detection accuracy [36]. Some anomaly detection studies that use Bayesian networks include works by Reazul *et al.* [37] and Ding *et al.* [38].

3) RANDOM TREE (RT)

Basically, Random Tree is a decision tree that is built on a collection of random attributes (random). A decision tree is a group of nodes and branches. A node represents a test attribute and branches represent the results. Decision leaves show the final decision taken after calculating all attributes in the form of class labels [39]. Some anomaly detection studies using this method include [40], [41] and [42].

4) NAIVE BAYES (NB)

Bayesian classification is a statistical classification that is able to predict the probability of class membership. Bayesian classification is based on the Bayes theorem [43]. The Bayesian classification is better known as the Naïve Bayes classification. Naïve Bayes assumes that the influence of attribute values on class is independent of other attribute values. Some anomaly detection studies using Naive Bayes include works by Goeschel [44], and Shakya and Sigdel [45].

5) J48

J48 or C4.5 is a widely used machine learning algorithm and is included in the decision tree algorithm. This algorithm builds a decision tree from a set of training data with the entropy concept [43]. It differs from IDE3 in that it builds a decision tree, where J48 or C4.5, can receive continuous and categorical attributes [46]. Some anomaly detection studies using this algorithm include works by Sahu and Mehtre [15] and Muniyandi *et al.* [47].

E. ANALYSIS TOOLS

All simulations in this experiment are executed on a computer with specification of Intel Core i7 processor with 2.70 GHz 8 GB RAM, running Windows 10 as Operating System. For analysis purposes, the Weka 3.9 with heap size of 3072 MB, as machine learning software is used.

IV. EXPERIMENTS, RESULTS AND ANALYSIS

This section presents the data preparation, detail of experimenting with feature selection classification, and lastly, results and discussions of the experimentations.

TABLE 3. Data distribution of labeled attack on 20% Machinelearningcsv data.

New Labels	Old Labels	# of Instances	Fraction to Majority Class	Fraction to Total Instance
Normal	Benign	454,396	100.00	80.25
Bot	Bot	367	0.081	0.06
Brute Force	FTP- Patator, SSH-Patator	2,717	0.598	0.48
Dos/DdoS	DDoS, DoS, GoldenEye, DoS Hulk, DoS Slow, httptest, DoS slowloris, Heartbleed	76,445	16.82	13.50
Infiltration	Infiltration	6	0,001	0.00
Portscan	PortScan	31,882	7.061	5.63
Web Attack	Web Attack– Brute Force, Web Attack– Sql Injection, Web Attack– XSS	426	0.094	0.08
Total Instances		**566,239**		

TABLE 4. The distribution of training & testing data.

New Labels	Instances # of Training Data	Instances # of Testing Data
Normal	318,087	136,219
Bot	265	102
Brute Force	1,904	813
Dos/DdoS	53,427	23,018
Infiltration	5	1
Port Scan	22,324	9,558
Web Attack	292	134
Total instances	**396,304**	**169,845**

A. DATASET PREPARATION

The eight CSV files as listed in Table 2 are combined into one CSV file. Next, to process the dataset using Weka software, this CSV file is converted into the ARFF file. The experiment uses only 20% of MachineLearningCSV data. There are 78 regular features and one class label used in this study. The dataset contains two features or columns named ''Fwd Header Length'' that make it as redundant features, so one of those columns must be removed. Thus, after removing the redundant features, only 77 features are available to be analyzed. As described in the CICIDS-2017 data prone to high-class imbalance will impact low detection accuracy and high false alarm. By adopting solution suggested by Karimi *et al.* [30] and Panigrahi and Borah [48] a new labeling attack traffic is introduced as listed in Table 3. The 77 features are already in numerical data type, so no data transformation is required to feed the data into Weka software.

After relabeling the attack classes, the 20% of MachineLearningCSV data are split into two portions as 70% and 30%. The 70% portion is used for training data and the other 30% portion is used for testing data as tabulated in Table 4. The 70:30 data portion was used in [49]. The experimental

TABLE 5. Feature rank generated by information gain.

