Detection of Network Anomalies with Machine Learning Methods

İhsan Rıza Kara
Department of Computer Engineering
College of Engineering and Natural Sciences
Maltepe University
Istanbul, Turkey
k.ihsan95@gmail.com
ORCID Number:0000-0002-8155-4608

Asaf Varol
Department of Computer Engineering
College of Engineering and Natural Sciences
Maltepe University
Istanbul, Turkey
asafvarol@maltepe.edu.tr
ORCID Number: 0000-0003-1606-4079

2022 10th International Symposium on Digital Forensics and Security (ISDFS) | 978-1-6654-9796-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISDFS55398.2022.9800814

***Abstract*— The present study, aimed to detect cyber-attacks, and unexpected access requests on devices in the telecommunication networks, enabling the necessary measures to be taken early. With K-Nearest Neighbors (KNN) and Naive Bayes machine learning methods, predicted whether the raw data packets contain cyber-attack according to different properties of these packets using the UNSW-NB15 dataset. KNN algorithms with different K values and the Naive Bayes method were compared according to accuracy rates and the results were given in the table. As a result, changes in accuracy rates were observed according to different k neighbor values in the KNN algorithm. Higher accuracy rates than Naive Bayes were achieved in the models created with the KNN algorithm.**

Keywords*— *Cyber Attack Detection, Supervised Learning, K-Nearest Neighbor Algorithm, Naive Bayes Theorem, UNSW-NB15 dataset

I. Introduction

Along with the benefits, information technologies also bring some important threats and risks. Cyber-attacks, one of the most important of these threats, are crucial for businesses and personal security. At the same time, it is seen that cybercrime causes very serious damage and costs worldwide [1].

Cyber-attacks have been increasing in both frequency and complexity over the years. This increasing frequency and complexity bring further advances and continuous innovation in defense strategies. Although, traditional intrusion detection and deep packet inspection methods are still widely used and recommended, they are no longer sufficient to meet the demands of increasing security threats [2]. Intrusion Detection Systems (IDS) are being developed to identify and classify cyber-attacks. Artificial intelligence-based methods are used more frequently to improve IDSs [3].

II. Literature Review

Many machine learning studies have been carried out in the literature to develop IDS systems. Since the performance of the proposed machine learning methods was evaluated with the UNSW-NB15 dataset, studies using this dataset are included in the present study.

Olamantanmi et al. developed an artificial neural network-based IDS suitable for real-time cyber-attack detection in their study in 2020. UNSW-NB15 dataset was used to evaluate the developed IDS. Neural Networks were used as the machine learning method. To minimize the difficulties encountered while detecting cyber-attacks in real-time network traffic from the study, feature selection was made using the "gain rate" method. The number of features in the original dataset, was reduced to 30 using these methods. Experimental results showed that the accuracy rate was 76.96%. Also, they mentioned that the dataset is a suitable dataset for the evaluation of IDSs [4, 5].

G. Kocher compared the classification performance of machine learning methods using the UNSW-NB15 dataset in their study in 2020. They found the highest accuracy rate with the "random forest" as 95.43% [6].

In the study performed by Yang et al., a new IDS was developed by combining the Improved Conditional Variational AutoEncoder (ICVAE) and Deconvolution Neural Network (DNN). NSL-KDD and UNSW-NB15 datasets were used to evaluate the performance of the ICVAE-DNN model. The proposed model creates new attacks based on intrusions to stabilize and increase the diversity of training data. Thus, unbalanced attacks are detected. Compared with KNN, Multinomial Naive Bayes, Random Forest, Support Vector Machine (SVM), DNN, and Deep Belief Network (DBN) algorithms higher performance values were obtained with the proposed model [7].

In the study by Aleesa et al., deep learning models based on ANN, RNN, and DNN have been proposed for intrusion detection. The UNSW NB15 dataset used in the study was handled separately as binary and multiple classifications. The highest performance output was obtained by the ANN method with an accuracy value of 99.26% in binary classification. In multi-classification (99.59%), it was obtained with the DNN method with an accuracy value [8].

In the present study aimed to predict these attacks with machine learning methods by using the UNSW-NB15 dataset containing 9 types of attacks: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. While detecting the specified types of cyber-attacks, the raw data package properties given in Table-1 were used.

