International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)
ISSN: 2305-0543 (Online), ISSN: 2411-6173 (Print)
Vol. 10(38), Oct. 2020, PP. 4792-4799

A Feature Selection Method to Improve Intrusion Detection in Computer Networks using Decision Tree and Genetic Algorithm

Masomeh Latif
Young Researchers and Elite Club, West Tehran Brach
Islamic Azad University
Tehran, Iran
masomeh13latif@gmail.com

Zohreh Bateni
Department of Computer Engineering, Central Tehran Branch
Islamic Azad University
Tehran, Iran
zbateni@hotmail.com

***Abstract*—Today, network security has become a very hot topic in the field of information technology. Therefore, increasing the efficiency and reducing the error of intrusion detection systems is very important. The data that are collected from network traffic by different methods of data mining are analyzed to distinguish normal behaviors from attacks. One of the important challenges of data mining techniques that increase training time and reduce accuracy is the large number of features *of* network traffic data. Some of these features are very important for intrusion detection and some others are insignificant and must be removed from the dataset before using data mining algorithm. Therefore, the feature selection is an important step before classification and data analysis. Feature selection is a kind of NP optimization problem. The metaheuristic and evolutionary algorithms are widely used in solving optimization problems and provide more accurate and better responses than classical methods. Genetic algorithm is one of these metaheuristic algorithms. This algorithm is able to select the optimal subset with high accuracy. The standard NSL-KDD dataset is used for training and testing in this study. Genetic algorithm and C4.5 decision tree will be used for feature selection and classification. The results after feature selection indicate the improvement of all performance measurement parameters. Also, the feature vector obtained from the feature selection is used in the KNN classifier, which improves the classification accuracy and reduces its error.**

***Keywords*—Intrusion detection, Feature Selection, Network Security, Genetic Algorithm**

I. INTRODUCTION

With the increasing number of attacks on computer networks and dependence of various domain activities on the services provided by computer networks, efforts to prevent and detect of intrusion are urgently needed. Intrusion is a set of activities that endanger the confidentiality, integrity or availability of a resource. Intrusion detection is the process of identifying and responding to malicious activity of network resources. The purpose of intrusion detection systems (IDS) is not to prevent attacks, but to detect attacks and security flaws in computer networks and report them to the system administrator. IDS inspect all networks and host activity on the network and identify suspicious patterns that may indicate a network or system attack. In today's IDS, intelligent data-based data mining methods and technologies are used to identify effective and efficient patterns in intrusion detection. IDS use analytical methods to detect attacks, identify attack sources, and send alerts to network administrators [1]. IDSs are classified into two groups, Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS), based on their location in the network system and scope of activity. These systems are also commonly used in combination to achieve maximum performance and security, known as Distributed Intrusion Detection Systems (DIDS) [2, 3]. HIDS detect unauthorized activities on the host computer. These systems run only on host or single computers and are not aware of the entire network. These types of systems only monitor incoming and outgoing packages to a computer and alert the network administrator or computer user when detecting intrusions or suspicious activity [3]. NIDS monitor and analyze traffic across the network to identify threats. These systems detect malicious activities such as denial of service (DOS), port scanning and other attacks throughout the network. These systems detect the intrusion detection of network traffic for each packet in real-time and close to it in order to identify intrusion patterns [3]. Intrusion detection is classified into two categories: signature-based method and anomaly-based method. The signature-based intrusion detection method uses known attack patterns to detect intrusions. This method has a database of signatures or attack patterns. This method works very well in detecting known attacks that have their signatures in the intrusion detection database, but is not able to detect new attacks that do not exist in the database [4, 5]. Anomaly-based intrusion detection method compares user behavior with normal behavior stored in a database and uses it to detect unknown attacks and uses statistical methods to find activities do not match the normal behavior pattern. Creating normal behavior in this method is very important because the normal behavior of users may change and therefore the intrusion detection system uses this method must update itself with these changes. An important issue in constructing a normal behavior model is to select the appropriate features as input for the model. If the input parameters are determined by the security expert, there is no guarantee all effective intrusion detection features will be selected correctly. If important intrusive-related features are missed incorrectly, it will be very difficult to distinguish an attack from normal behavior. Also, non-intrusive features can

