Uluslararası İleri Doğa Bilimleri ve Mühendislik Araştırmaları Dergisi
Sayı 7, S. 489-500, 10, 2023

Araştırma Makalesi

IJANSER
https://alls-academy.com/index.php/ijanser
ISSN: 2980-0811

International Journal of Advanced Natural Sciences and Engineering Researches
Volume 7, pp. 489-500, 10, 2023

Research Article

Network Anomaly Detection Using a Hybrid Approach of Machine Learning Algorithms

Feyza ÖZGER[1], Halit ÖZTEKİN[2]

[1]*Elektrik Elektronik Mühendisliği, Sakarya Uygulamalı Bilimler Üniversitesi, Sakarya*
[2]*Bilgisayar Mühendisliği, Sakarya Uygulamalı Bilimler Üniversitesi, Sakarya*

[1]*y205004009@subu.edu.tr*

(Received: 30 October 2023, Accepted: 30 November 2023)

REFERENCE: Özger F., Öztekin H. (2023). Network Anomaly Detection Using a Hybrid Approach of Machine Learning Algorithms. *International Journal of Advanced Natural Sciences and Engineering Researches*, 7(10), 489-500.

Abstract – Internet, while being of vital importance, has also brought along cyber attacks and threats. Detection systems in cybersecurity have gained importance to counter these threats. Systems like network anomaly detection can identify abnormal activities by learning normal network traffic. Hybrid models have shown high success in cyber attack detection. In tests conducted on the KDD Cup 1999 dataset, machine learning methods such as Decision Trees, Logistic Regression, Naive Bayes, Random Forest, and k-Nearest Neighbors have exhibited high accuracy levels. Two different hybrid feature selection methods, PCA + RFECV and RFECV + FS, were compared, and it was observed that feature selection plays a critical role in classification performance. These methods can enhance classification performance by reducing the dimensionality of the dataset and selecting meaningful features. This study emphasizes the importance of cybersecurity detection systems in minimizing the potential damage of digital attacks while safeguarding the information of individuals and organizations.

Keywords- Network Security, Supervised Learning, Machine Learning, Metaheuristic Algorithms, KDD Cup 1999.

I. INTRODUCTION

The rapid increase in internet usage in today's world has led to a proportional increase in cyber threats. This situation has made it mandatory for institutions and organizations to enhance their cybersecurity measures. In this context, detecting anomalous behaviors within a network, as initiated by cyber attackers, holds significant importance in terms of cybersecurity. The aim of this thesis is to investigate methods used for the detection of anomaly behaviors and to assess the performance of commonly used supervised machine learning algorithms on the relevant dataset, as well as to compare the performances of these algorithms. The results of this study will contribute to institutions and organizations in taking more effective measures for network security. Experiments were conducted on the KDD-CUP 99 dataset in relation to this topic.

Anomaly detection is the process of identifying data that deviates from normal behavior in a dataset and is typically accomplished through data analysis, machine learning, or statistical methods [1].

Anomaly detection methods can be classified based on various factors. Some of these factors include [2]:

Point Anomaly: A single data point in the dataset stands out as different from the others. This can encompass situations where the data point has a much higher or lower value compared to the others. For example, in a financial dataset, an unusually high transaction or an unexpected number of transactions in an account [1].

Contextual Anomaly: Occurs when a data point deviates from normal behavior. This is assessed based on the behavior of other data points in the dataset. For example, in the field of network security, deviation from the normal functions of a device or an unusual increase in visitor traffic to a website [1].

Collective Anomaly: Detection of groups that deviate from normal behavior models through analyses conducted on multiple features or attributes. For example, in a company's employee performance evaluations, identifying situations that deviate from the norm [1].

A. Network Anomaly Detection

Network Anomaly Detection is a collection of methods used to detect suspicious activities within a network, encompassing various disciplines such as network traffic analysis and system modeling [3]. The three main approaches are as follows:

Signature-Based Anomaly Detection: Compares network packets with known patterns and signatures. This method cannot detect new or unknown anomaly types [3, 4].

Behavior-Based Anomaly Detection: Analyzes system behaviors and network traffic to build normal models. It can detect both known and unknown anomaly types [3].

Machine Learning-Based Anomaly Detection: Utilizes a model trained to learn normal behaviors and classifies new data as normal or anomalous. Its flexible structure allows it to detect new and unknown anomaly types [3, 5].

The test data sets created for network anomaly detection simulate a large amount of normal and malicious network traffic. This is crucial to overcome the challenges of using real network traffic and to train/test machine learning algorithms. It includes different network structures, protocols, and types of anomalies/attacks in various targets. Some of the popular datasets are NSL-KDD [6], UNSW-NB15 [7], CICIDS2017 [8], DARPA 1998 [9], and KDDCUP99 [10]. The performances of machine learning algorithms used in network anomaly detection are usually evaluated on the KDD Cup 1999 dataset.

Network traffic analysis has critical importance due to the increasingly complex network structures and cyber threats. Various approaches have been presented to detect anomalies and solve problems, using feature reduction techniques such as PCA, and machine learning algorithms. In particular, methods like ANN, SVM, One-Class SVM, Autoencoder, Naive Bayes, and deep learning models have been reported to be effective in network traffic analysis. In this study, two different hybrid feature reduction methods, PCA + RFECV and RFECV + FS, were compared to assess the effectiveness of data mining and machine learning techniques in the field of network security. In the PCA + RFECV method, dimensionality was reduced using principal component analysis, and then the best features were selected using the Recursive Feature Elimination with Cross-Validation (RFECV) method. As evaluation metrics, Cross-Validation and ROC curves were preferred;

The organization of this article is as follows. In the second section, a comprehensive literature review is presented, including the existing studies in the field of network anomaly detection and the developments in this area. The third section provides a detailed examination of the various machine learning algorithms used for network anomaly detection and the criteria used to assess the performance of these algorithms. The fourth section presents a series of experiments applied to a network traffic dataset using the selected machine learning algorithms, as well as the results of these experiments. This section also evaluates the effectiveness of the proposed two different hybrid feature reduction methods. In the fifth and final section, a summary of the findings, the limitations of the study, and recommendations for future research are provided.