III. Method

In the present study, K-Nearest Algorithm (KNN) and Bayes Theorem methods which are frequently used in Supervised Learning with MATLAB were used to predict cyber-attacks. One of the most important advantages of KNN and Naive Bayes algorithms is that their fitting and prediction times are fast. Also, the KNN model is easy to learn and effective if the training data is large. Distance metric for KNN was determined as Euclidean and accuracy rates were compared KNN Algorithm (for k=1, 2, 3, 4, 5 values) and Naive Bayes Theorem.

A. *Predict Models*

Supervised Learning: Machine learning, which can be considered a subfield of artificial intelligence, has been defined as a methodology for improving the performance of an application and making predictions using historical information, which is generally collected in electronic environment and made available to the learner in the form of electronic data, and has been made available for analysis [9]. Learning from experience can only be attributed to humans, whereas computers do not have the ability to learn. However, in supervised learning, algorithms can be used with the existing dataset to make predictions by comparing known results [9]. A higher prediction success can be achieved by using more data in supervised learning.

1) K-Nearest Neighbor Algorithm: K-Nearest Neighbor Algorithm is among the supervised learning methods that solves classification problems. A value of k (number of neighbors) is determined to predict the data class. The value indicates the number of nearest neighbors to the predicted data. In the method, the similarities of the data to be classified according to the normal behavior data in the learning set are determined, and the distance to the nearest neighbor is calculated with the distance metric. In the present study, MATLAB's default distance metric, Euclidean distance, was used for the K-Nearest Neighbor Algorithm. The result shows the nearest data class.

- Euclidean Distance: The Euclidean distance function is used to determine the closest distance between two points. For the K-Nearest neighbor algorithm, the distance between the test data and the training data closest to the test data is calculated and the data class is determined.

$$d(x,y) = \left(\sum_{i=1}^{n}(x_i - y_i)^2\right)^{1/2} \qquad (1)$$

d	: Distance
n	: Number of dimensions
i	: Initial value
x_i, y_i	: Data points

- Manhattan Distance: The Manhattan distance function is used to determine the closest distance between two points. The distance value calculated by the sum of the absolute values of the difference between the lines connecting the points in the coordinate plane [9].

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i| \qquad (2)$$

- Minkowski Distance: “Minkowski Distance is a generalized form of Euclidean and Manhattan Distance Metrics. It is used for distance calculations based on n variables” [9].

$$d(x,y) = \left(\sum_{i=1}^{k} | x_i - y_i| ^n\right)^{1/n} \qquad (3)$$

2) Naive Bayes Theorem: Naive Bayes Theorem aims to produce results by using universal facts and observations in creating a model of any situation. The most important feature that distinguishes this approach from classical statistical methods is that it uses observations and subjective opinions in the prediction of imprecise information [10].

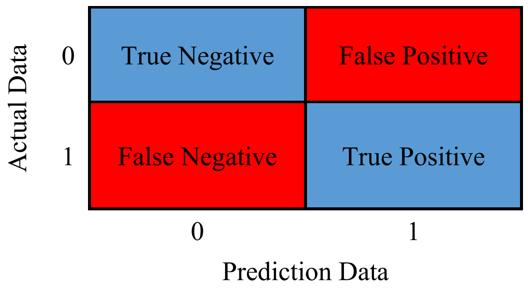
For any two events A and B,

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)} \qquad (4)$$

P(A): The probability of event A occurring
P(B): The probability of event B occurring
P(A|B): The probability of event A occurring when event B is known
P(B|A): The probability of event B occurring when event A is known

B. *Performance Metrics*

The actual and predicted values used in the calculations of performance criteria in classification problems are calculated by considering the confusion matrix. The meanings of the fields in the confisuon matrix are given in Figure-1.



Actual Data	0	1
0	True Negative	False Positive
1	False Negative	True Positive
	Prediction Data	

Fig. 1. Structure of Confusion Matrix

True Negative (TN): Indicates that data with an actual value of negative (0) was predicted correctly as negative.
True Positive (TP): Indicates that data with an actual value of positive (1) was predicted correctly as positive.
False Positive (FP): Indicates that data with a negative (0) actual value was incorrectly predicted as a negative.
False Negative (FN): Indicates that data with a positive (1) actual value was incorrectly predicted as a positive.