No.	Feat. ID	Feature Names	Weight	No.	Feat. ID	Feature Names	Weight
1	41	Packet Length Std	0,638	40	17	Fwd Packet Length Std	0,280
2	13	Total Length of Bwd Packets	0,612	41	29	Bwd IAT Mean	0,271
3	65	Subflow Bwd Bytes	0,612	42	5	Fwd IAT Std	0,268
4	8	Destination Port	0,609	43	15	Fwd Packet Length Min	0,234
5	42	Packet Length Variance	0,577	44	38	Min Packet Length	0,231
6	20	Bwd Packet Length Mean	0,567	45	70	Active Mean	0,231
7	54	Avg Bwd Segment Size	0,567	46	27	Fwd IAT Min	0,229
8	18	Bwd Packet Length Max	0,560	47	73	Active Min	0,228
9	67	Init_Win_bytes_backward	0,554	48	69	min_seg_size_forward	0,227
10	12	Total Length of Fwd Packets	0,546	49	72	Active Max	0,226
11	63	Subflow Fwd Bytes	0,546	50	31	Bwd IAT Min	0,226
12	66	Init_Win_bytes_forward	0,542	51	23	Flow IAT Min	0,216
13	52	Average Packet Size	0,535	52	76	Idle Max	0,205
14	40	Packet Length Mean	0,526	53	74	Idle Mean	0,197
15	39	Max Packet Length	0,512	54	77	Idle Min	0,195
16	14	Fwd Packet Length Max	0,495	55	68	act_data_pkt_fwd	0,186
17	22	Flow IAT Max	0,467	56	6	Bwd IAT Std	0,179
18	36	Bwd Header Length	0,448	57	46	PSH Flag Count	0,106
19	9	Flow Duration	0,443	58	51	Down/Up Ratio	0,088
20	26	Fwd IAT Max	0,438	59	47	ACK Flag Count	0,069
21	55	Fwd Header Length	0,431	60	75	Idle Std	0,036
22	24	Fwd IAT Total	0,415	61	43	FIN Flag Count	0,033
23	25	Fwd IAT Mean	0,390	62	48	URG Flag Count	0,028
24	21	Flow IAT Mean	0,379	63	71	Active Std	0,025
25	2	Flow Bytes/s	0,379	64	44	SYN Flag Count	0,012
26	1	Bwd Packet Length Std	0,360	65	32	Fwd PSH Flags	0,012
27	64	Subflow Bwd Packets	0,355	66	45	RST Flag Count	0
28	11	Total Backward Packets	0,355	67	50	ECE Flag Count	0
29	16	Fwd Packet Length Mean	0,351	68	61	Bwd Avg Bulk Rate	0
30	53	Avg Fwd Segment Size	0,351	69	49	CWE Flag Count	0
31	19	Bwd Packet Length Min	0,324	70	57	Fwd Avg Packets/Bulk	0
32	3	Flow Packets/s	0,311	71	56	Fwd Avg Bytes/Bulk	0
33	37	Fwd Packets/s	0,309	72	34	Fwd URG Flags	0
34	30	Bwd IAT Max	0,306	73	33	Bwd PSH Flags	0
35	7	Bwd Packets/s	0,304	74	35	Bwd URG Flags	0
36	10	Total Fwd Packets	0,291	75	60	Bwd Avg Packets/Bulk	0
37	62	Subflow Fwd Packets	0,291	76	58	Fwd Avg Bulk Rate	0
38	28	Bwd IAT Total	0,287	77	59	Bwd Avg Bytes/Bulk	0
39	4	Flow IAT Std	0,281				

results in [50] shows that the use of the 70:30 portion of training and testing data leads to the same level of accuracy as the portions of 80:20 and 60:40. Meanwhile, experimental result of using 70:30 data portion in other work by Abualkibash [51] results high accuracy. Therefore in this study, the researchers divide the training and testing data with a portion of 70:30. Although the dataset is transformed into a new attack label, the ''Infiltration'' attacks have a very small portion of data compared to other types of attacks. Later, the data will be analyzed by the feature selection technique.

B. FEATURE SELECTION USING INFORMATION GAIN

As mentioned in Section 1, the main issue in a large dataset is dimensionality. Feature selection technique reduces the dimensionality of data by selecting relevant features. The Information Gain evaluates the features by calculating their entropies. In this study, feature selection is implemented by Weka software and the process is shown in algorithm 1.

Table 5 presents the feature rank as the result of feature selection by Information Gain. As mentioned in sub-section 3.C, the feature selection in this experiment uses a filter-based approach. In other words, the feature selection filters

Algorithm 1 Calculate Feature Rank

1: **procedure** Feature_Rank()
2: **Input Fn** = Training dataset, processing 77 features f1,f2,f3... f77
3:**For every feature Fn**
4:**Calculated** Feature Information Weight with Information Gain
5: **Rank** feature with their Weight
6: **Store** Rank, Feature ID, Feature name and feature Weight on Feature_Ranked data

throughout the weight scores, in which features are grouped based on the score of the feature's weight. As listed in Table 6, there are seven groups of features and we called as new features subsets.

C. EXPERIMENTAL RESULT

To analyze the performance of the feature selection performed by Information Gain and the five (5) classifier algorithms, seven (7) measurement metrics are used, they are:

TABLE 6. Selected features by information gain.

Feature Weight	Number of Selected Feature	Selected Features (New Feature Subset)
>0.6	4	41, 13, 65, 8
>0.5	15	41, 13, 65, 8, 42, 20, 54, 18, 67, 12, 63, 66, 52, 40, 39
>0.4	22	41, 13, 65, 8, 42, 20, 54, 18, 67, 12, 63, 66, 52, 40, 39, 14, 22, 36, 9, 26, 55, 24
>0.3	35	41, 13, 65, 8, 42, 20, 54, 18, 67, 12, 63, 66, 52, 40, 39, 14, 22, 36, 9, 26, 55, 24, 25, 21, 2, 1, 64, 11, 16, 53, 19, 3, 37, 30, 7
>0.2	52	41, 13, 65, 8, 42, 20, 54, 18, 67, 12, 63, 66, 52, 40, 39, 14, 22, 36, 9, 26, 55, 24, 25, 21, 2, 1, 64, 11, 16, 53, 19, 3, 37, 30, 7, 10, 62, 28, 4, 17, 29, 5, 15, 38, 70, 27, 73, 69, 72, 31, 23, 76
>0.1	57	41, 13, 65, 8, 42, 20, 54, 18, 67, 12, 63, 66, 52, 40, 39, 14, 22, 36, 9, 26, 55, 24, 25, 21, 2, 1, 64, 11, 16, 53, 19, 3, 37, 30, 7, 10, 62, 28, 4, 17, 29, 5, 15, 38, 70, 27, 73, 69, 72, 31, 23, 76, 74, 77, 68, 6, 46
All	77	*All Feature*

True Positive Rate (TPR), False Positive Rate (FPR), Precision, Recall, Accuracy, percentage of incorrectly classified and execution time. The execution time is measured during the training time (the time measured from the classification process starts until the classification process stops). In the experiment, each feature subset is classified by RT, BN, RT, NB and J48 classifiers. The overall process is shown in Algorithm 2. To evaluate the performance of classification algorithms, this research uses 10-fold cross-validation. The 10-fold cross-validation is used because it reduces computing time while maintains the performance of the classification algorithms in term of accuracy. Hence, the input dataset will be randomly divided into 10 folds with exactly the same size. For each of the 10 fold data, cross-validation will use 9 fold for training and 1 fold for testing. This process is repeated for 10 times until each fold becomes a test fold. This cross-validation method has been widely used in IDS researches, such as in [52], [53], and [54].