II. LITERATURE REVIEW

Network traffic analysis plays a vital role in protecting information and ensuring the continuity of network performance. The increasingly complex network structures and cyber threats bring along various challenges in network traffic analysis, necessitating the research of new and effective methods to overcome these challenges. In this context, numerous approaches are presented in studies conducted, focusing on detecting anomalies in network traffic and solving the problems caused by these anomalies.

While the use of eigen-decomposition techniques in network traffic analysis has been found beneficial by Hirose et al., Sheyner O. and colleagues have addressed how anomalies in network traffic data can be detected using statistical analysis techniques [11, 12]. In particular, focusing on the combination of

data mining and statistical analysis, Last M. et al. have examined One-Class SVM and Autoencoder algorithms in this context [13]. On the other hand, Chandola et al. have noted that artificial neural networks are superior to traditional methods in detecting anomalies in network traffic [14].

Mukkamala et al. have addressed the detection of network attacks through machine learning algorithms, particularly ANN and SVM [4]. Approaches regarding time and frequency domain analyses in this field have been presented by M. Thottan and C. Ji [1]. Particularly, while Liu and colleagues emphasized the effectiveness of k-means, BIRCH, and DBSCAN algorithms in anomaly detection, the effectiveness of the Naive Bayes algorithm in real-time network data detection was investigated by Zhao et al. [15, 16]. With the deep learning approach, the superiority of Convolutional Neural Networks (CNN) and LSTM-based models has been highlighted [17]. However, it has been stated that approaches based on Deep Belief Networks (DBN) within the scope of Software-Defined Networks (SDN) have produced superior results compared to CNN [18].

While Zhang et al. explored how statistical traffic traces could be used for the detection of P2P botnet activities, Tan et al. proposed multivariate correlation analysis for the detection of denial-of-service attacks in TCP and UDP protocols [19,20]. On the other hand, noteworthy methods in this field have been presented in studies conducted by Limthong et al., who proposed a wavelet-based neural network, and Bloedorn et al., who introduced a data mining approach over TCP/IP network traffic [21,22].

While Lakshman explored the efficiency of network attack detection using sampling and game theory [23], Boughaci et al. examined the capacity of the autonomous agents approach to detect network threats [24]. Additionally, Jain and Abouzakhar highlighted the advantages of combining Hidden Markov Model and Support Vector Machine [25], while Shyu et al. proposed an anomaly detection method that combines PCA and K-NN classifier [26]. While Garcia-Teodoro et al. presented a distributed network anomaly detection method [27], G. Poojitha et al. conducted studies on datasets for network attacks using artificial neural networks [28].

In the subjects of network anomaly and anomaly detection, there are numerous feature reduction methods and classification techniques in the literature. Particularly, PCA (Principal Component Analysis), as demonstrated in the research conducted by Chandola and his team [14], signifies a successful approach in this context. PCA reduces the correlations between features by compressing high-dimensional datasets.

However, as indicated by the literature, feature selection and reduction should not be limited to a single method. In this context, we aimed to assess the classification performance of features more specifically by combining PCA's linear transformation capabilities with RFECV (Recursive Feature Elimination with Cross-Validation). RFECV, as also mentioned by Mukkamala et al. [4], is a robust method capable of iteratively evaluating the classification performance of features.

The efficacy of feature reduction is directly proportional to the correct selection of features. Although RFECV on its own is quite a successful method of feature selection, we have combined it with the Feature Selection (FS) algorithm to refine this feature selection even further. The feature ranking created by RFECV has been further optimized with FS to increase the number of experiments and to confirm the accuracy of the results. We observed that this approach has not been used in the literature, especially on the KDDCUP 99 dataset, and we aimed to fill this gap. This hybrid approach demonstrates our innovative approach to feature reduction and its advantages when compared to other methodologies in the literature.

As documented in many studies in the literature, such as those by Mukkamala et al. [4], the success of different classification algorithms in network traffic analysis has been proven. These algorithms include methods such as Support Vector Machine (SVM), Naive Bayes, Decision Tree (DT), Logistic Regression (LR), Random Forest (RF), and K-Nearest Neighbors (KNN). Each of these algorithms offers the opportunity to comprehensively evaluate many approaches, as they are based on different mathematical and statistical foundations.

III. ANALYSIS OF MACHINE LEARNING ALGORITHMS AND PERFORMANCE METRICS

In this study, a research on network traffic analysis and performance evaluation has been conducted. Various machine learning algorithms such as KNN, NB, LR, DT, and RF have been utilized on the KDDCUP99 dataset to detect network anomalies [29]. The aim of the study is to

compare the sensitivities and performances of these algorithms to feature reduction methods. The research was carried out using the Windows 10 operating system, the Python 3.10.9 programming language, and libraries such as Numpy, Pandas, Scikit-learn, and Matplotlib. The study was conducted on an HP ProBook 450 G8 model computer, equipped with an Intel® Core™ i5 processor, 16 GB RAM, and an Intel® Iris® Xe Graphics unit. The focal point of the study is to evaluate the performances of different algorithms and to understand their responses to feature reduction methods. For this purpose, various performance metrics such as precision, sensitivity, F1 score, ROC curve, AUC, and Log Loss have been utilized. These metrics help evaluate the overall success of the model, its balanced performance, and the accuracy of prediction probabilities. Such a comprehensive evaluation method is particularly important in imbalanced datasets and critical application areas.

Accuracy: It is the ratio of correctly predicted samples to the total number of samples. The accuracy rate is calculated with equal weight for all classes.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (3.1)$$

TP (True Positive) The number of correct positive predictions,

TN (True Negative) The number of correct negative predictions,

FP (FalsePositive) The number of incorrect positive predictions, and

FN (FalseNegative) The number of incorrect negative predictions.