- Accuracy: Indicates the accuracy of the predicted values in the dataset.

$$\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{5}$$

- Precision: Precision indicates how many of the positive samples classified in the prediction area of the confusion matrix are correctly classified.

$$\frac{\text{TP}}{\text{TP+FP}} \tag{6}$$

- Recall: Recall indicates the ratio of correctly predicted positive values to the sum of true positive and false negative values. If the recall ratio is low, it means that the incorrect values are more and the correct values are less, while the high ratios indicate that the correct ones are too many and the incorrect values are less.

$$\frac{\text{TP}}{\text{TP+FN}} \tag{7}$$

- F1 Score: F1 score gives the harmonic average of precision and recall.

$$\text{F1 score} = 2 * \frac{\text{Presicion}\times\text{Recall}}{\text{Presicion+Recall}} \tag{8}$$

IV. DATASET

In the present study, raw network packets of the UNSW-NB15 dataset were used, where real modern normal activities and synthetic attack behaviors were generated by the IXIA PerfectStrom tool in the Cyber Range Lab of the Australian Cyber Security Center (ACCS) [11]. The UNSW-NB15 dataset consists of a total of 257,673 rows and 49 columns and includes the source and destination IP packets properties.

In the present study, by using 47 different features of the UNSW-NB15 dataset raw data packets given in Table-1, an answer was sought whether the packets contain cyber-attacks. The attack types given in the original dataset were not used in the present study. All attack types are considered cyber-attack.

V. PROBLEM

Cyber-attacks, one of the problems brought by the age of technology, are increasing day by day. Developing technology also brings some important threats and risks along with the benefits.

In the present study, using machine learning methods aimed to detect cyber-attacks, and unexpected access requests on devices in the telecommunication networks, enabling the necessary measures to be taken early.

TABLE I. PROPERTIES OF DATA PACKETS

Name	Description
srcip	source IP address
sport	source IP address
dstip	destination IP address
dsport	destination port number
proto	transaction protocol
state	indicates the state and dependent protocol
dur	record total duration
sbytes	source to destination transaction bytes
dbytes	destination to source transaction bytes
sttl	source to destination time to live (TTL) value
dttl	destination to source time to live (TTL) value
sloss	source packets retransmitted or dropped
dloss	destination packets retransmitted or dropped
service	used service (http,ftp, smtp, dns, etc.)
sload	source bits per second
dload	destination bits per second
spkts	source to destination packet count
dpkts	destination to source packet count
swin	source TCP base advertisement value
dwin	destination TCP window advertisement value
stcpb	source TCP base sequence number
dtcpb	destination TCP base sequence number
smeansz	packet size transmitted by the source
dmeansz	packet size transmitted by the destination
trans_depth	the pipelined depth into the connection of http request/response transaction
res_bdy_len	actual uncompressed content size of the data transferred from the server's http service.
sjit	source jitter (mSec)
djit	destination jitter (mSec)
stime	record start time
ltime	record last time
sintpkt	source interpacket arrival time (mSec)
dintpkt	destination interpacket arrival time (mSec)
tcprtt	TCP connection setup round-trip time, the sum of 'synack' and 'ackdat'.
synack	TCP connection setup time, the time between the SYN and the SYN_ACK packets.
ackdat	TCP connection setup time, the time between the SYN_ACK and the ACK packets.
is_sm_ ips_ports	if the source and destination IP addresses are equal and port numbers are equal then, this variable takes value 1 else 0
ct_state_ttl	number. for each state according to a specific range of values for source/destination TTL
ct_flw_http _mthd	number of flows that have methods such as Get and Post in HTTP service.
is_ftp_login	if the FTP session is accessed by the user and password then 1 else 0.
ct_ftp_cmd	number of flows that have a command in FTP session
ct_srv_src	number of connections that contain the same service and source address in 100 connections according to the last time
ct_srv_dst	number of connections that contain the same service and destination address in 100 connections according to the last time
ct_dst_ltm	number of connections of the same destination address in 100 connections according to the last time
ct_src_ ltm	number of connections of the same source address in 100 connections according to the last time
ct_src_dport _ltm	number of connections of the same source address and the destination port in 100 connections according to the last time
ct_dst_spor t _ltm	number of connections of the same destination address and the source port in 100 connections according to the last time
ct_dst_src _ltm	number of connections of the same source and the destination address in 100 connections according to the last time

