Detecting Cyber Attacks with High-Frequency Features using Machine Learning Algorithms

Ahmet Nusret Özalp[1], Zafer Albayrak[2]

[1] Department of Computer Engineering, Karabuk University, Karabük, Turkey, ahmetnusretozalp@karabuk.edu.tr

[2] Department of Computer Engineering, Sakarya University of Applied Sciences, Sakarya, Turkey, zaferalbayrak@subu.edu.tr

Abstract: In computer networks, intrusion detection systems are used to detect cyber-attacks and anomalies. Feature selection is important for intrusion detection systems to scan the network quickly and accurately. On the other hand, analyzes performed using data with many attributes cause significant resource and time loss. In this study, unlike the literature studies, the frequency effects of the features in the data set are analyzed in detecting cyber-attacks on computer networks. Firstly, the frequencies of the features in the NSL-KDD data set were determined. Then, the effect of high-frequency features in detecting cyber-attacks has been examined with the widely used machine learning algorithms of Random Forest, J48, Naive Bayes, and Multi-Layer Perceptron. The performance of each algorithm is evaluated by considering Precision, False Positive Rate, Accuracy, and True Positive Rate statistics. Detection performances of different types of cyberattacks in the NSL-KDD dataset were analyzed with machine learning algorithms. Precision, Receiver Operator Characteristic, F1 score, recall, and accuracy statistics were chosen as success criteria of machine learning algorithms in attack detection. The results showed that features with high frequency are effective in detecting attacks.

Keywords: Attribute selection;Cyberattacks; Machine Learning; IDS;NSL-KDD; Anomaly detection

1 Introduction

The purpose of intrusion detection systems is to predict the attacks like infiltration, attack and malware in advance. From the security of information systems perspective, connection protocol bugs must be eliminated while the connection interfaces of network devices must be configured correctly. The detection of attacks is ensured by monitoring network as well as malicious traffic with port scans. For such a purpose, incoming and outgoing network packet information is watched over the network traffic. With the information received,

data is collected to detect suspicious connections. With active and passive scans, vulnerabilities of IP address blocks, open ports, operating system information, running services, active devices and active hosts are discovered accordingly.

In general, three methods are used to detect cyber-attacks. These are signature-based attack detection, anomaly-based attack detection and hybrid-based attack detection systems. In the signature-based detection method, each attack is recorded by creating a dictionary (wordlist) with a uniquely defined signature. Each newly detected attack is stored in this dictionary. Thus, a defence system is formed upon known and discovered attacks [1]. The anomaly detection method evaluates whether there is an unusual situation or not by taking information packets from the traffic on the network. If an abnormal situation is detected, then the intrusion prevention system is activated. Anomaly detection-based systems can detect attacks that signature-based detection systems cannot detect. In order to increase detection success, hybrid systems have been developed by combining these two approaches. According to their usage areas, hybrid intrusion detection systems can be divided into two parts. The first part is an anomaly based hybrid intrusion detection system while the second one is a parallel-based intrusion detection system. Regarding intrusion detection systems, applications performed with machine learning (ML), data mining and deep learning (DL) algorithms are available in the literature. Besides, various data mining techniques are also used to detect abnormal conditions in network traffic [2-4]. In the classification of traffic, the focus has been on machine learning techniques [5-8]. Performance values such as accuracy, positive accuracy rate and detection time have been tested in intrusion detection applications using machine learning techniques [9-11]. In the detection of attacks, machine learning techniques provide higher accuracy over network traffic. The results showed that intrusion detection approaches using machine learning algorithms provide higher success compared to other methods [8, 12]. The effects of scanning methods on intrusion detection systems have been provided in Table 1. In addition, the access control lists of the packets sent and the packets returned through the firewall have been determined. The obtained information constitutes an important parameter for attacks so that checklists are created against port scanning attacks while the firewall is prompted to correctly detect port scanning operations. Devices that perform routing and filtering processes bypass certain source ports. Special rules are obtained for unwanted ports that prevent unauthorized access. According to these rules, data are collected on the network traffic. The status of the traffic on the network is monitored via the amount of collected data. If an abnormal situation is detected, then the determination of the attack method is requested. In network-based intrusion detection systems, fuzzy set theory [4, 13, 14], artificial neural networks [6, 10, 15], ML [14, 16, 17], and DL techniques are used to detect links that contain anomalies [18-19]. Log files and datasets for analysis are the main components for the studies to be conducted. While detecting anomalies with real-time packet analysis, it is difficult to determine the parameters such as performance and accuracy [2]. Datasets are used in cases such as excessive energy use and memory

insufficiency in devices [7, 18]. They are also used as training and test data in studies where anomaly detection is performed with deep learning and ML algorithms. In such a case, the trained data precisely detect the attacks in real-time and provide information regarding the measures to be taken. All information added to the training data needs to be analyzed and controlled [10, 20].

Table 1
Network scanning attacks

Scanning Techniques	***Packet Sent***	***Port Open Close Detection***	***Returned Packet***	***Three-way Handshake***	***IDS Firewall Check***
TCP Connect/Full Open Scan	TCP	Yes	RST	Yes	Yes
Stealth Scan/Half-open Scan	TCP	Yes	RST	Yes	Yes
Inverse TCP Flag Scanning	FIN,URG, PSH	Yes	RST	Yes	Yes
Xmas Scan (Xmas Scanning)	IN,URG, PSH,TCP	Yes	Inverse TCP	Yes	Yes
ACK Flag Probe Scanning	TCP /ACK	Yes	RST	Yes	Yes
IDLE/IPID Header Scan	TCP,SYN	Yes	RST/SYN ACK/ RST	Yes	Yes

Thus, learning is provided with the tagged data. Increases in the number of users and devices, as well as difficulties in detecting real-time attacks cause hardware inadequacies and cause higher costs in detecting attacks by devices. In this study, more effective detection of cyber-attacks by attribute selection is proposed as it contributes to more effective cyber-attacks detection with high-frequency feature selection in datasets carrying attack information.