Precision: It is the ratio of true positive predictions to the total number of samples predicted as positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (3.2)$$

Recall (Sensitivity): It is the ratio of true positive predictions to the actual positive samples.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (3.3)$$

F1 Score: It is the harmonic mean of precision and recall metrics.

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.4)$$

ROC Curve (Receiver Operating Characteristic Curve): It is a curve calculated by plotting the False Positive Rate (FPR) against the True Positive Rate (TPR). The ROC curve demonstrates the performance of the classifier at different thresholds.

AUC (Area Under the Curve): It is the calculation of the area under the ROC curve. AUC quantitatively measures the performance of the classifier with a single numerical value.

Log Loss (LogLoss): It measures how well the probability predictions of the classifier correspond with the true class labels.

$$\text{Log Loss} = -\left(\frac{1}{n}\right) \times \sum(y_i \times \log p_i + (1 - y_i) \times \log(1 - p_i)) \quad (3.5)$$

Here, n is the number of samples, y_i is the true class label, p_i is the predicted probability, and log( ) refers to the natural logarithm.

In this study, machine learning algorithms such as Naive Bayes, Decision Tree, Logistic Regression, K-Nearest Neighbor, and Random Forest have been used.

Naive Bayes: It is a probability-based classification algorithm and is commonly used especially in natural language processing applications. This algorithm is based on Bayes' Theorem and makes the independence assumption among features. For example, it can perform email classification.

Decision Tree: It is an algorithm used for both classification and regression problems. It models the data set in a tree structure and places the attributes with the highest information gain at the nodes of the tree. The leaves of the tree represent the classes or values.

Logistic Regression: It is a classification algorithm used for dividing into two or more categories. It predicts the probability of the target variable using the weighted sum of the independent variables. It transforms the output into a value between 0 and 1 using the sigmoid function.

K-Nearest Neighbors (KNN): It is an algorithm used for both classification and regression problems, which classifies or evaluates a given sample by looking at its K nearest neighbors. The computational complexity is directly proportional to the size of the data set.

Random Forest: It is an ensemble learning algorithm created by combining decision trees. Each

tree is trained with random subsets of the dataset and features. For classification or regression, the predictions of the trees are combined through voting. It provides high accuracy and low variance.

These algorithms offer flexible and robust methods that can be used to solve various classification and regression problems.

IV. APPLICATION

A study has been conducted observing how tools and technologies used in machine learning and data science research can affect the success and applicability of the model. In the research, industry-standard tools and technologies were used throughout the process, from data processing to model training. The Anaconda distribution was preferred, coding was done in Python language on Jupyter Notebook, and libraries such as Sklearn, Numpy, and Pandas were utilized. The research was conducted on a PC on a 64-bit Windows platform. It is emphasized that machine learning techniques have been successfully used in many fields in recent years and are effective in detecting network attacks. In this study, an anomaly detection approach was adopted on the KDDCUP99 dataset, and an attack detection model was created, stating that this model could offer innovative solutions in the field of computer network security. The basic stages of this model are as follows:

a. Preprocessing the dataset
b. Determining the classification model by analysing the dataset
c. Identifying the appropriate features for classification
d. Evaluating results

All experiments conducted during the study were carried out in the “Jupyter Notebook” development environment using the “Pyhton” programming language.

A. Preprocessing

Data preprocessing is the process of preparing the dataset for analysis. During this stage, the accuracy of the data was checked, missing and inconsistent data were corrected, noisy data were cleaned, and categorical data were converted to numerical form. The "attack_type" column was used to differentiate between attack and normal traffic, median value assignment was made for missing values, and some columns were normalized with MinMaxScaler. Subsequently, unnecessary columns were removed, and the feature matrix (X) and label vector (y) were created. These operations aim to balance and clean the dataset, facilitating more effective training for the model.

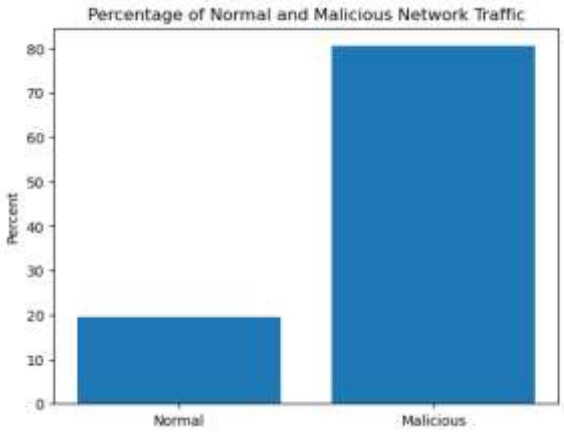


Figure 4.1: Percentage of Normal and Malicious Network Traffic.

Figure 4.1 reflects the ratio of normal to malicious network traffic in the dataset. As can be understood from Table 4.1, this dataset has an imbalanced distribution. This presents a significant challenge that can be encountered while training classification algorithms. The number of data representing an attack is much greater than the number of data representing normal traffic. This has resulted in 250,436 attack (malicious) data and 60,593 normal traffic data.

Imbalanced datasets, particularly in classification problems, can lead to models excessively learning the majority class and neglecting the minority class. This situation can result in an inability to accurately detect the rare classes (in this example, normal traffic), leading to security breaches. To solve this problem, the dataset has been balanced. Attack and normal traffic samples have been selected separately, obtaining an equal number of data points from both classes. The balanced dataset has been visualized with a bar graph, showing that both classes are equally represented in the dataset.

Figure 4.2 displays the percentage of normal and malicious balanced network traffic in the dataset.