Performances of classifiers using four (4) features selected by Information Gain are listed in Table 7. The RF and RT have the highest accuracy of 96.48% compared to other classifiers. Nonetheless, RF has NaN value. NaN is defined as Not a Number or undefined. Compare to the other classifiers, NB is able to detect DoS/DDoS attack up to 0.999 of TPR, however achieves low TPR in detecting Normal and Infiltration traffics. Surprisingly BN has the lowest FPR of 0.010 compared to others. Overall, with these four (4) selected features, the classifiers only can detect DoS/DDoS, PortScan and Brute Force attacks. For Normal traffic only NB suffers for that.

Algorithm 2 Overall Process

1: **procedure** Process()
2:**Input: Fr** = Feature_Ranked data
3:**Output:** Features Subsets, TPR, FPR, Accuracy, Recall, Precision
4: **Reduce**77 features to n features based on a feature weight
5: **For** every feature Fr in Feature_Ranked data
6: **Start to Select feature** with Feature Weight and store on Feature Groups
7: Group1 = all feature with weight >= 0.6
8: Group2 = all feature with weight >= 0.5
9: Group3 = all feature with weight >= 0.4
10: Group4 = all feature with weight >= 0.3
11: Group5 = all feature with weight >= 0.2
12: Group6 = all feature with weight >= 0.1
13: Group7 = all features
14: **For each** Feature groups
15: **Feed** Selected Features to RF, BN, RT, NB, J48 using CICIDS-2017-20%
16: **Apply Classifier**
10: C1 = Random Forest model accuracy
11: C2 = Bayes Network model accuracy
12: C3 = Random Tree model accuracy
13: C4 = Naïve Bayes model accuracy
14: C5 = J48 model accuracy
15: **Calculate** TPR, FPR Accuracy, Recall, Precision
16: **Compare** the Accuracy of C1, C2, C3, C4 and C5

TABLE 7. Performance metric using four features.

Detection	RF	BN	RT	NB	J48
Normal	0.960	0.943	0.960	0.174	0.961
DoS/ DDoS	0.992	0.996	0.992	0.999	0.991
Port Scan	0.995	0.992	0.995	0.983	0.995
Bot	0.438	0.642	0.430	0.687	0.381
Web Attack	0.072	0.031	0.072	0.000	0.072
Infiltration	0.000	0.000	0.400	0.400	0.000
Brute Force	0.792	0.991	0.792	1.000	0.790
Recall	0.965	0.962	0.970	0.903	NaN
Precision	NaN	0.953	0.965	0.335	0.965
FPR	0.016	0.010	0.016	0.026	0.016

The performances of classifiers with 15 features are tabulated in Table 8. The RF achieves the highest accuracy of 99.81% compared to other classifiers. The result shows RF, RT and J48 have good ability to detect Normal, DoS/DdoS, Bot and Brute Force traffic, however suffer in detecting Web Attack and Infiltration traffics. Furthermore, RF, RT and J48 have a low FPR of 0.005, and the lowest FPR achieved by BN with FPR of 0.002. The RF, RT and J48 have good Precision and Recall with value of 0.998.

Next, the classifiers' performances with 22 selected features are listed in Table 9. The result shows RF again has the highest accuracy of 99.86% compared to others. Even this classifier has a good recall value of 0.999 and low FPR value of 0.003, unfortunately the precision value indicates a NaN. On the other hand, RF cannot detect Infiltration using

TABLE 8. Performance metric with 15 features.

Detection	RF	BN	RT	NB	J48
Normal	0.999	0.874	0.999	0.304	0.999
Dos/ DDoS	0.999	0.97	0.999	0.965	0.999
Port Scan	0.997	0.995	0.997	0.992	0.997
Bot	0.706	0.985	0.725	0.457	0.713
Web Attack	0.116	0.993	0.116	0.829	0.110
Infiltration	0.200	0.400	0.600	0.600	0
Brute Force	0.995	0.996	0.995	0.999	0.996
Recall	0.998	0.996	0.998	0.436	0.998
Precision	0.998	0.895	0.998	0.913	0.998
FPR	0.005	0.002	0.005	0.031	0.005

TABLE 9. Performance metric with 22 features.

Detection	RF	BN	RT	NB	J48
Normal	0.999	0.927	0.999	0.358	0.999
DoS/ DDoS	0.999	0.981	0.997	0.723	0.999
Port Scan	0.996	0.992	0.994	0.991	0.999
Bot	0.762	0.989	0.777	0.570	0.698
Web Attack	0.788	0.986	0.743	0.846	0.130
Infiltration	0	0.600	0.400	0.800	0
Brute Force	0.997	0.994	0.996	0.983	0.995
Recall	0.999	0.938	0.998	0.447	0.998
Precision	NaN	0.995	0.998	0.925	NaN
FPR	0.003	0.004	0.004	0.017	0.004

TABLE 10. Performance metric with 35 features.