Article History:
Received Date: Apr. 14, 2020, Accepted Date: Jul. 06, 2020, Available Online: Sep. 01, 2020

reduce the efficiency of intrusion detection. Therefore, feature selection has a significant impact on detecting anomalies in network traffic [6]. Feature selection is an important issue in the data mining field, especially for high-dimensional data. If the dataset has N attributes, then it will have 2^N subsets and checking all of these subsets will be a NP-hard problem and there is no definite solution to these problems so far [7]. In recent approaches, metaheuristic algorithms such as Genetic, Ant Colony, Particle Swarm and etc have been used by many researchers for feature selection. In this paper, in the proposed feature selection method, the genetic algorithm will be used to find the optimal subset of attributes. The C4.5 decision tree is used as a classifier with the aim of minimizing its classification error by using a subset of the NSL_KDD intrusion detection dataset. The purpose of this paper is to minimize the KNN classification error by applying the feature vector obtained by minimizing the C4.5 decision tree classification error. The results show that this feature vector also enhances the classification accuracy in KNN.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 gives a brief overview of methods and proposed technique used in this research. Section 4 presents the dataset and experimental results and discussion and finally, conclusions are provided in Section 5.

II. Related work

In network intrusion detection, accurate feature selection leads to optimal performance. To remove unnecessary features, several methods have been proposed. In [8] an intrusion detector is introduced using the ant colony optimization algorithm (ACO) for feature selection and k-nearest neighbor algorithm (KNN) for classification. In the proposed method, the memory consumption is significantly reduced and because of the reduction in the number of features used in intrusion detection, the amount of CPU needed is reduced. The proposed method reduced the number of features by approximately 88% and the detection error reduced by around 24% using KDD Cup 99 test data set. In [1], a new feature display method called CANN is introduced. This method uses clustering and KNN to represent the new one-dimensional dataset. This method converts the features of the original dataset into a distance-based one-dimensional dataset. Then this new dataset is tested with the KNN classifier. Experimental results on the KDD Cup 99 dataset show that this proposed method has a higher accuracy and detection rate than the KNN and support vector machine (SVM) methods on the 6-dimensional KDD dataset and its false positive rate is lower. [9] Proposed a feature selection method based on the modified binary coded ant colony optimization algorithm (MBACO) combined with the genetic algorithm (GA). In this method every feature is like a binary bit and has two directions: one to select the feature and one to not selecting it. The proposed method is tested with GA methods, binary ant colony optimization (BACO), advanced binary ant colony optimization (ABACO), binary particle swarm optimization (BPSO) and binary differential evolution (BDE) on well-known datasets in UCI. The experimental results show that the proposed method is robust and consistent and performs better than the other methods compared in this paper. [10] Introduced a multi-step algorithm for detecting anomalies in network traffic. Initially, a feature selection method is introduced. This method can identify the optimal subset of attributes. Then, a tree-based subspace clustering algorithm is developed for the high-dimensional dataset. A fast distributed framework for feature extraction and preparation of raw network data is also provided. Before implementing the proposed anomaly detection algorithm, the first depth-clustering algorithm arranges the data. The proposed method is evaluated on various datasets such as UCI ML, TUIDS, KDD Cup 99 and NSL-KDD. The performance of the proposed method is compared with all well-known classification methods such as C4.5, ID3, CN2, and SVM. Experimental results show that the proposed method performs better in almost all datasets in terms of detection rate, false positive rate and detection accuracy. [11] Proposed a three-level network intrusion detection method. These three levels are: 1- Identifying underlying contextual patterns of network traffic data by establishing reliable rules for network anomaly detection. 2- Creating a predictive model to determine the exact categories of attack. 3- Developing of a visual analytical tool for interactive visual analysis and validation of the results of this proposed method. The discrete wavelet transform (DWT) method is used for feature selection. When the features are raw, the boundary between the Probe, DOS and R2L attack categories is not clear, but the DWT feature extraction method distinguishes between these groups. The NSL-KDD dataset is used to examine this method. The proposed prediction model can detect attack categories with accuracy of about 96%. In [9], a new approach is proposed using the feature reduction and multiple classification methods. In the proposed method, ID3 and KNN-ACO are tested as classifier on the KDD Cup 99 dataset. The ID3 decision tree is used to select and reduce features using entropy and information gain. The KNN and ant colony optimization (ACO) algorithms are used to classify the data in the reduced KDD dataset. Experimental results show that the proposed method is superior to the detection of all four types of Probe, DOS, R2L and U2R attacks over the Support Vector Machine (SVM) and Post-Release Neural Network (BKP). In [13] an intrusion detection framework is proposed using a new optimization method called time-varying chaotic particle swarm optimization (TVCPSO) witch simultaneously performs parameter tuning and feature selection for multi-criteria linear programming (MCLP) and Support Vector Machine (SVM). The results of the experiments performed on the NSL-KDD dataset show that the proposed method not only has higher detection accuracy rate but also is able to select the best set of features. Evaluations show that multi-criteria linear programming in intrusion detection problem is comparable with SVM method and also has higher true detection rate than SVM method in detecting R2L and U2R attacks. In [14], GA is used for feature selection. Feature selection is also performed by other methods such as CFS, IG and CAE, and then the reduced dataset is tested with both Naive Bayes and J48 classifiers. Experimental results show that the proposed method on the NSL-KDD dataset will select the optimal feature dataset and will improve the performance

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

of the Naïve Bayes classifier and increase the accuracy and reduce the time required for classification.