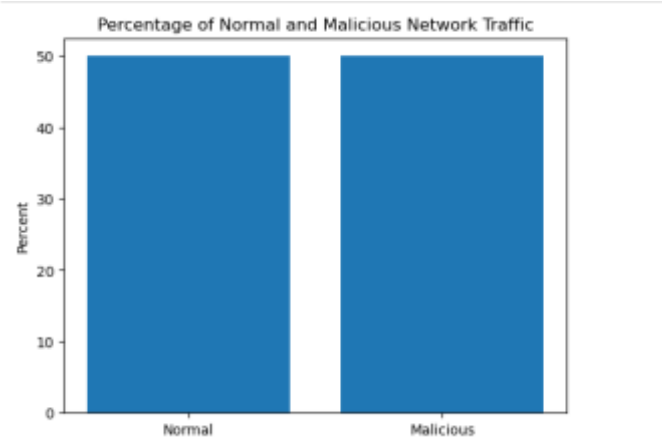


Figure 4.2: Balanced Percentage of Normal and Malicious Network Traffic.

Some of the attack types found in the dataset are presented in Table 4.1.

Table 4-1: Distribution of Attack Data in the Dataset.

No	Attack Name	Record Count
1	smurf	164091
2	normal	60598
3	neptune	58001
4	snmpgetattack	7741
5	mailbomb	5000
6	guess_passwd	4367
7	satan	1633
8	warezmaster	1602
9	back	1098
10	mscan	1053
11	apache2	794
12	processtable	759
13	saint	736
14	portsweep	354

B. *Feature Selection*

A new dataset containing the best features has been created using preprocessing and feature reduction methods. The classification model has been retrained on this newly created dataset, and the performance of the model has been evaluated.

Various metrics have been used to assess the performance of the model. These metrics include accuracy, precision, recall, F1 score, and ROC AUC score. These metrics are important for comprehensively evaluating the classification performance of the model.

The results have shown that the implementation of feature selection methods significantly enhanced the performance of the model. In particular, it has been observed that hybrid approaches such as PCA + RFECV and RFECV + FS have helped the model achieve better results in crucial metrics including accuracy, precision, recall, and F1 score.

As a result of this study, it has been concluded that feature selection and dimensionality reduction are of critical importance for developing an effective model to be used in intrusion detection systems. Additionally, it has been demonstrated that hybrid approaches are effective in enhancing the model's performance and optimizing processing time.

In conclusion, the feature selection methods and hybrid approaches developed in this study have significantly reduced the size of the dataset, enhancing the performance of the classification model and optimizing the processing time. Although the reduced number of features and the optimized processing time vary depending on the method used, both approaches have been effective in increasing the efficiency of classification models, especially in large and complex datasets, and in shortening the processing times. This plays a significant role in enabling attack detection systems to operate in real-time and effectively, thereby enhancing cybersecurity.

C. *Classification*

After data preprocessing and feature selection, 80% of the data was used to train the model, while 20% was set aside for performance testing. The binary classification method was chosen, yielding effective results on imbalanced datasets, and the results were simplified for sharing with non-technical stakeholders. A quick and efficient solution was provided for critical situations such as attack detection. The model's performance was evaluated using algorithms such as Decision Tree, Logistic Regression, Naive Bayes, Random Forest, and K-Nearest Neighbors. By reducing the data size with hybrid feature reduction methods, the model's performance was enhanced, and the processing time was shortened.

D. *Research Findings*

The values obtained for the classification without applying the feature reduction method are presented in Table 4.2 and Table 4.3.

Table 4-2: Test Results of Classification Performed without Using Feature Reduction Method.

Classifier	Confusion Matrix $\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$	Cross Validation
KNN	$\begin{bmatrix} 11653 & 310 \\ 584 & 11691 \end{bmatrix}$	0,9664
NB	$\begin{bmatrix} 11712 & 251 \\ 917 & 11358 \end{bmatrix}$	0,9539
LR	$\begin{bmatrix} 11675 & 288 \\ 1154 & 11121 \end{bmatrix}$	0,9440
DT	$\begin{bmatrix} 11921 & 42 \\ 426 & 11849 \end{bmatrix}$	0,9826
RF	$\begin{bmatrix} 11922 & 41 \\ 407 & 11868 \end{bmatrix}$	0,9833

Table 4-3: Validation Results of Classification Performed without Using Feature Reduction Method.

Classifier	Confusion Matrix $\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$	Cross Validation
KNN	$\begin{bmatrix} 11604 & 359 \\ 311 & 11964 \end{bmatrix}$	0,9784
NB	$\begin{bmatrix} 11878 & 85 \\ 9140 & 3135 \end{bmatrix}$	0,6263
LR	$\begin{bmatrix} 11747 & 216 \\ 706 & 11569 \end{bmatrix}$	0,9643
DT	$\begin{bmatrix} 11928 & 35 \\ 417 & 11858 \end{bmatrix}$	0,9834
RF	$\begin{bmatrix} 11928 & 35 \\ 403 & 11872 \end{bmatrix}$	0,9841

In the experiments, classification algorithms were initially tested without feature reduction, and basic performance metrics were determined. Subsequently, feature reduction was performed by combining PCA and RFECV methods, and the performance of the algorithm was re-evaluated. Through the comparison of these two stages, the impact of feature selection and reduction on classification performance was analyzed.

The results obtained for the classification based on the first feature reduction method are presented in Table 4.4 and Table 4.5.

Tablo 4-4 : Classification Test Results Based on the First Feature Reduction Method Results.

Classifier	Accuracy Score	Precision Score	Recall Score	F1 Score
KNN	0,9723	0,9708	0,9746	0,9806
NB	0,6193	0,7719	0,6193	0,5604
LR	0,9619	0,9816	0,9424	0,9616
DT	0,9813	0,9970	0,9660	0,9812
RF	0,9819	0,9970	0,9671	0,9818

Tablo 4-5 : Classification Validation Results Based on the First Feature Reduction Method Results.

Classifier	Accuracy Score	Precision Score	Recall Score	F1 Score
KNN	0,9631	0,9741	0,9524	0,9631
NB	0,9518	0,9532	0,9518	0,9517
LR	0,9405	0,9747	0,9059	0,9391
DT	0,9806	0,9964	0,9652	0,9806
RF	0,9815	0,9965	0,9668	0,9806

The ROC curves shown based on the initial classification results are presented in Figure 4-3, Figure 4-4, Figure 4-5, Figure 4-6, and Figure 4-7.