In this study, the main contribution of our study to the literature is the analysis of both anomaly-based attacks in the network and DDOS, U2R, R2L attacks with high classification success machine learning algorithms. Unlike previous studies, high-frequency features were determined as a result of the sequencing, and the detection rates of the attacks were analyzed by machine learning algorithms.

• 41 features with dataset were first dimensioned by using One-R, Chi-square (Chi-S), Correlation-Based Self-Attribute Selection (CBS), Symmetrical Uncertainty Coefficient (SUC), Gain Rate (GR), Information Gain (IG) selection methods. Unlike the studies in the literature, the frequencies of the features to be used in classification were determined. Then, the effects of high-frequency features in detecting different attacks were examined. In particular, 4, 5, 6, 29 and 30 valued attributes were effective in detecting anomalies, while 3, 4, 5 and 6 valued attributes were found to be effective in detecting DoS attacks.

• Feature vectors were classified by using Random Forest, J48, Naive Bayes, and Multi-Layer Perceptron algorithms. For the classification, accuracy, time, positive correct rate, and positive false rate were considered for the performance criteria of the algorithms. Besides, the effect of the attributes was provided while calculating the performance criteria of five different attacks.

• The performances of machine learning algorithms were compared according to the criteria of Precision (P), False Positive Rate (FPR), True Positive Rate (TPR), Accuracy (Acc) according to the high-frequency attributes.

The content of the article is organized as follows. In Section 2, information about the studies on the subject is provided. Feature selection methods are explained in Section 3. In Section 4, information about the machine learning model approaches used in intrusion detection is expressed as well as the classification algorithms used in the study. In Section 4, the results of the analysis are also evaluated while examining the effects of parameters on anomaly detection in computer networks. Finally, the results are summarized in Section 5.

2 Background

Most of the approaches such as fuzzy logic and data mining in detecting cyber-attacks have not yielded the desired results in larger datasets. As there are many attributes in the data collected from the computer network, the classification and detection of attacks cause a waste of time [3]. On the other hand, the information contained in the attributes is important for the accuracy of the classification. If the number of attributes is low, then the classification quality decreases while the error rate in the detection of attacks increases due to the generalizations. Besides, data processing time increases and real-time attacks become difficult to detect if the number of attributes is high. In attack detection systems, attribute sizing operations in the dataset are observed to decrease the attack detection time and increase the accuracy [4, 5]. Some studies perform machine learning and deep learning approaches in attack detection systems. The most recent ones are listed in Table 2 for to features of attribute selection and dataset usage as well as machine learning and deep learning algorithms. The current study is also compared with the existing studies according to the same criteria. Anomaly detection studies come to the forefront in studies using data-based techniques in detecting attacks [21, 22]. In machine learning algorithms, the NSL-KDD dataset is preferred due to its high number of attributes and its reliability in attack detection scenarios [17, 21]. Attribute numbers and selection methods are important parameters in detecting anomalies in machine learning algorithms [23, 24]. As a result of the classification according to the number of attributes selected, anomaly detection percentages between 97% and 99% were obtained [25, 26]. According to the analysis of 41 attributes in the proposed DNN approaches, the trained data was

determined to be insufficient due to the increase in the number of classes [5]. The above-mentioned features reveal the difficulties of collecting packets from network traffic in real-time attack detection as well as their performance reduction effect [6-20, 27-30]. In this study, the attribute selection performance was examined over the NSL-KDD dataset as the first step. Machine learning algorithms were preferred for classification algorithms after considering fuzzy logic, data mining, machine learning, and deep learning algorithms. Machine learning algorithms tend to represent high performance according to criteria such as accuracy, precision, and time in classification for the attributes selected on high-dimensional datasets [16, 20, 31, 32]. As the size of the data increase, the difficulties in interpreting the data reveal new approaches such as deep learning [23, 28, 33, 34]. In machine learning, it is desirable to store, change, and process the data in a suitable format to make it meaningful. The data are converted to matrix format and processed in tables before the estimation is performed. When deep learning approaches are examined, an appropriate model is designed [20, 29, 31]. By forming a model with the existing parameters, the suitability of the available data for the model is examined. Deep learning provides successful results in areas such as natural language processing, anomaly detection, and pattern recognition. In order to apply deep learning, the problem must be defined correctly as the first step. The mathematical model of the problem can be created while the improvement in the system can thus be observed by applying the relevant techniques. The dataset of this study has 24 different network attack examples in 4 categories. DoS (denial of service) is the name given to attacks to prevent network access. Probe (search) is defined as the scanning of IP and ports to detect vulnerabilities in the target. In R2L (remote to local), the attackers do not have the privilege to log in but can send the packet to the destination.

Table 2
Comparison with related research work

Study	***Attribute Selection***	***Machine Learning Approach***	***Deep Learning Approach***
Xin et al. [6]	No	Yes	Yes
Da Costa et al. [7]	Yes	No	No
C.bouni et al. [8]	Yes	Yes	Yes
Berman et al. [9]	Yes	Yes	Yes
Mazini M. et al. [10]	No	Yes	Yes
Sultana et al. [11]	No	Yes	No
Ferrag et al. [12]	No	Yes	Yes

On the U2R (user to root), they are able to monitor password entries to gain aggressive access. Even access right in standard user mode is possible, authorized users try to access. The 41 attributes in the dataset can be evaluated individually as well as under 4 categories according to the attack types.