A. Solution Methods

In the present study, the UNSW-NB15 dataset with a total of 257,673 records was used. 257,673 records were divided into two parts consisting of 175,341 and 82,332 records, and 175,341 records were used as training datasets and 82,332 records as test datasets. The training dataset containing 175,341 records was used to train the KNN and Naive Bayes models. Using the values of the data packet properties in Table 1, an answer was sought to the question of whether the packets contain a cyber-attack. For the K-Nearest Neighbor Algorithm, the k value is set to 1, 2, 3, 4, and 5 respectively. The Distance Metric for all k values is Euclidean. The models were tested with the dataset containing 82,332 records and predictions accuracy rates, precision, recall, and F1 score values are given in Table 2. Confusion matrices were used to calculate performance metrics. Confusion matrices for KNN (k=1, 2, 3, 4, 5) and Naive Bayes are given in Figure 2, Figure 3, Figure 4, Figure 5, Figure 6, and Figure 7, respectively.

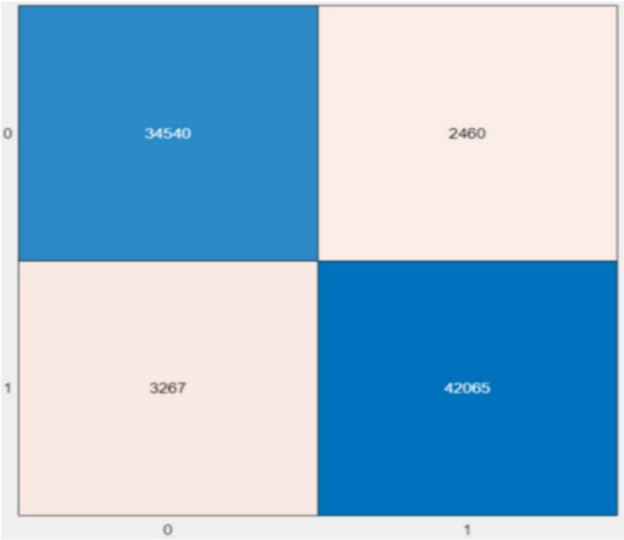


Fig. 2. Confusion Matrix for KNN Algorithm k=1

As shown in the Figure 2, with the model created for the k=1 value, 34,540 correct and 2,460 incorrect predictions were made from a total of 37,000 data that did not contain cyber-attacks. Of the 45,332 data containing the cyber-attack, 42,065 data were correctly predicted to be cyber-attack, and 3,267 data were incorrectly predicted as not cyber-attack, although they were cyber-attack. The accuracy rate of the created model is 93.04%.

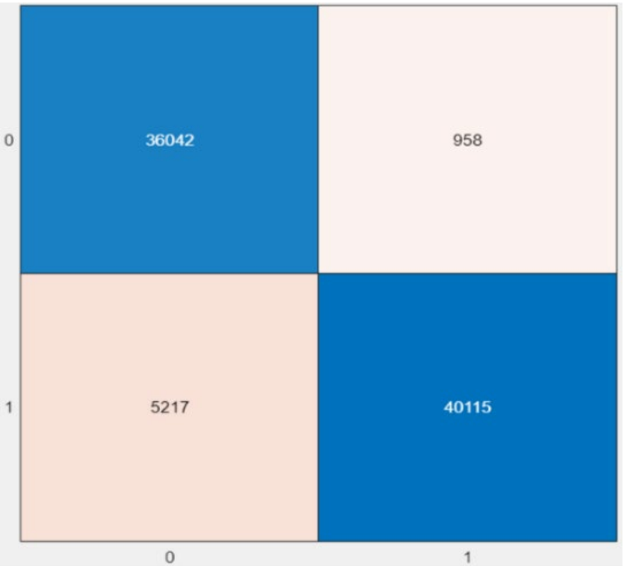


Fig. 3. Confusion Matrix for KNN Algorithm k=2

As shown in the Figure 3, with the model created for the k=2 value, 36,042 correct and 958 incorrect predictions were made from a total of 37,000 data that did not contain cyber-attacks. Of the 45,332 data containing the cyber-attack, 40,115 data were correctly predicted to be cyber-attack, and although 5,217 data were cyber-attack, it was incorrectly predicted that they were not cyber-attack. The accuracy rate of the created model is 92.50%.