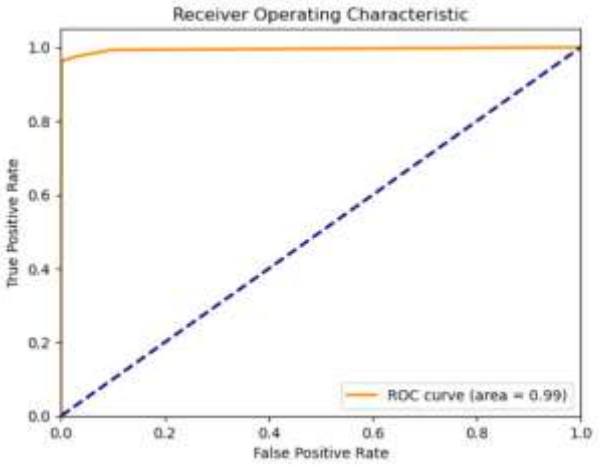
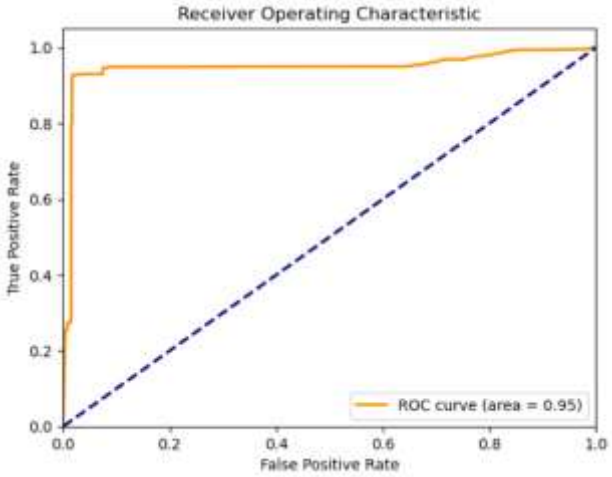
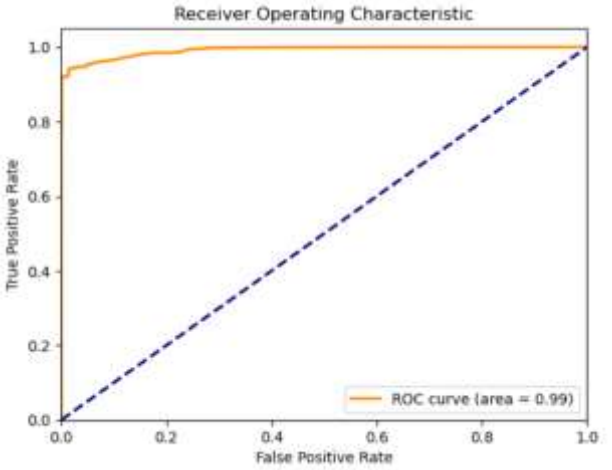


Figure 4-3: ROC Curve for the KNN Classifier in the First Experiment.



Şekil 4-4 : ROC Curve for the NB Classifier in the First Experiment.



Şekil 4 -5 : ROC Curve for the LR Classifier in the First Experiment.

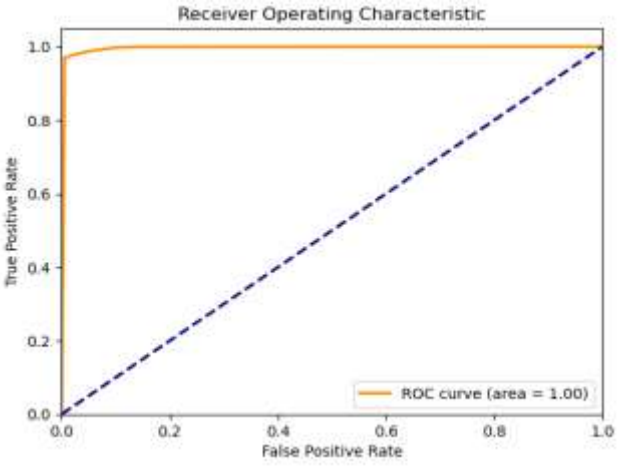
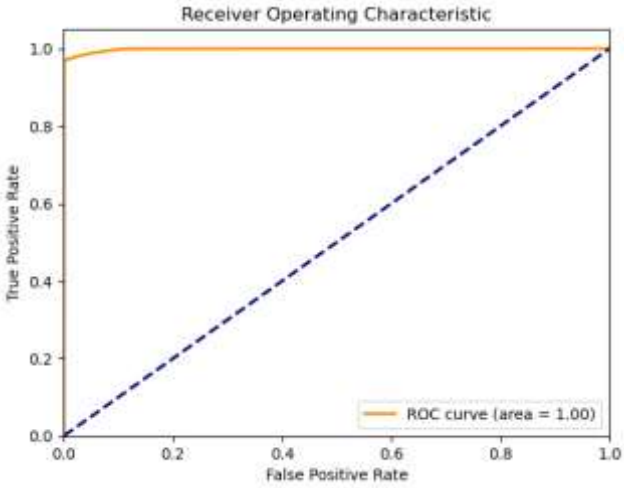


Figure 4-6: ROC Curve for the DT Classifier in the First Experiment.

Tablo 4-6 :Test Results for Classification Based on the Second Feature Reduction Method .