Detection	RF	BN	RT	NB	J48
Normal	0.999	0.92	0.998	0.693	0.999
Dos/ DDoS	0.998	0.983	0.998	0.673	0.999
Port Scan	0.994	0.991	0.993	0.989	0.999
Bot	0.713	0.985	0.755	0.494	0.691
Web Attack	0.651	0.990	0.716	0.955	0.116
Infiltration	0.000	0.600	0.200	0.800	0.200
Brute Force	0.993	0.989	0.993	0.947	0.993
Recall	0.998	0.933	0.998	0.708	0.998
Precision	NaN	0.993	0.998	0.923	0.998
FPR	0.004	0.006	0.004	0.013	0.004

the selected features. With 22 selected features, all classifiers have good TPR to detect DoS/DDoS. PortScan and Brute Force. For Normal traffic RF, BN, RT and J48 achieve good TPR, only NB has a low TPR.

The performances of the classifiers with 35 selected features are listed in Table 10. Similar to the previous results, RF has the highest accuracy of 99.83%, the recall of 0.998, and FPR of 0.004. Nevertheless, the precision noted as NaN. This result shows that RF cannot detect Infiltration. Surprisingly NB achieves better performance than before with 70.84% accuracy, even this achievement lower than other methods, however, it has a good precision with a value of 0.923.

The performances of classifiers with 52 selected features are tabulated in Table 11. It is shown that J48 has a better performance with accuracy of 99.87%, recall of 0.999, precision of 0.999 and low FPR of 0.002 compared to other classifiers.

The performances of classifiers using 57 selected features are listed in Table 12. BN is able to detect all types of traffic with good TPR values.

TABLE 11. Performance metric with 52 features.

Detection	RF	BN	RT	NB	J48
Normal	0.999	0.932	0.998	0.400	0.999
Dos/ DDoS	0.998	0.978	0.997	0.715	0.999
Port Scan	0.994	0.991	0.993	0.931	0.999
Bot	0.668	0.989	0.732	0.774	0.698
Web Attack	0.942	0.990	0.925	0.993	0.949
Infiltration	0.000	1.000	0.000	0.800	0.000
Brute Force	0.993	0.994	0.992	0.963	0.993
Recall	0.998	0.942	0.998	0.476	0.999
Precision	NaN	0.994	0.998	0.880	0.999
FPR	0.004	0.009	0.004	0.035	0.002

TABLE 12. Performance metric with 57 features.

Detection	RF	BN	RT	NB	J48
Normal	0.999	0.932	0.999	0.358	0.999
Dos/ DDoS	0.998	0.973	0.997	0.724	0.999
Port Scan	0.994	0.991	0.993	0.489	0.999
Bot	0.668	0.989	0.751	0.777	0.721
Web Attack	0.932	0.990	0.911	0.993	0.949
Infiltration	0.000	1.000	0.200	0.800	0.000
Brute Force	0.993	0.994	0.990	0.963	0.993
Recall	0.998	0.942	0.998	0.871	0.999
Precision	NaN	0.994	0.998	0.871	0.999
FPR	0.004	0.011	0.004	0.037	0.002

TABLE 13. Performance metric with 77 (all) features.

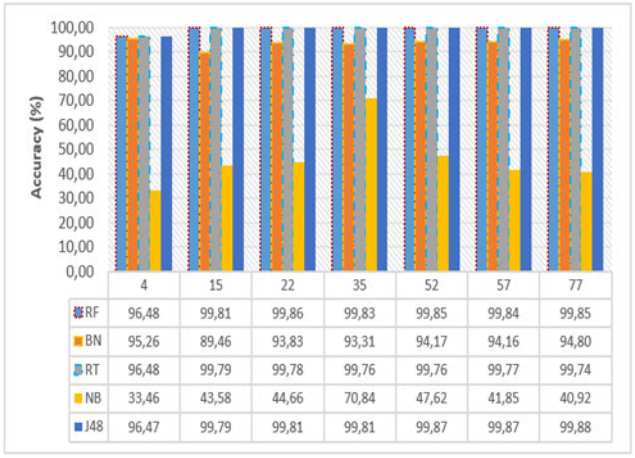
Detection	RF	BN	RT	NB	J48
Normal	0.999	0.940	0.998	0.333	0.999
Dos/ DDoS	0.998	0.974	0.996	0.731	0.999
Port Scan	0.994	0.991	0.993	0.660	0.999
Bot	0.653	0.989	0.675	0.774	0.740
Web Attack	0.935	0.990	0.894	0.983	0.966
Infiltration	0.000	1.000	0.200	0.800	0.000
Brute Force	0.994	0.995	0.992	0.979	0.995
Recall	0.998	0.948	0.997	0.409	0.999
Precision	NaN	0.993	0.997	0.874	NaN
FPR	0.004	0.010	0.005	0.040	0.002

Lastly, the performances of classifiers using all features are tabulated in Table 13. By using all features, BN is able to detect all types of traffic with good TPR. Observation on Table 11, Table 12, and Table 13 leads to conclusion that RF, RT, and J48 with 53, 57, and all features have a good ability to detect Normal, Dos/DDoS, Brute Force as well as Bot attacks traffics. However, RF, RT, and J48 suffer in detecting Infiltration attack traffic, whereas BN and NB have a good ability to detect it.