III. OVERVIEW OF METHODS AND PROPOSED TECHNIQUE

A. Genetic Algorithm

Genetic algorithm (GA) is a bio-inspired search method that offers an approximate answer and solution to optimization problems [15]. The main components of the genetic algorithm are: Chromosome, Population and Fitness function. This algorithm has four operators: initialization, selection, crossover and mutation [16]. The algorithm consists of the following steps [14]:

1- Generating a random initial population of chromosomes (a chromosome is a set of genes that can be bits, numbers, or characters).
2- Evaluating initial population by the fitness function.
3- Selecting parents and using the crossover operator to create a population of children.
4- Selecting members of the population to apply the mutation operator to them and create the mutant population.
5- Combining the three populations and creating a new main population and then evaluating the new population by the fitness function.
6- If the termination condition does not exist, the algorithm is repeated from step 2.
7- End.

B. Decision Tree

Decision tree is one of the classification algorithms. Classification algorithms learn how to create a model from the dataset. The decision tree is initially made from pre-classified data. The most important issue is which feature should be selected to best split the set of instances in each recursion. Different implementations of the decision tree use different feature selection metrics to select the best feature at each tree level. In the ID3 the Information Gain, in C4.5 the Gain Ratio and in the CART the Gini criterion is used to select the attribute at each tree level [17]. C4.5 is an improved ID3 algorithm. As mentioned, C4.5 uses the Gain Ratio criterion to select the best feature in constructing each level of the tree. Suppose D is a Labeled training dataset with h classes: C_1... Ch.

The expected information for classifying a sample in D is calculated as follows [18]:

$$Info\ (D) = -\sum_{i=1}^{h} p_i \log_2 (p_i) \qquad (1)$$

P_i: is the probability that an arbitrary sample from dataset D belongs to class C_i.

Info (D): is the expected information for classifying a data sample.

accuracy will be increased. In a classification problem, if X is the feature vector of the dataset, t is the target output of the dataset, Y is the output of the classification model, and F is

If D is divided into k distinct subsets: D_1, ... , D_k based on the value of X, then the information required for accurate classification is as follows:

$$Info_X\ (D) = \sum_{j=1}^{k} \frac{|D_j|}{|D|} \times Info(D_j) \qquad (2)$$

The difference between *Info (D)* and $Info_X\ (D)$ is called information gain and is calculated as follows

$$Gain\ (X) = Info\ (D) - Info_X\ (D) \qquad (3)$$

The ID3 algorithm uses the information gain criterion to select the separator feature. Information gain biases toward partitions with many outcomes. To overcome this problem, C4.5 uses the measure of gain ratio, which is calculated as follows:

$$GainRatio\ (X) = \frac{Gain(X)}{SpilitInfo(X)} \qquad (4)$$

$$SplitInfo_X\ (D) = -\sum_{j=1}^{k} \frac{|D_j|}{|D|} \times log_2(\frac{|D_j|}{|D|}) \qquad (5)$$

In C4.5, the feature with the highest gain ratio is selected as the separator feature.

C. K-Nearest Neighbors (KNN)

The K-Nearest Neighbor (KNN) algorithm is one of the simplest classification methods, which is able to classify data with the least prior knowledge of data distribution. In this algorithm, a sample is categorized by the votes of the majority of its neighbors, and the sample is assigned in the nearest class, between k near neighbors. The k parameter of KNN classifier represents the number of neighbors in a set of training observations that are nearest to the given observation in validation or testing dataset. Variation of this parameter will affect the accuracy of classifier [19]. If k = 1, the sample is simply assigned to the class of nearest neighbors. If k=3, KNN look to three nearest neighbors of the new instance to find a class for the unknown sample. The KNN classifier often uses Euclidean distance to measure the similarity between two points, which is calculated as follows:

$$dist\ ((x_1, y1), (x_2, y2)) = \sqrt{(x_1 - x_2)^2 + (y_2 - y_2)^2} \qquad (6)$$

Where (x_1, y_1) is the coordinate of the new instance and (x_2, y_2) is the coordinate of the existing sample.