Table 3
Comparison of attribute selection with relevant research studies

Author(s)	***Datasets***	***Approaches***	***Attribute selection type***
S.Thaseen et al. [14]	NSL-KDD	Weighted majority voting	Chi-S
Kasongo et al. [15]	NSL-KDD, UNSW-NB15	Two-stage ensemble	CBS
Mazini et al. [10]	NSL-KDD, ISCX 2012	Ada boost, Naïve Bayes	Bee colony
Verma et al. [16]	Private	Boosted tree,NB	-
Pham et al. [17]	NSL-KDD	Bagging,J48	GR
Aljawarneh et al. [18]	NSL-KDD	Majority voting, MLP	IG
Zaman et al. [19]	Kyoto 2006+	Majority voting	Information entropy
Al-Jarrah et al. [20]	NSL-KDD, Kyoto+	Random forest, Naïve Bayes	-
Vigneswaran et al. [21]	KDD Cup99	Random forest	-

These are the attacks made according to Transmission Control Protocol (TCP) connection characteristics, time-tagged attacks of two seconds, attacks lasting more than two seconds, and attack attributes based on content information [15]. The dataset is used in many academic studies, especially because of its high potential for anomaly detection and detection of new attacks. In literature studies have been conducted with this dataset using algorithms such as Support Vector Machine (SVM), K-Means, Random Forest, and J48 [15, 37]. Table 3 lists the studies conducted according to the methods used in attribute selection. Considering other methods, information gain was also preferred in this study. The reason behind this preference is the decision making capability in more diverse datasets although it does not have much data in the information acquisition method [11, 35, 40-43]. In the case of larger datasets, the need of being supported by other selection methods is required [21, 26, 35, 38, 39, 44-45].

3 Attribute Selection

For the attribute selection procedure, 10 attributes for NSL-KDD are selected according to the sorting criteria. Then, the selected attributes are classified accordingly and transferred to the model. The model suggested in this study is presented in Figure 1. A total number of 10 attributes were selected for NSL-KDD

by reducing the received datasets into subsets and sequencing them with attribute selection methods. As presented in Figure 1, the size of the training set should be redefined and brought into the appropriate evaluation range. In this process, the rate of gain, correlation-based attribute selection, information gain, chi-square, symmetric uncertainty coefficient, and One-R are selected as the attribute determination method. Since the best attributes are determined at this stage, the importance of this stage is inevitable. In particular, reducing the size of the collected data for anomaly detection reduces the burden during attack detection.

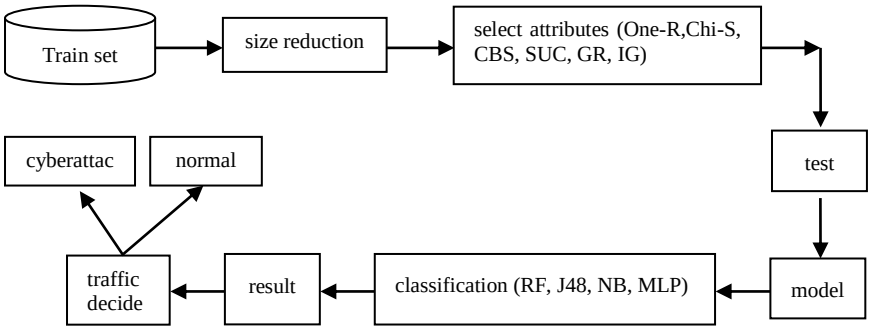

Figure 1
The model structure

The success of the system increases with the attributes obtained from known attack types. The received dataset is divided into subsets by methods such as random and complete search. Thus, a sub-dataset for evaluation is formed. Depending on the selection, dependent or independent criteria are determined. During this stage, the process continues until enough subsets are formed to determine the best subset. Since there is uncertainty, entropy is used during attribute selection. The following criteria ought to be considered for the selection of the attributes.

3.1 Correlation-Based Self-Attribute Selection (CBS)

CBS is based on determining clusters that are not directly related to each other since it has a filtering logic. The low correlated attributes are eliminated and the data with high frequencies are used by the following formula [19].

$$M_s = \frac{rfc}{\sqrt{l + k(k-1)rff}} \tag{1}$$

M_s= Merit value of the subset S with k features

rcf= Correlation between the class tag and the associated attribute

rff= Correlation of attributes.

3.2 Chi-square (CS)

CS is a statistical method where initial values observed by classes are calculated based on X2 statistics. In the next step, a selection is performed according to the number of attributes minus 1 in the dataset depending on its importance status. If the expected frequency value matches, x2 approaches zero while it indicates incompatibility otherwise according to the following formula [20].

$$x^2 = \sum_{i=1}^{n} \frac{(o-e)^2}{e} \tag{2}$$

n: number of attributes in the dataset

o: Observed frequency value for the ith attribute

e; Expected frequency value for the ith attribute.

3.3 One-R (1-R)

Each attribute in the dataset allocated for training is classified according to the determined rule. Depending on the error rate, sorting is performed according to the most frequently encountered attributes. Defines a rule for the entire prediction, each value of the prediction made, and the frequency of each value in the class is counted. The class with the highest frequency is determined while the corresponding prediction is added to the rule. The total error is calculated and the one with the lowest error rate is selected [23].