D. ANALYSIS

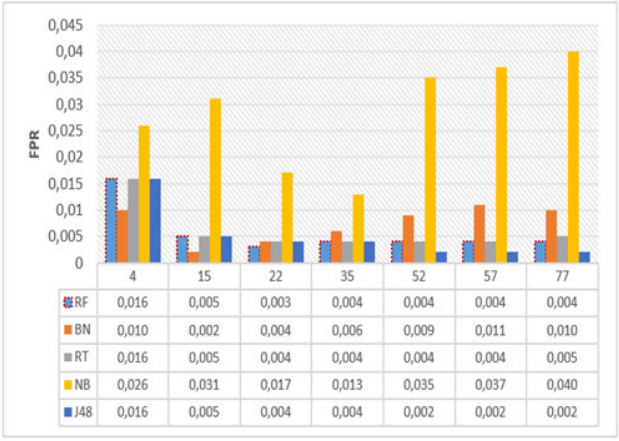
Implementation of the proposed Information Gain feature selection in the experiments yields ranked features according to their weight scores. Features with higher weight scores represent more relevant and significant features of an attack.

As can be observed from Table 5, the top four features (out of 77) with their scores are resulted from the experiment. Thus, features with IDs 41, 13, 65, and 8 are the most relevant and significant features for detecting any attacks and appear in any of features subsets.



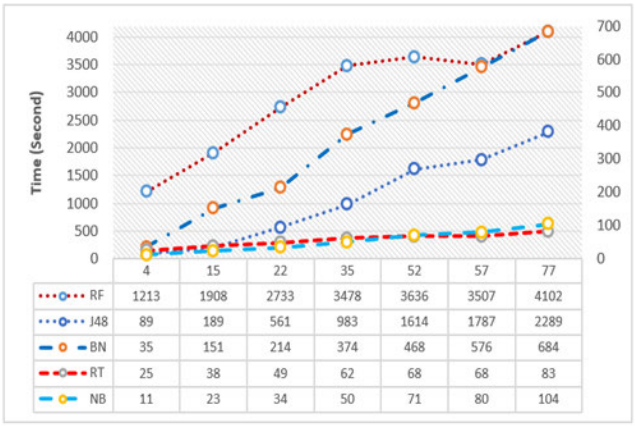
	4	15	22	35	52	57	77
RF	96,48	99,81	99,86	99,83	99,85	99,84	99,85
BN	95,26	89,46	93,83	93,31	94,17	94,16	94,80
RT	96,48	99,79	99,78	99,76	99,76	99,77	99,74
NB	33,46	43,58	44,66	70,84	47,62	41,85	40,92
J48	96,47	99,79	99,81	99,81	99,87	99,87	99,88

FIGURE 2. Accuracy of selected features.



	4	15	22	35	52	57	77
RF	0,016	0,005	0,003	0,004	0,004	0,004	0,004
BN	0,010	0,002	0,004	0,006	0,009	0,011	0,010
RT	0,016	0,005	0,004	0,004	0,004	0,004	0,005
NB	0,026	0,031	0,017	0,013	0,035	0,037	0,040
J48	0,016	0,005	0,004	0,004	0,002	0,002	0,002

FIGURE 3. FPR of selected features.



	4	15	22	35	52	57	77
RF	1213	1908	2733	3478	3636	3507	4102
J48	89	189	561	983	1614	1787	2289
BN	35	151	214	374	468	576	684
RT	25	38	49	62	68	68	83
NB	11	23	34	50	71	80	104

FIGURE 4. Execution time.

Overall, RF, BN, RT and J48 classifiers are able to detect well the normal traffic, DoS/DDoS, Port Scan, Brute Force and Web attacks traffic using the features subsets of 35, 52, and 77. Literatures study supports this finding as the classifiers use robust decision tree learning algorithm.

For the case of Infiltration attack traffic detection, NB is able to detect with TPR value of 0.800 using features subsets of 22 and 35, and perfectly detect (with TPR value of 1.000) using features subsets of 52, 57 and 77. The reason is, because significant features representing infiltration attack traffic appears in the features subsets of 52, 55, 77. Unfortunately, other classifiers; RF, BN, RT and J48 are unable to detect well the Infiltration attack traffic. The small amount of this type of attack traffic in the dataset may cause the bad performance of its detection. As mentioned in subsection 4.A, CCIDS-2017 contains imbalanced data, which is a big challenge in detecting anomalies/attacks.

Similar to the case of Infiltration attack, all classifiers are not able to detect well the Web Attack traffic using features subset of 4. Then, only BN and NB classifiers are able to detect the Web Attack traffic using features subset of 15 with the TPR value of 0.993 and 0.829, respectively.

As for Bot Attack traffic detection, RF, BN, RT, and J48 are able to detect the traffic using certain features subsets, but with lower TPR values.

Furthermore, considering the Precision and Recall values, in general the five classifiers detect the traffic relatively well. Nevertheless, in some cases the classifiers produce NaN values. Those cases may happen because of the implementation of 10-Fold Cross Validation in the experiment, which divides the dataset into ten folds (data portion). As the amount of attack traffics for Infiltration, Bot and Web attacks are relatively small, thus, some folds do not contain those traffics. Therefore, it affects the ability to detect the attack during the training stage. Specifically, for the Infiltration attack traffic which has very small amount in the dataset.

The experiment results show that the type and number of selected features may impact significantly the performance of the detection. Fig. 2 Shows the summary of classifiers' accuracy impacted by the number of selected features resulted by the proposed Information Gain. The proposed Information Gain achieves the highest accuracy of 99.86% for RF and 99.78% for RT, using features subset of 22.