D. Proposed Method

The proposed method for finding the optimal subset of features as follows:

First, the feature selection is formulated in the form of an optimization problem. In an optimal feature selection, due to the elimination of redundant features, classification

the classification algorithm, the objective of the feature selection is to approximate Y with another feature vector such as X^* as follow:

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

$$Y = F(X) \cong F(X^*) \qquad X^* \subset X \tag{7}$$

Feature selection is a two-objective optimization problem and the first step in solving this problem is to determine the objective function.
The first purpose is to minimize the value of E:

$$Min \quad E \quad = \quad |t- \quad F(X^*)| \tag{8}$$

The second goal that exists simultaneously in the feature selection is to reduce the number of inputs to the classification algorithm.

$$Min\ nf = |X^*| \tag{9}$$

The objective function of feature selection can be considered as follows:

$$Z \sim E + nf \tag{10}$$

By adding the coefficients w1 and w2 to the proportion 10, the proportion becomes the equation. Then, the sides of the equation are divided by w1. The result of dividing w2 by w1 is considered to be w. w can be considered a factor of E. This coefficient is called β. Figure 1 shows how the objective function is formulated.

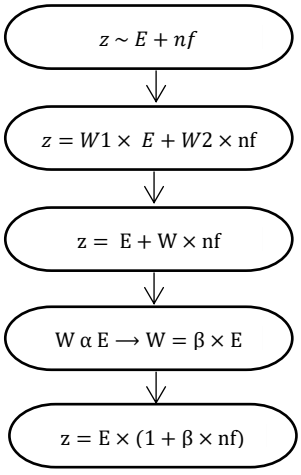

Fig. 1. Objective function

After determining the objective function which is called the cost function in the genetic algorithm, feature selection is done by genetic algorithm. This algorithm, with the appropriate initial parameters, is able to select the optimal subset with high accuracy. Algorithm 1 describes how to do this.

Algorithm 1: Feature selection using binary Genetic algorithm

npop: Number of initial population
M: Number of iterations
nc: Number of offspring
nm: Number of mutants Population

Input : Training data set
Output : Subset of features

```
determine the cost function
Initialize parameters
for i=1 to npop do
    Initialize population
end for
Evaluate and sort population
Store Best population
for j=1 to M do
    for c=1 to nc/2 do
        Select Parents
        Apply Crossover
        Evaluate offspring
    end
    Store popc
    for m=1to nm do
        Select Parents
        Apply Mutation
        Evaluate Mutant
    end
    Store popm
Create Merged Population
Sort Population
Store Best Solution ever found
end
Show Best Solution
```

First, cost function is called in the body of the genetic algorithm and the feature vectors are randomly generated. These vectors are binary, so zero means not selecting that feature and one means selecting it. In the cost function, C4.5 decision tree is created and trained with data from the NSL_KDD training dataset. In this process, the tree uses random feature vectors generated by the genetic algorithm. Training is done by the Kfold method and repeated several times and the average error is considered as the final error. The steps are repeated with the number of iterations specified in the algorithm and finally in the last iteration, the vector with the least cost function is introduced as the optimal solution.
By applying the optimal feature vector, decision tree is trained on the NSL_KDD training dataset. Then the test is done on the constructed tree. Test is performed on the NSL_KDD test dataset. It has 22544 records and contains 17 new attacks from four categories of Dos, Probe, R2L and U2R attacks. The confusion matrix is computed separately for both train and test datasets before and after using the optimal feature vector. Performance measurement parameters including classification accuracy, intrusion detection rate and false alarm rate are calculated for each of the confusion matrix.
In wrapper feature selection methods, the classification algorithms used in feature selection and final model construction are the same. Here, it is assumed that if the feature selection algorithm is highly efficient and the feature vector is selected appropriately, if it is used in any other classification algorithm it should improve the results and

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

reduce the classification error. Therefore, by using this feature vector, the K-Nearest Neighbors (KNN) classifier on the training dataset is trained and modeled. Then the test operation is performed on the test dataset. The confusion matrix is computed separately for both train and test datasets before and after using the feature vector. Performance measurement parameters are also calculated for each of the confusion matrix. Finally, the results are compared before and after the feature selection.

IV. Dataset and Experimental Results

In this section, the intrusion detection dataset is introduced and the results are presented and compared. The proposed method has been implemented in the Matlab 2016 programming language, and the operating environment for the implementation has been a personal computer with an Intel Core i7 2.6 GHz processor and 8GB of memory.

A. dataset

The NSL-KDD dataset is an improved subset of the KDDCUP99 dataset. It is used to evaluate the proposed method in this paper. The NSL-KDD dataset is composed of the train and test dataset, which have 125973 and 22544 records respectively and the test dataset contains 17 kinds of new different attacks. The NSL-KDD dataset contains 41 features and 5 classes, with one normal class and four attack classes. Attack classes are: Denial of Service Attack, Probing Attack, User to Root Attack and Remote to Local Attack, Which are as follows[20]:

- Denial of Service (DoS): In this type of attack, excessive use of system resources causes legitimate requests for resources to be denied.
- Probing: In these attacks, the network or host is scanned to gather information and identify known vulnerabilities. A probe attack is considered to be the first step of a real network or host attack. In these attacks, the attacker may obtain important information for executing dangerous attacks on the system.
- User to Root (U2R): It occurs when an attacker accesses a normal user account by using means such as social engineering, password sniffing and so on.
- Remote to Local (R2L): an attacker with remote access to the victim's machine uses the user's legal account to send packets over the network.