Classifier	Accuracy Score	Precision Score	Recall Score	F1 Score
KNN	0,9668	0,9776	0,9564	0,9669
NB	0,8914	0,9010	0,8914	0,8909
LR	0,9702	0,9930	0,9479	0,9699
DT	0,9812	0,9966	0,9662	0,9812
RF	0,9820	0,9971	0,9672	0,9819

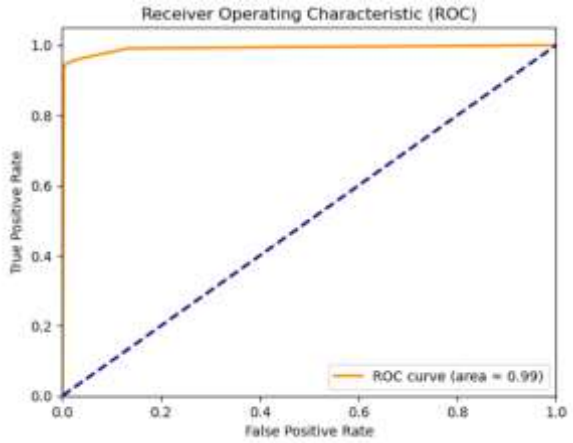


Şekil 4 -7 : ROC Curve for the RF Classifier in the First Experiment.

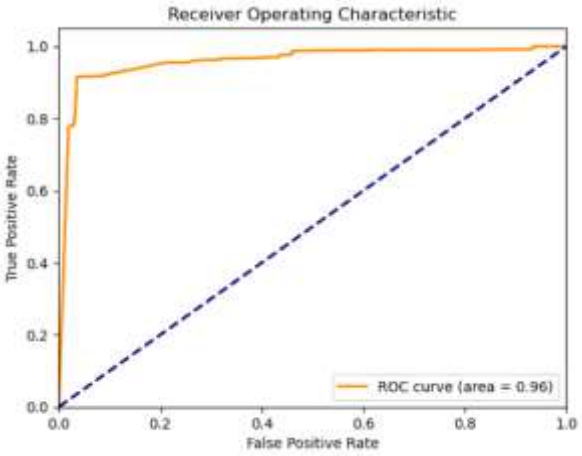
In the second experiment conducted with RFECV and FS methods, significant features were initially determined by applying RFECV with Decision Tree, and then a new dataset was created with these features, and feature selection was performed by applying FS method with RandomForestClassifier. The indices of the best features have been stored in the variable "selected_features". This process has been carried out to improve classification performance.

As a result, classification operations were performed using the best features determined by the RFECV + FS method, and the impact of this method in the feature selection stage was evaluated. The results related to classification operations are presented in Table 4.6 and Table 4.7.

The ROC curves obtained as a result of the analyses performed on the dataset reduced to 20 features using the RFECV and FS hybrid method can be seen in Figure 4.8, Figure 4.9, Figure 4.10, Figure 4.11, and Figure 4.12.



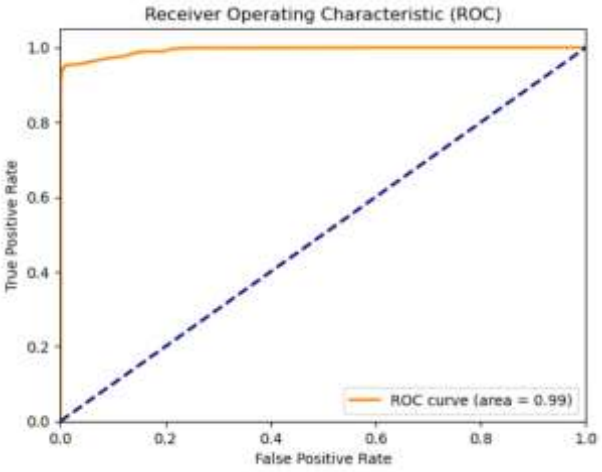
Şekil 4-8 : ROC Curve for KNN Classifier in the Second Experiment.



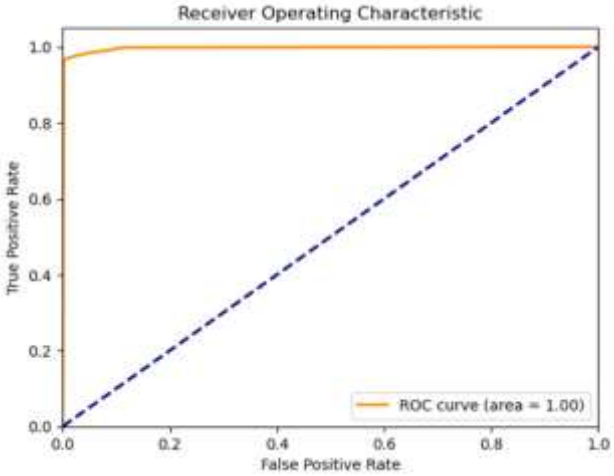
Şekil 4-9 : ROC Curve for NB Classifier in the Second Experiment.

Tablo 4-7 : Validation Results for Classification Based on the Second Feature Reduction Method.

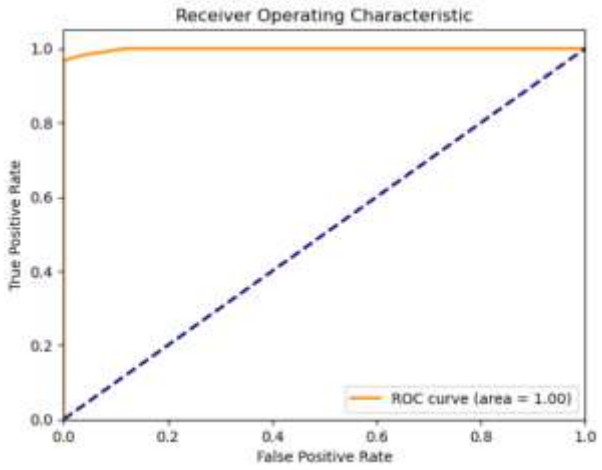
Classifier	Confusion Matrix $\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$	Cross Validation
KNN	$\begin{bmatrix} 11941 & 63 \\ 1599 & 48603 \end{bmatrix}$	0,947
NB	$\begin{bmatrix} 11573 & 390 \\ 2240 & 10035 \end{bmatrix}$	0,8949
LR	$\begin{bmatrix} 11881 & 82 \\ 639 & 11636 \end{bmatrix}$	0,9716
DT	$\begin{bmatrix} 11923 & 40 \\ 414 & 11861 \end{bmatrix}$	0,9834
RF	$\begin{bmatrix} 11929 & 34 \\ 402 & 11873 \end{bmatrix}$	0,9842



Şekil 4-10 : ROC Curve for LR Classifier in the Second Experiment..