On the other hand, the proposed Information Gain improves NB's accuracy by up to 70.84% with 35 selected features. BN and J48 do not have any significant improvement compared with the use of all features in the analysis.

Besides the accuracy, selected features impact the FPR, as shown in Fig. 3. As for the FPR, the use of 22 selected features affected RF's FPR up to 0.003. It is slightly decrease compared to the use of all features. In the case of BN, 15 selected features affected FPR up to 0.002. This is the lowest FPR amongst the number of selected features. Similar to RF, the use of 22 selected features affected RT's FPR up to 0.004. The proposed Information Gain feature selection has a significant impact on NB's FPR. This impact affected by 4, 15, 22, and 35 features subsets. For J48, the proposed Information Gain does not reduce FPR, only increases when compared to all features subset.

This work also analyzes the effect of execution time for the selected features process. Fig. 4 shows the summary of the execution time to obtain each feature subset using RF, J48,

BN RT, and NB. The relevant selected features process has very significant impact on RF, J48, and BN. The execution time of RT and NB are relatively very small. Overall, the more numbers of features to analyze the more time is required for execution.

V. CONCLUSIONS

This work has discussed experimentation as a proof of concept on impact of feature selection in improving anomaly detection accuracy. Information Gain is designated because of its ability to calculate the weight of features' information.

RF classifier outperforms others in the experiments using features subsets of 15, 22 and 35. Whilst J48 performs the best using features subsets of 52, 57 and 77. Other finding in the experiment is that, although BN has a low accuracy level compared to RF and J48, however it is able to detect all traffics using features subsets of 52, 57 and 77. Furthermore, experiment results show that the selected features decrease the FPR level, especially for BN.

With regards to the investigation on processing time, experimental results confirm that the number of selected features affect the execution time.

The proposed Information Gain produces ranked features based on their weight values. However, expert intervention is still needed to determine the minimum weight value, which affects the number of features selected.

The authors plan to work on different feature selection methods to design an optimal feature selection mechanism. Analysis of each features subset that affects each type of attack will also be carried out as a future work.

REFERENCES

[1] J. Zhang, H. Li, Q. Gao, H. Wang, and Y. Luo, "Detecting anomalies from big network traffic data using an adaptive detection approach," *Inf. Sci.*, vol. 318, pp. 91–110, Oct. 2015.

[2] V. Jyothsna and V. V. Rama Prasad, "FCAAIS: Anomaly based network intrusion detection through feature correlation analysis and association impact scale," *ICT Express*, vol. 2, no. 3, pp. 103–116, Sep. 2016.

[3] A. Satoh, Y. Nakamura, and T. Ikenaga, "A flow-based detection method for stealthy dictionary attacks against secure shell," *J. Inf. Secur. Appl.*, vol. 21, pp. 31–41, Apr. 2015.

[4] A. Juvonen and T. Hamalainen, "An efficient network log anomaly detection system using random projection dimensionality reduction," in *Proc. 6th Int. Conf. New Technol., Mobility Secur. (NTMS)*, Mar. 2014, pp. 1–5.

[5] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Electr. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014.

[6] A. S. Eesa, Z. Orman, A. Mohsin, and A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," *Expert Syst. Appl.*, vol. 42, no. 5, pp. 2670–2679, Apr. 2015.

[7] I. Ahmad, M. Hussain, A. Alghamdi, and A. Alelaiwi, "Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components," *Neural Comput. Appl.*, vol. 24, nos. 7–8, pp. 1671–1682, Jun. 2014.

[8] S.-H. Kang and K. J. Kim, "A feature selection approach to find optimal feature subsets for the network intrusion detection system," *Cluster Comput.*, vol. 19, no. 1, pp. 325–333, Mar. 2016.

[9] A. I. Madbouly, S. A. King Abdulaziz University Jeddah, A. M. Gody, and T. M. Barakat, "Relevant feature selection model using data mining for intrusion detection system," *Int. J. Eng. Trends Technol.*, vol. 9, no. 10, pp. 501–512, Mar. 2014.

[10] E. Popoola and A. Adewumi, "Efficient feature selection technique for network intrusion detection system using discrete differential evolution and decision tree," *Int. J. Netw. Secur.*, vol. 19, no. 5, pp. 660–669, 2017.

[11] B. A. Tama and K. H. Rhee, "A combination of PSO-based feature selection and tree-based classifiers ensemble for intrusion detection systems," *Adv. Comput. Sci. Ubiquitous Comput.*, vol. 373, pp. 489–495, Feb. 2015.

[12] N. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Procedia Comput. Sci.*, vol. 89, pp. 213–217, Jan. 2016.

[13] M. S. Pervez and D. M. Farid, "Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs," in *Proc. 8th Int. Conf. Softw., Knowl., Inf. Manage. Appl.*, Dec. 2014, pp. 1–6.

[14] R. Sheikhpour, M. A. Sarram, S. Gharaghani, and M. A. Z. Chahooki, "A survey on semi-supervised feature selection methods," *Pattern Recognit.*, vol. 64, pp. 141–158, Apr. 2017.

[15] S. Sahu and B. M. Mehtre, "Network intrusion detection system using J48 decision tree," in *Proc. Int. Conf. Adv. Comput. Commun. Informat.*, 2015, pp. 2023–2026.

[16] A. Tesfahun and D. L. Bhaskari, "Effective hybrid intrusion detection system: A layered approach," *Int. J. Comput. Netw. Inf. Secur.*, vol. 7, no. 3, pp. 35–41, Feb. 2015.