3.4 Symmetrical Uncertainty Coefficient (SUC)

In order to eliminate and eventually normalize the negative cases arising in the information gain method, the entropies of the attributes sampled as X and Y are added where the SUC is defined as [24].

$$\text{Coefficient} = \left(2 \frac{\text{IG}}{\text{H(y)} + \text{H(x)}} \right) \tag{3}$$

3.5 Information Gain (IG)

Information gain is a way of normalizing the negative parts in symmetrical uncertainty gain and is based on entropy. The X property and the Y property vary depending on their respective values (Eq. 4). The biggest drawback of this method is that it may make decisions in favor of more diverse datasets although it does not have much data [25].

$$IG = H(y) - H(y \| x) \tag{4}$$

By measuring the information gain according to the class, the property value is examined.

$$IG(class, attribute) = H(class) - H(class | attribute) \tag{5}$$

3.6 Gain Rate (GR)

Gain rate is used to normalize the information gained method to minimize the resulting diversity by [26].

$$GR = \left(\frac{IG}{H(x)} \right) \tag{6}$$

4 Experimental Work

In this study, 6 different attribute determination methods of the NSL-KDD dataset are used. 20% of the data set features were selected. This ratio was also taken into account during the determination of the training and test dataset. In the studies conducted in the literature, no specific reason for the number of selected features has been revealed [25, 44]. In this study, the 10 most successful attributes were selected according to their performance order for each method. Table 4 lists the attributes chosen as the basis for ordering. When the studies conducted with the NSL-KDD dataset are examined [32], it was observed that the number of selected attributes, attribute selection method, and classification approaches are different. 10 attribute names, their numbers in the dataset, and their frequencies obtained as a result of the attribute selection methods are shown in Table 5. The list in the table indicates that the frequencies of the attributes numbered 4, 5, 6, 12, 29, and 30 are high. At the end of the feature selection in Table 5, features with high frequency are observed. As a result of the feature selection procedure for 6 different methods, the order of each attribute was determined. The features with the highest frequency in this ranking are provided in Table 6. Here, flag data checks the connection status while src-byte and dst-byte check the link status of

source and destination points during the connection. Diff-srv-rate and same-srv-rate attributes represent the connection status of the attacker to the same point. These parameters were obtained to be effective in detecting 5 different attack types in the dataset as well as detecting anomalies in the network due to their high frequency. These attribute subsets were analyzed using 4 different classification algorithms such as Naive Bayes, J48, Multi-Layer Perceptron, and Random Forest respectively [46]. No specific intrusion detection reference measure is obtained as it is decided by the classification and modeling of current attack types. In this study, precision, false-positive rate, accuracy, and true-positive rate were considered the criteria for detecting the attacks. The common feature of the high-frequency attributes is their usage possibility in the detection of DoS attacks and anomaly status in the network. The similarity rate of the features used in the detection of the other three types of attacks is obtained to be 73%.

Table 4

NSL-KDD Dataset results obtained with attribute selection methods and their ranking

One-R			**Correlation-Based Self-Attribute Selection**		
Attributes			**Attributes**		
Ranked	**Number**	**Name**	**Ranked**	**Number**	**Name**
96.374	5	src_bytes	0.747	3	service
91.558	3	service	.0725	5	src_bytes
90.900	6	dst_bytes	0.695	12	logged_in
88.090	4	flag	0.692	4	flag
87.380	30	diff_srv_rate	0.691	6	dst_bytes
87.324	29	same_srv_rate	0.634	29	same_srv_rate
85.426	34	dst_host_s_sr_rate	0.595	30	diff_srv_rate
85.010	33	dst_host_srv_count	0.576	25	serror_rate
83.920	35	dst_hst_di_srv_rate	0.563	26	srv_serror_rate
82.947	12	logged_in	0.531	33	dst_host_srv_count
Gain Ratio Feature			**Chi-Square**		
Attributes			**Attributes**		
Ranked	Number	Name	Ranked	Number	Name
0.418	12	logged_in	109.922	5	src_bytes
0.373	26	srv_serror_rate	93.032	3	service
0.339	4	flag	87.820	6	dst_bytes
0.332	25	serror_rate	75.735	4	flag
0.332	39	dst_host_srv_s_rate	74.897	30	diff_srv_rate
0.267	30	diff_srv_rate	73.850	29	same_srv_rate
0.264	38	dst_host_serror_rate	69.215	33	dst_host_srv_count
0.258	6	dst_bytes	67.900	34	dst_host_s_sr_rate
0.231	5	src_bytes	62.343	35	dst_hst_di_srv_rate
0.224	29	same_srv_rate	60.430	12	logged_in

Symmetrical Uncertainty Coefficient			**Information Gain Attribute**		
Attributes			**Attributes**		
Ranked	**Number**	**Name**	Ranked	Number	Name
0.411	12	logged_in	0.816	5	src_bytes
0.411	4	flag	0.671	3	service
0.377	6	dst_bytes	0.633	6	dst_bytes
0.367	26	srv_serror_rate	0.519	4	flag
0.362	39	dst_host_srv_s_rate	0.518	30	diff_srv_rate
0.360	25	serror_rate	0.509	29	same_srv_rate
0.360	5	src_bytes	0.475	33	dst_host_srv_count
0.353	30	diff_srv_rate	0.438	34	dst_host_s_sr_rate
0.320	38	dst_host_serror_rate	0.410	35	dst_hst_di_srv_rate

Table 5
Attribute frequencies

Attribute Number	***Attribute Name***	***Frequency***
3	service	4
4	flag	6
5	src-bytes	6
6	dst-bytes	6
12	logged-in	5
25	serror-rate	3
26	srv-serror-rate	3
29	same-srv-rate	6
30	diff-srv-rate	6
33	dst-host-srv-count	3
34	dst-host-same-srv-rate	3
35	dst-host-diff-srv-rate	3
38	dst-host-serror-rate	3
39	dst-host-srv-serror-rate	2

Table 6
High-frequency features

No	***Attribute name***	***Description***	***Sample Data***
4	Flag	Connection status Normal or Error	SF
5	src-bytes	Number of data bytes transferred from source to destination in a single connection	491
6	dst-bytes	Number of data bytes transferred from destination to source in a single connection	1
29	same srv-rate	Percentage of connections to the same service among the connections aggregated	1
30	diff-srv-rate	Percentage of connections to different services among the connections collected	1

4.1 Performance Criteria

The accuracy of the classification process is measured by the "Confusion Matrix". This matrix provides an understanding of the probability outcomes in classification. If there is a dual classification such as anomaly detection, the labelling is done as normal and abnormal. Four conditions arise in binary guessing. True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). While measuring the accuracy of the model put forward accordingly, True Positive Rate (TPR) and False Positive Rate (FPR) are used.