Şekil 4-11 : ROC Curve for DT Classifier in the Second Experiment.



Şekil 4-12 : ROC Curve for RF Classifier in the Second Experiment

E. *Discussion of the Results of Experiments Conducted with Two Different Feature Groups*

The distinction of this study in the field of network anomaly detection lies particularly in the choice of dataset, the implementation of algorithms, and innovative approaches in analysis methods. For instance, the use of the KDDCUP99 dataset provides a rich data source that reflects real-world scenarios. This helps us understand how well the algorithm performances adapt to real-world situations. Compared to recent studies, this research stands out with its in-depth examination of the sensitivity of algorithms to feature selection and reduction methods. This aspect is especially evident in the application of PCA+RFECV and RFECV+FS methodologies. Such an approach can open new avenues in the field of network security and anomaly detection.

When we examine the results presented in Table 4.2, Table 4.3, Table 4.4, Table 4.5, Table 4.6, and Table 4.7, we can observe the impacts of the classifications conducted without feature reduction method and the feature reduction processes performed with PCA+RFECV and RFECV+FS methods on classification performance.

In the initial table obtained without applying feature reduction, it can be observed that the Naive Bayes classifier achieved 95.18% accuracy, 95.32% precision, 95.18% recall, and a 95.17 F1 score. However, when the PCA+RFECV method is applied, the Naive Bayes classifier yields 61.93% accuracy, 77.19% precision, 61.93% recall, and a 56.04 F1 score, which could indicate that the method has disrupted the compatibility of Naive Bayes with the dataset. When RFECV+FS is applied, the performance of Naive Bayes improves but does not reach the initial values. This situation demonstrates that RFECV+FS is a more suitable feature selection method for Naive Bayes compared to PCA+RFECV.

When PCA+RFECV is applied, the performance of Logistic Regression (LR) improves, and this improvement becomes even more pronounced with the RFECV+FS method. This indicates that RFECV+FS is a more effective feature selection method for LR.

The performance of the Decision Tree (DT) and Random Forest (RF) classifiers has improved with both feature reduction methods; however, the accuracy rate of RF has remained almost the same between these two methods, indicating that RF might be more resistant to feature selection. The performance of K-Nearest Neighbors (KNN) has also improved, especially the RFECV+FS method has proven to be effective in enhancing the precision and recall metrics of KNN, showing that KNN is sensitive to feature selection.

V. CONCLUSION

This study aims to detect and report abnormal behaviors and anomalies in network traffic. Anomaly detection systems enhance network security and protect crucial data by learning normal network traffic behaviors and identifying abnormal activities. In line with this purpose, previous studies have been thoroughly reviewed, and critical elements for network anomaly detection have been identified in the literature. Machine learning algorithms have been observed to achieve high success rates in recent times. The KDDCUP99 dataset, reflecting real-world scenarios and containing different types of network attacks, has been utilized in this study.

In the study, Decision Tree (DT), Logistic Regression (LR), Naive Bayes (NB), Random Forest (RF), and K-Nearest Neighbors (KNN) algorithms have been employed as classifiers, and ROC curves, along with cross-validation, have been chosen as the evaluation metrics.

Due to the large size of the original dataset, a smaller dataset named "corrected" has been utilized. This dataset has been derived from the original dataset by filtering out certain features. This process has eliminated insignificant and low-value network traffic data, while also removing the "duration" column, which negatively affected performance. Categorical features such as "protocol_type", "flag",

and "service" have been converted to numerical values, and the dataset has been normalized, making it ready for modeling.

The results presented in Tables 4.4, 4.5, 4.6, and 4.7 clearly demonstrate the effects of two different hybrid methods used in the feature selection process - PCA + RFECV and RFECV + FS - on classification performance. These results indicate that effectively reducing the size of the dataset and selecting meaningful features can enhance the success of classification operations.

Without feature reduction, the RF, DT, and KNN classifiers have exhibited high performance. However, when PCA+RFECV was applied, there was a significant drop in the NB classifier's performance, indicating that PCA might negatively affect how certain algorithms interpret the data. With RFECV+FS, although the performance of NB remained low, an increase in performance was observed for other classifiers such as LR, suggesting that RFECV+FS has the potential to optimize classification performance. Especially KNN showcased high performance with RFECV+FS, indicating that it can provide effective results with fewer features and is sensitive to feature reduction methods.

Particularly noteworthy are the changes in the performance of the Naive Bayes, Logistic Regression, and K-Nearest Neighbors algorithms. The negative impact of the PCA+RFECV method on the Naive Bayes algorithm suggests that this method might disrupt the algorithm's compatibility with the structure of the dataset. On the other hand, the improvement in the performance of algorithms such as Logistic Regression and K-Nearest Neighbors with the RFECV+FS method indicates that this method provides an optimized feature selection for certain classification algorithms. These findings enhance the applicability of these algorithms in practical network security scenarios, simultaneously enabling the development of more efficient and targeted anomaly detection systems. For future work, it is recommended to test these feature selection methodologies on larger and more diverse datasets and evaluate their compatibility with different classification algorithms.

Overall, the results of this study underscore the critical importance of feature selection and reduction in determining the success of classification models. The effectiveness of feature reduction methods used to enhance classification performance can vary depending on the classification algorithm. This work serves as a significant reference for academics and industrial institutions working on classification problems. In the future, it is recommended to conduct an in-depth analysis of how feature selection methods perform with larger datasets and different classifiers.

REFERENCES

[1] Thottan, M., Ji, C. (2003). Anomaly detection in IP networks. *IEEE Transactions on Signal Processing*, 51(8), 2191-2204

[2] *https://www.datascience.com/blog/python-anomaly-detection*, (Erişim tarihi: 24.04.2023).