TABLE I. TYPE OF ATTACKS IN TRAIN AND TEST DATASET

Attack Class	Attack type in train dataset	New attack type in test dataset
Denial of Service(DoS)	Neptune, smurf,teardrop,pod, land, back	apache2, mailbomb,processtable, udpstorm
Probe	satan, ipsweep, nmap, portsweep	mscan, saint
Remote to Local(R2L)	imap, warezmaster, phf, multihop, guess password, ,spy, warezclient, ftp write	httptunnel, named, sendmail, snmpgetattack, xlock, xsnoop

In the NSL_KDD dataset, the data type of the features 2, 3 and 4 is symbolic. Features 7, 12, 14, 15, 21 and 22 are Boolean and the rest of the features are numeric. The NSL_KDD dataset feature descriptions and their types and names are shown in Table2 [20, 21]:

TABLE II. NSL_KDD DATASET FEATURE DESCRIPTION

Num	Feature Name	Description	Type
1	Duration	Length of the connection (second)	Numeric
2	Protocol-type	Type of protocol, e.g. tcp, udp, etc.	Symbolic
3	Service	Network service on the destination, e.g., http, telnet, etc.	Symbolic
4	Flag	Normal or error status of the connection	Symbolic
5	Src-bytes	Number of data bytes from source to destination	Numeric
6	Dst-bytes	Number of data bytes from destination to source	Numeric
7	Land	1 if connection is from/to the same host/port; 0 otherwise	Boolean
8	Wrong-fragment	Number of wrong fragments	Numeric
9	Urgent	Number of urgent packets	Numeric
10	Hot	Number of hot indicators	Numeric
11	Num-failed-logins	Number of failed login attempts	Numeric
12	Logged-in	1 if successfully logged in; 0 otherwise	Boolean
13	Num-compromised	Number of compromised condition	Numeric
14	Root-shell	1 if root shell is obtained; 0 otherwise	Boolean
15	Su-attempted	1 if su root command attempted; 0 otherwise	Boolean
16	Num-root	Number of root accesses	Numeric
17	Num-file-creations	Number of file creation operations	Numeric
18	Num-shells	Number of shell prompts	Numeric
19	Num-access-files	Number of operations on access control files	Numeric
20	Num-outbound-cmds	Number of outbound commands in an ftp session	Numeric
21	Is-host-login	1 if the login belongs to the hot list; 0 otherwise	Boolean
22	Is-guest-login	1 if the login is a guest login; 0 otherwise	Boolean
23	Count	Number of connections to the same host as the current connection in the past two seconds	Numeric
24	Srv-count	Number of connections to the same service as the current connection in the past two seconds	Numeric
25	Serror-rate	Percent of connections that have SYN errors	Numeric
26	Srv-serror-rate	Percent of connections that have SYN errors	Numeric
27	Rerror-rate	Percent of connections that have REJ errors	Numeric
28	Srv-rerror-rate	Percent of connections that have REJ errors	Numeric
29	Same-srv-rate	Percent of connections to the same services	Numeric
30	Diff-srv-rate	Percent of connections to dierent services	Numeric
31	Srv-diff-host-rate	Percent of connections to dierent hosts	Numeric
32	Dst-host-count	Count for destination host	Numeric
33	Dst-host-srv-count	Srv-count for destination host	Numeric
34	Dst-host-same-srv-rate	Same-srv-rate for destination host	Numeric
35	Dst-host-diff-srv-rate	Diff-srv-rate for destination host	Numeric
36	Dst-host-same-src-port-rate	Same-src-port-rate for destination host	Numeric
37	Dst-host-srv-diff-host-rate	Diff-host-rate for destination host	Numeric
38	Dst-host-serror-rate	Serror-rate for destination host	Numeric
39	Dst-host-srv-serror-rate	Srv-serror-rate for destination host	Numeric
40	Dst-host-rerror-rate	Rerror-rate for destination host	Numeric
41	Dst-host-srv-rerror- rate	Srv-serror-rate for destination host	Numeric

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

Before applying the feature selection and classification algorithm to the dataset, the train and test dataset data must be pre-processed and the values of all features converted to numeric data type. The features 2, 3, and 4 are symbolic and are converted to numbers. Feature 2 in the train and test dataset has three values tcp, udp and icmp which are converted to 0, 1, and 2 respectively. Feature 3 in the train dataset has 70 different types that are replaced with values of 0 to 69. The test dataset does not contain 6 cases of train dataset and for common cases the data type conversion is similar to the train dataset. Feature 4 of the train and test dataset has 11 different items that are replaced in both datasets with values of 0 to 10. Other features that are Boolean and numeric remain the same.