Precision (P): Precision refers to the possibility of making an accurate estimate with the data obtained and defined as [33],

$$\mathrm{P} = \left(\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \right) \tag{7}$$

False Positive Rate (FPR): It is the rate of classifying the obtained data with an erroneous approach formulated as [34],

$$\mathrm{FPR} = \left(\frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \right) \tag{8}$$

True Positive Rate (TPR): It is the number of correct samples included in the positively grouped class defined as, [36].

$$\mathrm{TPR} = \left(\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \right) \tag{9}$$

Accuracy (Acc): Of the total sample in the dataset, it is the percentage of the correctly estimated sample formulated as [35],

$$\mathrm{Acc} = \left(\frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + TN + FP + \mathrm{FN}} \right) \tag{10}$$

Receiver Operator Characteristic (ROC): ROC is used to calculate cost sensitivity in classification processes. It is obtained by drawing the curve between False Positive Rate and True Positive Rate in the detection of anomalies. In this way, the performance of the algorithm used as a classifier is compared between error costs and class distributions. The area under the curve shows the accuracy of the model estimation to be obtained as a result of the classification [33].

F-1 Score: It is an accuracy parameter for the test. It is calculated according to the sensitivity (P) and recall [37].

$$\mathrm{F1\text{-}Score} = \left(\frac{2\mathrm{TP}}{2\mathrm{TP} + FP + \mathrm{FN}} \right) \tag{11}$$

Recall: It is the ratio of positive correct predictions to samples in the positive grade [38].

$$\text{Recall} = \left(\frac{\text{TP}}{\text{TP} + \text{FP}} \right) \tag{12}$$

4.2 Machine Learning Approaches with Intrusion Detection Systems

While optimization provides a way to minimize loss functionality for deep learning and machine learning, the goal of both methods is fundamentally different. The former is mainly concerned with minimizing a goal while the latter is concerned with finding a suitable model when a limited amount of data is available. The purpose function of the optimization algorithm is to reduce the training error. When it is a loss function, it is usually based on the training dataset. However, the purpose of statistical inference is to reduce the generalization error. This indicates that machine learning and deep learning approaches can be used in intrusion detection systems. Structurally, intrusion detection systems have to respond quickly to cyber-attacks. The use of machine learning algorithms in intrusion detection systems can be evaluated by using classification method, scaling method, and both classification and scaling methods. The attributes selected from the dataset are processed by machine learning algorithms during the classification stage. The techniques used in the classification stage provide the appropriate model creation. The studies about deep learning indicate that it needs improvement although it provides successful results especially in detecting anomalies [23]. Machine learning approaches on the other hand provide successful results in detection and prevention.

4.2.1 Random Forest Algoritm (RF)

Decision trees form a tree structure for classification models. Information gain and entropy metrics are important parameters in this algorithm. First, a decision tree is created with the learning set. In the next step, each new input data is determined as a class label. The Random Forest algorithm, which is also known as the decision tree classification, is the classification algorithm used to detect cyber-attacks. It is frequently used in anomaly detection, analysis of malware and vulnerability analysis in the detection of cyber attacks [26]. New branches are created by comparing each node that makes up the tree and the attributes divided into subsets while the leaves of the tree are expressed as a class. The biggest advantage of this algorithm over other algorithms is the presence of fewer parameters. It does not take unnecessary action against the abnormal data in the dataset so that it works with lower loads [28]. The classifier is defined as,

$$h(x,k), k = 1,2,...i \tag{13}$$

h: Classifier θk: random vector x: tree class tag

4.2.2 Naive Bayes Algorithm (NB)

The Naive Bayes algorithm is a Bayes' approach for classification where each attribute pair is processed independently. It evaluates the data independently. The aim is to have an equal effect on the result for each parameter. Structurally, it is a very simple and fast algorithm. It is one of the preferred methods for detecting cyber-attacks [26]. It can also provide results in a short time since it requires less training data for sampling points. NB reduces the problem of separator classes to find classes with conditional marginal densities. For this reason, representing the probability that a given sample is one of the possible target classes. Unless it contains inputs associated with each other, NB performs well against other algorithms.

$$P(c \mid x) = \frac{P(x \mid c) * P(c)}{P(x)} \tag{14}$$

P(c | X)=P(x_1 | c)*P(x_2 | c)*…*P(x_n | c)*P(c)

P(c):Class Prior Probability P(c | x):Posterior Probability

P(x):Predictor Prior Probability P(x | c):Likelihood

4.2.3 Multi Layer Perceptron (MLP)

Multi-Layer Perceptron is a feed-forward neural network, which consists of at least three layers as input, hidden, and output layers [27]. Except for input nodes, every node uses a non-linear activation function. It uses a feedback supervised learning technique for training. In this respect, it can be used in a non-linear system to distinguish the desired data.

$$f(x) = \sum_{i=1}^{m} (wi * xi) + b \tag{15}$$

m: The number of neurons in the previous layer w: random weight

x: input value b: random bias

4.2.4 J48 Algorithm

J48 is an algorithm developed by Ross Quinlan and considered the continuation of the ID3 algorithm. As it can create a decision tree, it is used as a statistical classifier in the structural sense [28]. In this classifier, a flowchart in the form of a tree model is created while the problem is tried to be solved based on prediction.