[17] K. Rai, M. S. Devi, and A. Guleria, "Decision tree based algorithm for intrusion detection," *Int. J. Adv. Netw. Appl.*, vol. 7, no. 4, pp. 2828–2834, 2016.

[18] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *J. Comput. Sci.*, vol. 25, pp. 152–160, Mar. 2018.

[19] Akashdeep, I. Manzoor, and N. Kumar, "A feature reduced intrusion detection system using ANN classifier," *Expert Syst. Appl.*, vol. 88, pp. 249–257, Dec. 2017.

[20] A. A. A. Hadi, "Performance analysis of big data intrusion detection system over Random Forest algorithm," *Int. J. Appl. Eng. Res.*, vol. 13, no. 2, pp. 1520–1527, 2018.

[21] M. El Boujnouni and M. Jedra, "New intrusion detection system based on support vector domain description with information gain metric," *Int. J. Netw. Secur.*, vol. 20, no. 1, pp. 25–34, 2018.

[22] W. Wang, Y. He, J. Liu, and S. Gombault, "Constructing important features from massive network traffic for lightweight intrusion detection," *IET Inf. Secur.*, vol. 9, no. 6, pp. 374–379, Nov. 2015.

[23] M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," *IEEE Trans. Comput.*, vol. 65, no. 10, pp. 2986–2998, Oct. 2016.

[24] S. Bhattacharya and S. Selvakumar, "Multi-measure multi-weight ranking approach for the identification of the network features for the detection of DoS and probe attacks," *Comput. J.*, vol. 59, no. 6, pp. 923–943, Jun. 2016.

[25] F. Chen, Z. Ye, C. Wang, L. Yan, and R. Wang, "A feature selection approach for network intrusion detection based on tree-seed algorithm and K-Nearest neighbor," in *Proc. IEEE 4th Int. Symp. Wireless Syst.*, Sep. 2018, pp. 68–72.

[26] H. Peng, C. Ying, S. Tan, B. Hu, and Z. Sun, "An improved feature selection algorithm based on ant colony optimization," *IEEE Access*, vol. 6, pp. 69203–69209, 2018.

[27] P. Tao, Z. Sun, and Z. Sun, "An improved intrusion detection algorithm based on GA and SVM," *IEEE Access*, vol. 6, pp. 13624–13631, 2018.

[28] T. A. Alhaj, M. M. Siraj, A. Zainal, H. T. Elshoush, and F. Elhaj, "Feature selection using information gain for improved structural-based alert correlation," *PLoS ONE*, vol. 11, no. 11, 2016, Art. no. e0166017.

[29] M. K. Kundu, D. P. Mohapatra, A. Konar, and A. Chakraborty, "Decision tree techniques applied on NSL-KDD data and its comparison with Various feature selection techniques," *Smart Innov. Syst. Technol.*, vol. 27, no. 1, pp. 205–211, 2014.

[30] Z. Karimi, M. Mansour Riahi Kashani, and A. Harounabadi, "Feature ranking in intrusion detection dataset using combination of filtering methods," *Int. J. Comput. Appl.*, vol. 78, no. 4, pp. 21–27, Sep. 2013.

[31] P. Bereziáski, B. Jasiul, and M. Szpyrka, "An entropy-based network anomaly detection method," *Entropy*, vol. 17, no. 4, pp. 2367–2408, Apr. 2015.

[32] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. Burlington, MA, USA: Morgan Kaufmann, 2011.

[33] M. C. Belavagi and B. Muniyal, "Performance evaluation of supervised machine learning algorithms for intrusion detection," *Procedia Comput. Sci.*, vol. 89, pp. 117–123, May 2016.

[34] J. Jiang, Q. Wang, Z. Shi, B. Lv, and B. Qi, "RST-RF: A hybrid model based on rough set theory and random forest for network intrusion detection," in *Proc. ACM Int. Conf. Process.*, 2018, pp. 77–81.

[35] R. K. Singh, S. Dalal, V. K. Chauhan, and D. Kumar, "Optimization of FAR in intrusion detection system by using random forest algorithm," *SSRN Electron. J.*, vol. 5, pp. 3–6, Mar. 2019.

[36] B. Dhruba and K. Jugal, *Network Anomaly Detection A Machine Learning Perspective*. Boca Raton, FL, USA: CRC Press, 2014.

[37] M. Reazul, A. Rahman, and T. Samad, "A network intrusion detection framework based on Bayesian network using wrapper approach," *Int. J. Comput. Appl.*, vol. 166, no. 4, pp. 13–17, May 2017.

[38] N. Ding, H. Gao, H. Bu, and H. Ma, "RADM:Real-time anomaly detection in multivariate time series based on Bayesian network," in *Proc. IEEE Int. Conf. Smart Internet Things*, Aug. 2018, pp. 129–134.

[39] A. Niranjan, D. H. Nutan, A. Nitish, P. D. Shenoy, and K. R. Venugopal, "ERCR TV: Ensemble of random committee and random tree for efficient anomaly classification using voting," in *Proc. 3rd Int. Conf. Converg. Technol.*, Apr. 2018, pp. 1–5.

[40] R. Chitrakar and H. Chuanhe, "Anomaly detection using support vector machine classification with k-Medoids clustering," in *Proc. 3rd Asian Himalayas Int. Conf. Internet*, Nov. 2012, pp. 1–5, doi: 10.1109/AHICI.2012.6408446.

[41] S. Thaseen, *Intrusion Detection Model Using fusion of PCA and optimized SVM*. Boca Raton, FL, USA: CRC Press, 2014, pp. 879–884.