B. Performance Metric

In this section, we introduce the rates of accuracy, detection, false alarms, Precision and Recall which are widely used to evaluate the performance of intrusion detection. These parameters can be calculated by the confusion matrix as shown in Table3 [22].

TABLE III. CONFUSION MATRIX

Actual \ predicted	Normal	Attacks
Normal	TN	FP
Attacks	FN	TP

TP: The number of attacks which are correctly classified attacks
TN: The number of normal behaviors which are correctly classified normal
FP: The number of normal behaviors which are falsely classified attacks
FN: The number of attacks which are falsely classified normal

- ***Accuracy*:** Indicates how many percent of the total dataset is classified correctly. The following equation shows how to calculate accuracy:

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

- ***Detection:*** detection rate in the field of intrusion detection indicates what percentage of the data related to the attack classes are properly classified.

$$Detection\ rate = \frac{TP}{TP+FP}$$

- ***False Alarm rate:*** represents the percentage of cases where a normal behavior is incorrectly identified as an attack.

$$False\ Alarm = \frac{FP}{TN+FP}$$

- ***Precision:*** Shows what percentage of predictions is correct for an attack class.

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

- ***Recall:*** indicates what percentage of the total data in a class is classified correctly.

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

C. Results

This section provides experimental results before and after proposed feature selection.

TABLE IV. PROPOSED FS PERFORMANCE ON C4.5 DECISION TREEAND KNN CLASSIFIER ON TRAIN DATASET

	All features		Proposed FS	
classifier	**decision tree**	**KNN**	**decision tree**	**KNN**
Accuracy	99.74	99.81	99.91	99.91
Detection rate	0.996	0.996	0.998	0.999
False Alarm rate	0.0013	0.0004	0.0005	0.00005

TABLE V. PROPOSED FS PERFORMANCE ON C4.5 DECISION TREEAND KNN CLASSIFIER ON TEST DATASET

	All features		Proposed FS	
classifier	**decision tree**	**KNN**	**decision tree**	**KNN**
Accuracy	82.60	73.54	91.60	79.35
Detection rate	72.01	55.93	85.36	65.21
False Alarm rate	0.033	0.031	0.0014	0.019

Tables 6 and 7 show the Recall and Precision parameters on the test dataset before and after feature selection.

TABLE VI. RECALL PARAMETER

classifier	Features	Normal	Dos	Probe	R2L	U2R
	All features	96.61	94.07	80.75	9.12	9.25
	Proposed FS	99.86	98.22	83.73	54.23	54.25
	All features	96.83	74.81	56.79	8.34	20.5
	Proposed FS	98.05	81.55	85.34	7.13	40.5

TABLE VII. PRECISION PARAMETER

classifier	Features	Normal	Dos	Probe	R2L	U2R
	All features	76.89	97.17	75.89	95.49	41.25
	Proposed FS	86.42	98.07	90.25	99.57	74.32
	All features	0.66	91.34	71.13	88.75	36.5
	Proposed FS	71.22	98.22	75.1	95.79	73.68

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

Figure 2, 3 and 4 compare Accuracy rate, Detection Rate and False Alarm rate on decision tree and KNN classifier before and after feature selection respectively.