[3] Denning, D. E. (1987). An intrusion detection model. *IEEE Transactions on Software Engineering*, 13(2), 222-232.

[4] Mukkamala, S., Janoski, G., & Sung, A. (2002). Intrusion detection using neural networks and support vector machines. *In Proceedings of the IEEE International Joint Conference on Neural Networks* (Vol. 2, pp. 1702-1707).

[5] Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*, 28(1-2), 18-28.,

[6] Stallings, W. (2013). Network Security Essentials: Applications and Standards. Pearson Education.

[7] Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A. A. (2009). A detailed analysis of KDDCUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada.

[8] Moustafa, N., Slay, J. (2015). The Evaluation of Network Anomaly Detection Systems: Statistical Analysis of the UNSW-NB15 Data Set and the Comparison with the KDD99 Data Set. Information Security Journal: A Global Perspective, 24(1-3), 18-31.

[9] Url-3 <https://www.unb.ca/cic/datasets/ids-2017.html>, erişim Tarihi: 17.09.2019.

[10] Haines, J. W., Rossev, L. M., Lippmann, R. P., Cunningham, R. K. (2001). Extending the DARPA off-line intrusion detection evaluations. *Proceedings DARPA Information Survivability Conference and Exposition II. DISCEX'01*, Anaheim, CA, USA, 35-45.

[11] Hirose, Shunsuke, Yamanishi, Kenji, Nakata, Takayuki, & Fujimaki, Ryohei. (2009). Network anomaly detection based on Eigen equation compression. Sayfa 1185-1194. doi: 10.1145/1557019.1557147.

[12] Sheyner O., Haines J., Javitz H., Stolfo S.,(2000) Intrusion Detection in Computer Networks Based on Statistical Analysis of Traffic Parameters. International Conference on Cyber Conflict. IEEE.

[13] Last, M., Kandel, A., Bunke, H. (2004). Data Mining in Time Series Databases. World Scientific.

[14] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: a survey. ACM Computing Surveys, 15(3).

[15] Liu, D., Lung, C.-H., Lambadańs, I., & Seddigh, N. (2013). Network traffic anomaly detection using clustering techniques and performance comparison. In

Canadian Conference on Electrical and Computer Engineering (pp. 1-4).

[16] Zhao, S., Chandrashekar, M., Lee, Y., & Medhi, D. (2015). Real-time network anomaly detection system using machine learning. In 2015 11th International Conference on the Design of Reliable Communication Networks, DRCN 2015 (pp. 267-270).

[17] Thing, V. L. L. (2017). IEEE 802.11 Network Anomaly Detection and Attack Classification: A Deep Learning Approach. In 2017 IEEE Wireless Communications and Networking Conference (WCNC) (pp. 1-6). San Francisco, CA, USA: IEEE. https://doi.org/10.1109/WCNC.2017.7925567.

[18] Kyaw, T., Oo, M. Z., & Khin, C. S., (2020), Machine-Learning Based DDOS Attack Classifier in Software Defined Network, 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 431-434, https://doi.org/10.1109/ECTI-CON49241.2020.9158230.

[19] Zhang, J., Perdisci, R., Lee, W., Sarfraz, U., & Luo, X., (2011), Detecting stealthy P2P botnets using statistical traffic fingerprints, DSN, 121-132, https://doi.org/10.1109/DSN.2011.5958212.

[20] Tan, Z., Jamdagni, A., He, X., Nanda, P., & Liu, R., (2011), Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis, International Journal of Recent Technology and Engineering (IJRTE), 756-765, https://doi.org/10.1145/2490428.2490450.

[21] Limthong, K. (2015). A wavelet-based anomaly detection for outbound network traffic. (Doctor of Philosophy thesis). Department of Informatics, School of Multidisciplinary Sciences, The Graduate University for Advanced Studies (SOKENDAI).

[22] Bloedorn, E., Christiansen, A., Hill, W., Skorupka, C., Talbot, L., & Tivel, J., (2002), Data Mining for Network Intrusion Detection: How to Get Started, InternationalConference on Data Mining, IEEE.

[23] Lakshman, M., (2003), Detecting Network Intrusions via Sampling: A Game Theoretic Approach, Proceedings of the IEEE INFOCOM 2003 - The Conference on Computer Communications, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies,1880.

[24] Boughaci, D., Drias, H., Bendib, A., Bouznit, Y., & Benhamou, B., (2006), Distributed Intrusion Detection Framework based on Autonomous and Mobile Agents, 2006 International Conference on Dependability of Computer Systems, 248-255, doi: 10.1109/DEPCOS-RELCOMEX.2006.19.

[25] Jain, R., & Abouzakhar, N., (2013), A Comparative Study of Hidden Markov Model and Support Vector Machine in Anomaly Intrusion Detection, Journal of Internet Technology and Secured Transaction, 2, 176-184, doi: 10.20533/jitst.2046.3723.2013.0023

[26] Shyu, M.-L., Chen, S.-C., Sarinnapakorn, K., & Chang, L., (2003), A Novel Anomaly Detection Scheme Based on Principal Component Classifier, in conjunction with the Third IEEE International Conference on Data Mining (ICDM'03), IEEE Foundations and New Directions of Data Mining Workshop.

[27] García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E., (2009), Anomaly-based network intrusion detection: Techniques, systems and challenges, Computers & Security, 28, 18-28, doi: 10.1016/j.cose.2008.08.003.

[28] G. Poojitha, K. N. Kumar and P. J. Reddy, Intrusion Detection using Artificial Neural Network, 2010 Second International conference on Computing, Communication and Networking Technologies, Karur, India, 2010, pp. 1-7, doi: 10.1109/ICCCNT.2010.5592568.

[29] Özalp, A. (2023). Siber Saldırıların Tespitinde Yapay Zekâ Tabanlı Algoritma Tasarımı. (Yayınlanmamış doktora tezi). Karabük Üniversitesi.