The nodes in the tree indicate the samples taken for entry while the leaves represent the estimates based on this entry.

5 Experiment and Results

After selecting the attributes of the dataset in detecting the attacks, the classification process was performed. The accuracy rate of classification according to attack types is expected to be high. In particular, the correctness of the classification in the detection of anomalies ensures correct interpretation of the features in the dataset. The algorithms listed in Table 7 were chosen due to their high classification rate. Successful results were obtained in P, FPR, Acc, and TPR percentages depending on the time in the anomaly detection with the selected features. Table 7 represents the anomaly classification algorithm performances conducted on the NSL-KDD dataset. NSL-KDD data set with 58630 anomalies and 67343 normal traffic data was used during training while 12833 anomaly data and 9711 normal data were used as test data. Random Forest, J48, Naive Bayes and MLP-CNN showed success rates of 99.76%, 98.45%, 93.34% and 91.34%, respectively, in the classifications made with the selected 10 features. The success in this classification rate was used to determine the attributes used for the detection of attacks. Classification results were evaluated according to P, FPR, Acc and TPR, by examining the criteria specified in the literature. Machine learning methods were compared using false alarm rate, accuracy and detection rate to detect anomalies in the network. It has been observed that the high rate of classification success also affects the success of the algorithm in detecting anomalies.

Table 7
Anomaly detection rates with J48, MLP, RF, and NB classification approaches concerning the obtained feature selections.

		J48 Algorithm					**Multi-Layer Perceptron (MLP)**				
		Correctly Classified Instances: 99.7817 %					**Correctly Classified Instances: 98.4354 %**				
Attribute & Methods		**Time (sec)**	**P %**	**FPR %**	**Acc %**	TPR **%**	**Time (sec)**	**P %**	**FPR %**	**Acc %**	**TPR %**
5,3,6,4,30,29,34,33, 35,12	**One-R**	3.34	76.86	4.23	93.56	93.52	19.65	69.29	4.17	90.86	83.50
12,26,4,25,39,30,38,6,5,29	**GR**	4.11	73.20	6.58	90.23	89.42	30.43	77.41	7.12	94.52	91.47
5,3,6,4,30,29,33,34, 35,12	**CS**	3.01	74.52	4.23	93.46	91.12	18.40	68.86	5.21	90.14	83.20
5,3,6,4,30,29,3	**IG**	3.05	74.56	4.74	93.20	90.36	19.97	68.27	5.74	91.01	82.98

3,34, 35,38											
12,4,26,6,39,25 ,5,30, 38,29	**SUC**	3.98	76.59	7.52	93.76	89.93	23.43	65.98	7.51	99.52	81.26
3,5,12,4,6,29,3 0,25, 26,33	**CBS**	5.43	74.86	7.41	95.47	88.63	22.50	75.58	6.27	90.74	80.37
All Attributes	-	6.78	73.47	4.69	90.45	98.45	45.86	72.41	4.12	91.38	91.34
		Random Forest (RF)					**Naive Bayes (NB)**				
		Correctly Classified Instances: 99.9174 %					**Correctly Classified Instances: 90.4178 %**				
Attribute & Methods		**Time (sec)**	**P %**	**FPR %**	**Acc %**	TPR **%**	**Time (sec)**	**P %**	**FPR %**	**Acc %**	**TPR %**
5,3,6,4,30,29, 34,33,35,12	**One-R**	2.90	78.65	3.23	94.95	95.43	5.20	75.38	5.89	91.58	89.36
12,26,4,5,39, 30,38,6,5,29	**GR**	4.11	74.82	5.43	91.23	91.23	4.78	71.24	8.23	90.56	88.26
5,3,6,4,30,29, 33,34,35,12	**CS**	3.01	76.78	3.21	94.65	93.54	3.97	72.28	5.27	90.26	89.78
5,3,6,4,30, 29,33,34,35,38	**IG**	3.05	76.71	3.28	94.89	93.87	3.98	72.56	5.23	90.29	89.87
12,4,26,6,39, 25,5, 30,38,29	**SUC**	3.98	78.42	6.40	95.54	91.20	4.21	74.36	8.54	90.56	88.25
3,5,12,4,6,29, 30,25,26,33	**CBS**	5.43	76.43	6.54	96.54	89.65	6.23	72.57	9.23	91.14	87.41
All attributes	-	6,78	75,43	3,45	95,45	99,76	8,30	70,29	4,69	91,45	93,34

Table 8

Performance of probe attack, U2R, R2L, and DoS attack types in machine learning classifiers

	Algorithms	***P***	***ROC***	***F1-Score***	***Re-call***	***Acc (%)***
Probe Attack	Multi-Layer Perceptron (MLP)	0.954	0.996	0.998	0.998	98.510
	Naive Bayes	0.986	0.976	0.961	0.971	90.398
	Random Forest	**0.999**	**1.000**	**1.000**	**1.000**	**99.952**
	J48	0.994	0.999	0.999	1.000	99.951
	Algorithms	***P***	***ROC***	***F1-Score***	***Re-call***	***Acc (%)***
User Root Attack	Multi-Layer Perceptron (MLP)	0.995	0.995	0.995	0.995	99.210
	Naive Bayes	0.999	0.949	0.961	0.943	88.859
	Random Forest	0.999	**0.998**	0.997	**0.998**	**99.859**
	J48	**1.000**	0.937	**0.998**	**0.998**	99.674
	Algorithms	***P***	***ROC***	***F1-Score***	***Re-call***	***Acc (%)***
Remote to Local Attack	Multi-Layer Perceptron (MLP)	0.997	0.996	0.992	0.992	99.814
	Naive Bayes	0.999	0.957	0.935	0.889	98.928