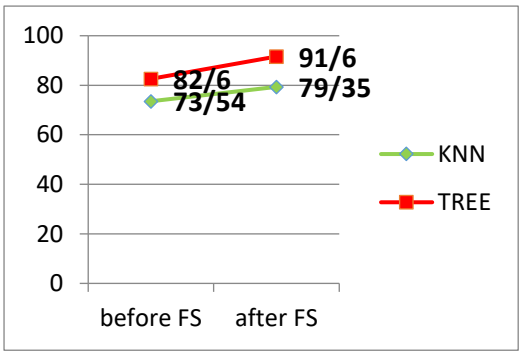

Fig. 2. Accuracy rate

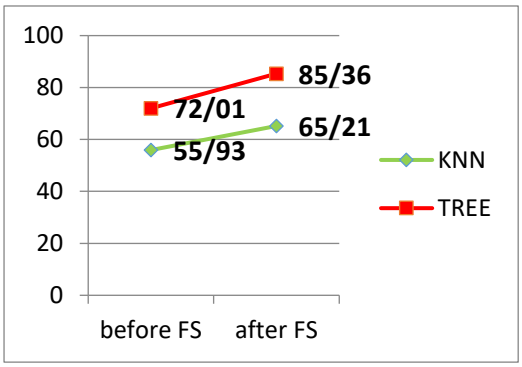

Fig. 3. Detection rate

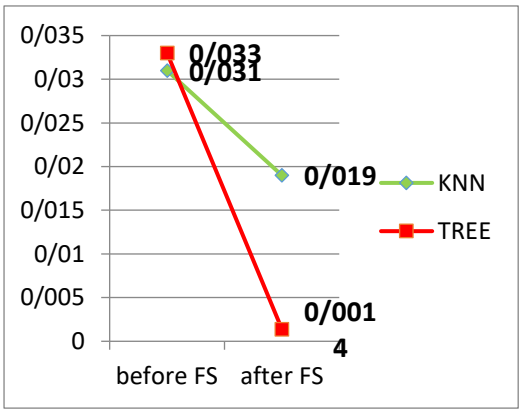

Fig. 4. False alarm rate

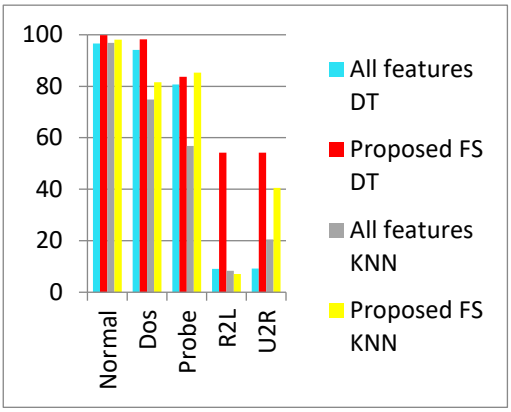

Fig. 5. Recall

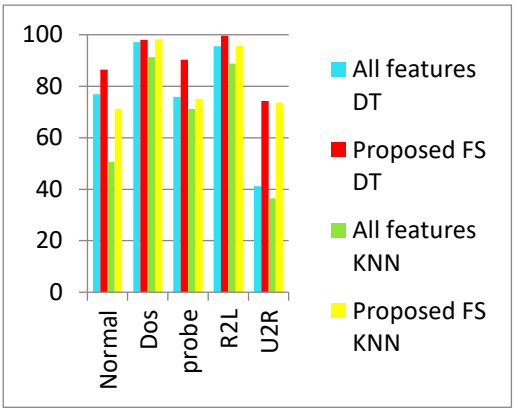

Fig. 6. Precision

V. CONCLUSIONS

This paper uses the wrapper feature selection method to improve the intrusion detection in computer networks. This feature selection method is classified into two general categories: sequential search algorithms and metaheuristic algorithms. The sequential search method increases exponentially by increasing the number of features, and therefore will result in high computational complexity. Using metaheuristic algorithms for features selection reduces computational complexity and increases accuracy. Genetic algorithm is a metaheuristic algorithm that is able to select the optimal subset with high accuracy. In proposed feature selection method, the goal was to minimize the C4.5 decision tree classification error on the NSL_KDD data set using the genetic algorithm. Using the feature vector obtained, training and testing operations were performed on the relevant datasets. Performance measurement parameters such as classification accuracy, detection rate and false alarm rate on the test dataset are 91.60, 85.36 and 0.0014, respectively, which show the improvement of the results. In this paper, it is assumed that if the feature selection algorithm is highly efficient and has selected the appropriate features, if this feature vector is used in another classifier, it will increase the accuracy of the classification and reduce the error. Therefore, the feature vector is used

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

in the KNN classifier and the values 65.21, 79.35 and 0.019 are obtained for the classification accuracy, detection rate and false alarm rate parameters that indicate the improvement of all parameters. In future research, other classification algorithms such as ANN, SVM and deep learning can be used and the results compared.

REFERENCES

[1] W.C. Lin, Sh.W.Ke and Ch.F.Tsai. "*CANN: An intrusion detection system based on combining cluster centers and nearest neighbors*." Knowledge-Based Systems ,Vol,٧٨. pp: 13-21, 2015.

[2] V.Jecheva and E.Nikolova. "*Some Clustering-Based Methodology Applications to Anomaly Intrusion Detection Systems.*" International Journal of Security and Its Applications, Vol.10, No.1, pp: 215-228, 2016.

[3] V.Jaiganesh, S.Mangayarkarasi and P.Sumathi. "Intrusion Detection Systems: *A Survey and Analysis of Classification Techniques*." International Journal of Advanced Research in Computer and Communication Engineering, Vol.2, Issu.4, pp: 1629-1635, April2013.