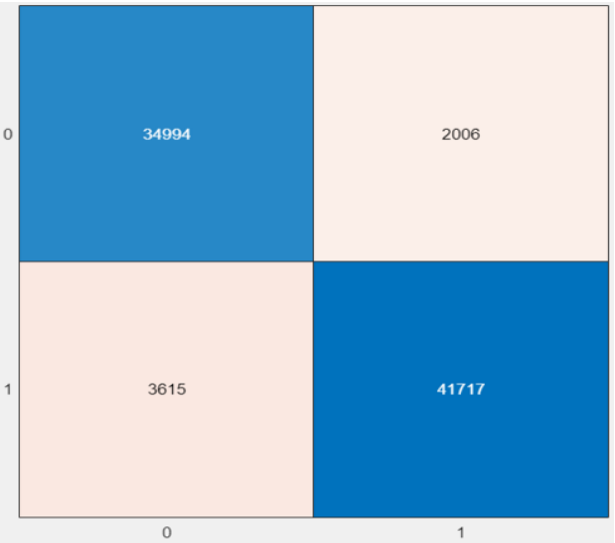


Fig. 4. Confusion Matrix for KNN Algorithm k=3

As shown in the confusion matrix Figure 4, with the model created for the k=3 value, 36,994 correct and 2,006 incorrect predictions were made from a total of 37,000 data that did not contain cyber-attacks. Of the 45,332 data containing the cyber-attack, 41,717 data were correctly predicted to be cyber-attack, and although 3,615 data were cyber-attack, it was incorrectly predicted that they were not cyber-attack. The accuracy rate of the created model is 93.17%.

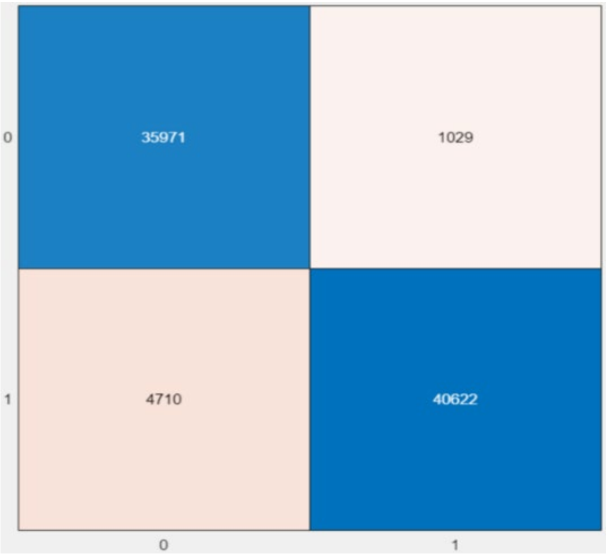


Fig. 5. Confusion Matrix for KNN Algorithm k=4

As shown in the Figure 5, 35,971 correct and 1,029 incorrect predictions were made from a total of 37,000 data that did not contain cyber-attack, with the model created for the k=4 value. Of the 45,332 data containing the cyber-attack, 40,622 data were correctly predicted to be cyber-attack, and although 4,710 data were cyber-attack, it was incorrectly predicted that they were not cyber-attack. The accuracy rate of the created model is 93.03%.