	Random Forest	**0.999**	**0.999**	**0.999**	**1.000**	**99.999**
	J48	0.998	0.995	0.998	0.999	99.997
	Algorithms	***P***	***ROC***	***F1-Score***	***Re-call***	***Acc (%)***
DoS Attack	Multi-Layer Perceptron (MLP)	0.954	0.841	0.948	0.998	95.752
	Naive Bayes	0.979	0.909	0.951	0.914	94.178
	Random Forest	**1.000**	**0.999**	**0.999**	**0.999**	**99.842**
	J48	0.995	0.667	**0.999**	**0.999**	99.774

From the results of experiments, it is seen that the number of features selected in the NSL-KDD dataset and the classification algorithm attacks affect the detection rate. Performance varies depending on the dataset size and the number of attributes selected. In previous studies, feature selection and the number of features were taken into consideration rather than high-frequency features. In previous studies, feature selection and the number of features were taken into consideration rather than high-frequency features. This situation was seen to directly affect the classification percentages of machine learning algorithms. With the attack detection study conducted with high-frequency features, the Random Forest algorithm was 1.7%; 0.97% of the J48 algorithm; 0.86% better results of NB algorithm, and 1.3% better results of MLP algorithm were obtained.

Conclusion

The results obtained in this study indicated that Random Forest Algorithm provides high performance in terms of classification and accuracy in the case of high-frequency features. Random Forest is followed by J48, NB, and MLP respectively. The most important feature identification function among datasets, which is one of the advantages of the Random Forest algorithm, has increased its success in attack analysis with the selection of high-frequency features. It has been observed that the success of the MLP algorithm used in linear functions in detecting cyberattacks is lower than other algorithms. When the features with high frequency are analyzed with machine learning algorithms, it is observed that especially the Random Forest algorithm produces 1.7% more accurate results.

References

[1] Meng, Weizhi, Wenjuan Li, and Lam-For Kwok, EFM: enhancing the performance of signature-based network intrusion detection systems using enhanced filter mechanism, Computers & Security 43 (2014), pp. 189-204

[2] Alabadi, Montdher, and Zafer Albayrak, Q-Learning for Securing Cyber-Physical Systems: A survey, In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), IEEE, 2020, pp. 1-13

[3] Nazir, Anjum, and Rizwan Ahmed Khan., Network Intrusion Detection: Taxonomy and Machine Learning Applications, Machine Intelligence and

Big Data Analytics for Cybersecurity Applications. Springer, Cham, 2021, pp. 3-28

[4] Dwivedi, Shubhra, Manu Vardhan, and Sarsij Tripathi., Distributed denial-of-service prediction on iot framework by learning techniques, Open Computer Science 10.1 (2020) pp. 220-230

[5] Alabadi, M., & Celik, Y. (2020, June) Anomaly detection for cyber-security based on convolution neural network: A survey. In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-14) IEEE

[6] Dua, Mohit, Attribute selection and ensemble classifier based novel approach to intrusion detection system, Procedia Computer Science 167 (2020) pp. 2191-2199

[7] Amiri, Fatemeh, et al, Mutual information-based feature selection for intrusion detection systems, Journal of Network and Computer Applications 34.4 (2011) pp. 1184-1199

[8] Akhter, A. F. M., Ahmed, M., Shah, A. F. M., Anwar, A., Kayes, A. S. M., & Zengin, A. (2021). A blockchain-based authentication protocol for cooperative vehicular ad hoc network. Sensors, 21(4) 1273

[9] Xin, Yang, Lingshuang Kong, Zhi Liu, Yuling Chen, Yanmiao Li, Hongliang Zhu, Mingcheng GAO, Haixia Hou, and Chunhua Wang, Machine learning and deep learning methods for cybersecurity, IEEE Access 6 (2018) pp. 35365-35381

[10] Da Costa, Kelton AP, et al, Internet of Things: A survey on machine learning-based intrusion detection approaches, Computer Networks 151 (2019) pp. 147-157

[11] Chaabouni, Nadia, et al., Network intrusion detection for IoT security based on learning techniques, IEEE Communications Surveys & Tutorials 21.3 (2019) pp. 2671-2701

[12] Berman, Daniel S., et al, A survey of deep learning methods for cyber security, Information 10.4 (2019) p. 122

[13] Dey, Samrat Kumar, and Md Rahman, Effects of machine learning approach in flow-based anomaly detection on software-defined networking, Symmetry 12.1 (2020) p. 7

[14] Otor, Samera Uga, et al., An improved bio-inspired based intrusion detection model for a cyberspace, Cogent Engineering 8.1 (2021) p. 1859667

[15] Alghamdi, Mohammed I., Survey on Applications of Deep Learning and Machine Learning Techniques for Cyber Security, International Journal of Interactive Mobile Technologies 14.16 (2020)

[16] Sultana, Nasrin, et al., Survey on SDN based network intrusion detection system using machine learning approaches, Peer-to-Peer Networking and Applications 12.2 (2019) pp. 493-501

[17] Ahmed, M., Moustafa, N., Suaib Ahther, A. F. M., Rezzak, I., Surid, E., Anwar, E., Shanen Shah, A.F.M & Zengin, A. (2021) A Blockchain-Based Emergency Message Transmission Protocol for Cooperative VANET. IEEE Transactions on Intelligent Transportation Systems (pp. 1-10)