[42] T. Mehmood, "SVM for network anomaly detection using ACO feature subset," in *Proc. Int. Symp. Math. Sci. Comput. Res.*, 2015, pp. 121–126.

[43] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016.

[44] K. Goeschel, "Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and naive bayes for off-line analysis," in *Proc. SoutheastCon*, Mar. 2016, pp. 1–6.

[45] S. Shakya and S. Sigdel, "An approach to develop a hybrid algorithm based on support vector machine and naive Bayes for anomaly detection," in *Proc. Int. Conf. Comput. Commun. Autom. (ICCCA)*, Jan. 2017, pp. 323–327.

[46] S. Aljawarneh, M. B. Yassein, and M. Aljundi, "An enhanced J48 classification algorithm for the anomaly intrusion detection systems," *Cluster Comput.*, vol. 22, pp. 10549–10565, Sep. 2017.

[47] A. P. Muniyandi, R. Rajeswari, and R. Rajaram, "Network anomaly detection by cascading K-means clustering and C4.5 decision tree algorithm," *Procedia Eng.*, vol. 30, pp. 174–182, Feb. 2012.

[48] R. Panigrahi and S. Borah, "A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems," *Int. J. Eng. Technol.*, vol. 7, no. 24, pp. 479–482, 2018.

[49] P. Soni and P. Sharma, "An intrusion detection system based on KDD-99 data using data mining techniques and feature selection," *Int. J. Soft Comput. Eng.*, no. 3, pp. 2231–2307, May 2014.

[50] M. Nikhitha and M. A. Jabbar, "K Nearest Neighbor based model for Intrusion Detection System," *Int. J. Recent Technol. Eng.*, vol. 8, no. 2, pp. 2258–2262, 2019.

[51] M. Abualkibash, "Machine learning in network security using KNIME analytics," *Int. J. Netw. Secur. Appl.*, vol. 11, no. 5, pp. 1–14, Sep. 2019.

[52] V. Vijayakumar and V. Neelanarayanan, "Intrusion detection model using chi square feature selection and modified Naïve Bayes classifier," in *Smart Innovation, Systems and Technologies*, vol. 49. Cham, Switzerland: Springer, 2016, p. 15.

[53] G. Kirubavathi and R. Anitha, "Botnet detection via mining of traffic flow characteristics," *Comput. Electr. Eng.*, vol. 50, pp. 91–101, Feb. 2016.

[54] G. Serpen and E. Aghaei, "Host-based misuse intrusion detection using PCA feature extraction and kNN classification algorithms," *Intell. Data Anal.*, vol. 22, no. 5, pp. 1101–1114, Sep. 2018.

[55] T. Hastie, R. Tibshirani, J. Friedman, "The elements of statistical learning: Data mining, inference, and prediction," in *Printing*. New York, NY, USA: Springer, 2017.

[56] A. Yulianto, P. Sukarno, and N. Suwastika, "Improving AdaBoost-based intrusion detection system (IDS) performance on CIC IDS 2017 dataset," *J. Phys., Conf. Ser.*, vol. 1192, Mar. 2019, Art. no. 012018, doi: 10.1088/1742-6596/1192/1/012018.

[57] S. S. Sivatha Sindhu, S. Geetha, and A. Kannan, "Decision tree based light weight intrusion detection using a wrapper approach," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 129–141, Jan. 2012.

[58] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, 2018, pp. 108–116, doi: 10.5220/0006639801080116.

KURNIABUDI (Member, IEEE) received the master's degree in computer science from Universitas Putra Indonesia YPTK Padang, West Sumatera, Indonesia. He is currently pursuing the Ph.D. degree with the Faculty of Engineering, Universitas Sriwijaya. He is currently a Senior Lecturer with the Faculty of Computer Science, Universitas Dinamika Bangsa, Indonesia. His research interests include technology adoption, information technology, information security, and network security.

DERIS STIAWAN received the Ph.D. degree in computer engineering from Universiti Teknologi Malaysia, Malaysia. He is currently an Associate Professor with the Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya. His research interests include computer networks, intrusion detection/ prevention systems, and heterogeneous networks.

DARMAWIJOYO received the Doctor of Mathematics degree from the Delft University of Technology, The Netherlands. He is currently an Associate Professor with the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Sriwijaya. His research interests include problem solving, applied mathematics, modeling, and mathematical thinking.

MOHD YAZID BIN IDRIS (Member, IEEE) received the M.Sc. degree in software engineering, in 1998, and the Ph.D. degree in information technology (IT) security, in 2008. He is currently an Associate Professor with the Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia. In software engineering, he focuses on the research of designing and development of mobile and telecommunication software. His main research interest in IT security includes intrusion prevention and detection (IPD).

ALWI M. BAMHDI received the M.Sc. and Ph.D. degrees in computer science from Heriot-Watt University, U.K., in 2010 and 2014, respectively. He is currently an Assistant Professor with the Department of Computer Sciences, College of Computing, Umm Al-Qura University, Al Qunfudhah, Saudi Arabia. His research interests include mobile ad hoc networks, wireless sensor networks, information security, cyber security, computer vision and simulation, and performance evaluation.

RAHMAT BUDIARTO received the B.Sc. degree from the Bandung Institute of Technology, in 1986, and the M.Eng. and Dr.Eng. degrees in computer science from the Nagoya Institute of Technology, in 1995 and 1998, respectively. He is currently a Full Professor with the College of Computer Science and IT, Albaha University, Saudi Arabia. His research interests include intelligent systems, brain modeling, IPv6, network security, wireless sensor networks, and MANETs.