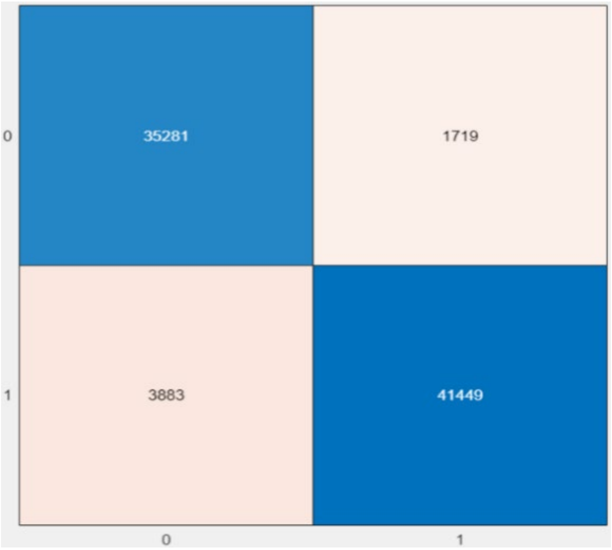


Fig. 6. Confusion Matrix for KNN Algorithm k=5

As shown in the Figure 6, 35,281 correct and 1,719 incorrect predictions were made from a total of 37,000 data that did not contain cyber-attack, with the model created for the k=5 value. Of the 45,332 data containing the cyber-attack, 41,449 data were correctly predicted to be cyber-attack, and although 3,883 data were cyber-attack, it was incorrectly predicted that they were not cyber-attack. The accuracy rate of the created model is 93.19%.

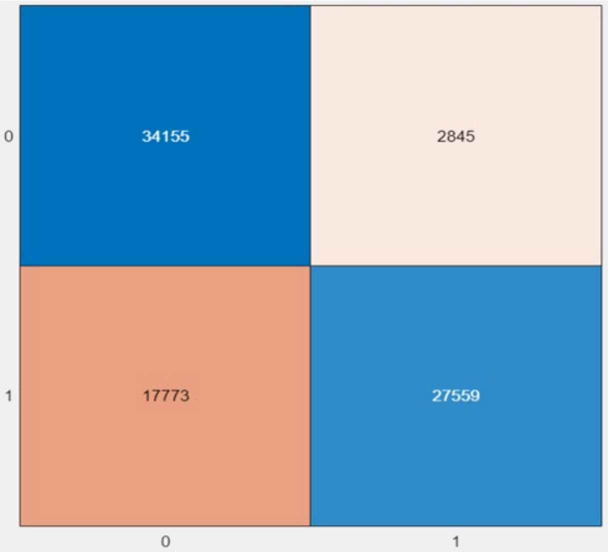


Fig. 7. Confusion Matrix for Naive Bayes

As shown in the Figure 7, with the Naive Bayes model, 34,155 correct and 2,845 incorrect predictions were made from a total of 37,000 data that did not contain cyber-attack. Out of 45,332 data containing the cyber-attack, 27,559 data were correctly predicted to be cyber-attack, and 17,773 data were incorrectly predicted as not cyber-attack, although they were cyber-attack.

It is seen in Table-2 that Naive Bayes is as successful as KNN on normal data that does not contain cyber-attack but makes 39.2% incorrect predictions on data containing cyber-attack. The accuracy rate of the created model is 74.95%.

TABLE II. PERFORMANCE METRICS OF THE METHODS USED

	KNN (k=1)	KNN (k=2)	KNN (k=3)	KNN (k=4)	KNN (k=5)	Naive Bayes
True Positive	34,540	36,042	34,994	35,971	35,281	34,155
True Negative	3,267	5,217	3,615	4,710	3,883	17,773
False Positive	2,460	958	2,006	1,029	1,719	2,845
False Negative	42,065	40,115	41,717	40,622	41,449	27,559
Accuracy Rate	93.04%	92.50%	93.17%	93.03%	93.19%	74.95%
Precision	0.9335	0.9741	0.9458	0.9722	0.9535	0.9231
Recall	0.4509	0.4733	0.4562	0.4696	0.4598	0.5534
F1 Score	0.6081	0.6370	0.6155	0.6333	0.6204	0.6920

In Table 2, correct prediction, and incorrect prediction values in normal data for different k values are listed. This table also shows how much of the cyber-attack data was predicted correctly and how much was incorrectly predicted. It is seen that the performance percentages are close to each other according to the changing k values in the KNN algorithm. It is noticed that the best performance is obtained for k=5. It is normal for performance percentages to change according to changing k values. Obtaining close values at varying k values for the dataset used indicates that the dataset is stable. In addition, the highest precision value was found in KNN for k=2 value, and the highest recall and F1 score value was found in Naive Bayes.