[18] Revathi, S., and A. Malathi, A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection, International Journal of Engineering Research & Technology (IJERT) 2.12 (2013) pp. 1848-1853

[19] Mazini, Mehrnaz, Babak Shirazi, and Iraj Mahdavi, Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms, Journal of King Saud University-Computer and Information Sciences 31.4 (2019) pp. 541-553

[20] Aljawarneh, Shadi, Monther Aldwairi, and Muneer Bani Yassein., Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, Journal of Computational Science 25 (2018) pp. 152-160

[21] Nkenyereye, Lewis, Bayu Adhi Tama, and Sunghoon Lim, A Stacking-Based Deep Neural Network Approach for Effective Network Anomaly Detection, Cmc-Computers Materials & Continua 66.2 (2021) pp. 2217-2227

[22] Aggarwal, Preeti, and Sudhir Kumar Sharma, Analysis of KDD dataset attributes-class wise for intrusion detection, Procedia Computer Science 57 (2015) pp. 842-851

[23] Hosseini, Soodeh, and Hossein Seilani, Anomaly process detection using negative selection algorithm and classification techniques, Evolving Systems (2019) pp. 1-10

[24] Mahdavifar, Samaneh, and Ali A. Ghorbani, Application of deep learning to cybersecurity: A survey, Neurocomputing 347 (2019) pp. 149-176

[25] Zhiqiang, Liu, et al., A Three-Layer Architecture for Intelligent Intrusion Detection Using Deep Learning, Proceedings of Fifth International Congress on Information and Communication Technology. Springer, Singapore, 2021, pp. 245-255

[26] Sumaiya Thaseen, I., et al., An integrated intrusion detection system using correlation-based attribute selection and artificial neural network, Transactions on Emerging Telecommunications Technologies 32.2 (2021) e4014

[27] Altunay, H. C., Albayrak, Z., Özalp, A. N., & Çakmak, M. (2021, June) Analysis of anomaly detection approaches performed through deep learning methods in SCADA systems. In 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-6) IEEE

[28] Kasongo, Sydney M., and Yanxia Sun, Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset, Journal of Big Data 7.1 (2020) pp. 1-20

[29] Tang, Chaofei, Nurbol Luktarhan, and Yuxin Zhao, SAAE-DNN: Deep Learning Method on Intrusion Detection, Symmetry 12.10 (2020) pp. 1695

[30] Gwon, Hyeokmin, et al., Network Intrusion Detection based on LSTM and Feature Embedding., arXiv preprint arXiv: 1911.11552 (2019)

[31] Ring, Markus, et al., A survey of network-based intrusion detection data sets. Computers & Security 86 (2019) pp. 147-167

[32] Umamaheswari, K., Subbiah Janakiraman, and K. Chandraprabha., Multilevel Hybrid Firefly-Based Bayesian Classifier for Intrusion Detection in Huge Imbalanced Data, Journal of Testing and Evaluation 49.1 (2021)

[33] Milenkoski, Aleksandar, et al., Evaluating computer intrusion detection systems: A survey of common practices, ACM Computing Surveys (CSUR) 48.1 (2015) pp. 1-41

[34] Sabaz, F., & Celik, Y. (2018) Systematic Literature Review on Security Vulnerabilities and Attack Methods in Web Services. International Conference on Advanced Technologies, Computer Engineering and Science (pp. 821-825)

[35] Akter, M., Dip, G. D., Mira, M. S., Hamid, M. A., & Mridha, M. F., Construing attacks of internet of things (IoT) and a prehensile intrusion detection system for anomaly detection using deep learning approach, Springer In International Conference on Innovative Computing and Communications, 2020, pp. 427-438

[36] Ferrag, Mohamed Amine, et al., Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study, Journal of Information Security and Applications 50 (2020) pp. 102419

[37] Buczak, Anna L., and Erhan Guven, A survey of data mining and machine learning methods for cyber security intrusion detection, IEEE Communications surveys & tutorials 18.2 (2015) pp. 1153-1176

[38] Mendez Mena, Diego, and Baijian Yang., Decentralized Actionable Cyber Threat Intelligence for Networks and the Internet of Things, IoT 2.1 (2021) pp. 1-16

[39] Issa, A. S., & Albayrak, Z. CLSTMNet: A Deep Learning Model for Intrusion Detection, Journal of Physics: Conference Series (2021)

[40] Pham, Ngoc Tu, et al., Improving performance of intrusion detection system using ensemble methods and feature selection, Proceedings of the Australasian Computer Science Week Multiconference. 2018, pp. 1-6

[41] Al-Jarrah, O. Y., et al., Machine-learning-based feature selection techniques for large-scale network intrusion detection, 2014 IEEE 34th international conference on distributed computing systems workshops (ICDCSW) IEEE, 2014

[42] Said A. A., Çakmak M. & Albayrak, Z., 6th International Conference on Smart City Applications (SCA2021) (2021) (pp. 1133-1140)

[43] Karimipour, Hadis, and Henry Leung, Relaxation-based anomaly detection in cyber-physical systems using ensemble kalman filter, IET Cyber-Physical Systems: Theory & Applications 5.1 (2020) pp. 49-58

[44] Patil, Tina R., MSSS Performance analysis of naive bayes and J48 classification algorithm for data classification, Journal of Computer Science and Applications 6.2 (2013)

[45] Bhati, Nitesh Singh, and Manju Khari, A Survey on Hybrid Intrusion Detection Techniques, Research in Intelligent and Computing in Engineering. Springer, Singapore, 2021, pp. 815-825

[46] Azzaoui, Hanane, et al., Developing new deep-learning model to enhance network intrusion classification, Evolving Systems (2021) pp. 1-9