[4] P.Kalarani and S.S.Brunda. "*A Survey on Efficient Data Mining Techniques for Network Intrusion Detection System (IDS).*" International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 9, September 2014.

[5] J.Singh and M.J.Nene. "*A Survey on Machine Learning Techniques for Intrusion Detection Systems*." International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 11, November 2013.

[6] M.Ahmed, A.N.Mahmood and J.Hu "*A survey of network anomaly detection techniques*." Journal of Network and Computer Applications , Vol.60, pp: 19-31, January 2016.

[7] G.Chandrashekar and F.Sahin. "A *survey on feature selection methods."* Computers and Electrical Engineering, Vol.40, pp: 16-28, 2014.

[8] M.Hosseinzadeh Aghdam and P.Kabiri. "*Feature Selection for Intrusion Detection System Using Ant Colony Optimization."* International Journal of Network Security, Vol.18, No.3, pp: 420-432, May 2016.

[9] Y.Wan, M.Wang, Z.Ye and X.Lai. "A *feature selection method based on modified binary coded ant colony optimization algorithm."* Applied Soft Computing, Vol.49, pp: 248-258, 2016.

[10] Monowar H. Bhuyan, D.K. Bhattacharyya and J.K. Kalita. "*A multi-step outlier-based anomaly detection approach to network-wide traffic."* Information Sciences, Vol.348, pp: 243-271, 2016.

[11] Soo-Yeon Ji, Bong-Keun Jeong, Seonho Choi and Dong Hyun Jeong. "A *multi-level intrusion detection method for abnormal network behaviors."* Journal of Network and Computer Applications, Vol.62, pp: 9-17, 2016.

[12] Sakchi Jaiswal, Khushboo Saxena, Amit Mishra and Shiv K. Sahu. "*A KNN-ACO approach for intrusion detection using KDDCUP 99 dataset."* IEEE 3rd International Conference on Computing for Sustainable Global Development (INDIACom)**,** 16-18 March 2016.

[13] Seyed Mojtaba Hosseini Bamakan, Huadong Wang, Tian Yingjie and Yong Shi. "*An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization."* Journal of Neurocomputing, Vol.199, pp: 90-102, 2016.

[14] K.S.Desale and R.Ade. "*Genetic Algorithm Based Feature Selection Approach for Effective Intrusion Detection Systems."* IEEE International Conference on Computer Communication and Informatics (*ICCCI*), January 2015**.**

[15] V.K. Kshirsagar, S.M.Tidke and S.Vishnu. "*Intrusion Detection System using Genetic Algorithm and Data Mining: An Overview."* Science and Information Conference (SAI), Vol.1, Issue.4, pp: 91-95, August 2014**.**

[16] U.Patil, R.Gunjal, A.Gadhem, R.Kulkarni and S.Mandlik. *"NETWORK INTRUSION DETECTION & PREVENTION SYSTEM USING FUZZY LOGIC AND GENETIC ALGORITHM**."*** International Conference On Emerging Trends in Engineering and Management Research, pp: 105-112, March 2016.

[17] M.Kumar, M. Hanumanthappa and T. V. Suresh Kumar. "*Intrusion Detection System Using Decision Tree Algorithm."* IEEE 14th International Conference on Communication Technology (ICCT), pp: 629-634, November 2012.

[18] LONGJIE LI, Huadong Wang, YANG YU, SHENSHEN BAI, YING HOU and XIAOYUN CHEN. "*An Effective Two-Step Intrusion Detection Approach Based on Binary Classification and k -NN."* Journal of Underwater Wireless Communications and Networking, Vol.6, pp: 12060 - 12073, 2017.

[19] A.A.Aburomman and M.B.Ibne Reaz. "*A novel SVM-PSO ensemble method for intrusion detection system."* journal Applied Soft Computing, Vol.38, pp: 360-372, 2016.

[20] L.Dhanabal and S.P. Shantharajah. "*A Study on NSL_KDD Dataset for Intrusion Detection System Based on Classification Algorithms."* International Journal of Advanced Research in Computer and Communication Engineering, Vol.4, Issue.6, pp: 446-452, June 2015.

[21] E.Popoola and A.Adewumi. "*Efficient Feature Selection Technique for Network Intrusion Detection System Using Discrete Differential Evolution and Decision Tree."* International Journal of Network Security, Vol.19, No.5, pp: 660-669, September 2017.

[22] YANQING YANG, KANGFENG ZHENG, BIN WU, YIXIAN YANG and XIUJUAN WANG. "*Network Intrusion Detection Based on Supervised Adversarial Variational Auto-Encoder with Regularization."* Journal of IEEE Access, Vol.8, pp: 42169 - 42184, February 2020.

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173