VI. Conclusion

The properties of the data given in Table 1 were used in the algorithms trained according to the dataset used, and predictions were made for the detection of cyber-attacks with the models created with these algorithms. The accuracy rate in the predictions made with the KNN Algorithm was found to be 93.04%, 92.50%, 93.17%, 93.03%, 93.19% for the k=1, 2, 3, 4, and 5 values, respectively. The accuracy rate in the test with the Naive Bayes Theorem was found to be 74.95%.

It has been determined that the Naive Bayes method is as successful as the KNN algorithm in data without cyber-attacks, but the accuracy rate in predicting data containing cyber-attacks is lower than KNN. The values found varied between 74.95% and 93.19%, and the highest accuracy rate was found for the KNN Algorithm k=5 value. It has been observed that the KNN Algorithm is more successful against the Naive Bayes Theorem.

The obtained accuracy rates are given in Table-2. There are 115,965 normal records and 59,376 cyber-attack records in the dataset used during the training of the models. It is seen that the normal records in the dataset used in the training are 56,589 more than the records containing cyber-attacks.

Since the imbalance between the number of cyber-attack data and the normal data number in the training dataset may adversely affect the Naive Bayes, it may cause the Naive Bayes model to interpret the cyber-attack data as normal data.

References

[1] M. S. Öztürk, “Siber Saldırılar, Siber Güvenlik Denetimleri ve Bütüncül Bir Denetim Modeli Önerisi,” Journal of Accounting and Taxation Studies, 10. Yıl Özel Sayısı, 2018, pp. 208-232. doi: 10.29067/muvu.340848.

[2] A. Delplace, S.M. Hermoso, & K. Anandita, “Cyber Attack Detection thanks to Machine Learning Algorithms”, 2020, doi: 10.48550/arXiv.2001.06309

[3] F. Demir, "Siber Saldırı Tespiti İçin Makine Öğrenmesi Yöntemlerinin Performanslarının İncelenmesi," Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Dergisi, c. 23, sayı. 2, ss. 782-791, Tem. 2021, doi:10.25092/baunfbed.876338.

[4] J. Mebawondu, Olamantanmi, Olufunso D. Alowolodu, Jacob O. Mebawondu, and Adebayo O. Adetunmbi, “Network Intrusion Detection System Using Supervised Learning Paradigm,” Scientific African 9: e00497, 2020, doi: 10.1016/J.SCIAF. 2020.E00497.

[5] Y. Türkyılmaz ve A. Şentürk, "Saldırı Tespitinde Makine Öğrenmesi Yöntemlerinin Performans Analizi," Avrupa Bilim ve Teknoloji Dergisi, sayı. 32, ss. 107-112, Ara. 2021, doi:10.31590/ejosat.1045551.

[6] G. Kocher and G. Kumar, “Performance Analysis of Machine Learning Classifiers for Intrusion Detection Using UNSW-NB15 Dateset”, 31–40, 2020, doi: 10.5121/csit.2020.102004.

[7] Y. Yang, K. Zheng, C. Wu, & Y Yang, “Improving the Classification Effectiveness of Intrusion Detection by Using Improved Conditional Variational AutoEncoder and Deep Neural Network,” Sensors, 2019, Basel, Switzerland https://doi.org/10.3390/s19112528

[8] A. Aleesa, M. Younis, A. A. Mohammed, and N. Sahar, “Deep-Intrusion Detection System with Enhanced UNSWNB15 Dataset Based on Deep Learning Techniques,” Journal of Engineering Science and Technology, 2021, 16(1), pp. 711-727.

[9] G. Gürsoy and A. Varol, "Prediction of Arrhythmia with Machine Learning Algorithms," 2021 9th International Symposium on Digital Forensics and Security (ISDFS), 2021, pp. 1-5, doi: 10.1109/ISDFS52919.2021.9486383.

[10] M. Akar, and S. Gündoğdu, “Bayes Teorisinin Su Ürünlerinde Kullanım Olanakları,” Journal of FisheriesSciences.com, 8(1), 2014, pp. 8-16.

[11] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Dataset for Network Intrusion Detection Systems (UNSW-NB15 Network Dataset)," 2015 Military Communications and Information Systems Conference (MilCIS), 2015, pp. 1-6, doi: 10.1109/MilCIS.2015.7348942.