İSTANBUL UNIVERSITY
CERRAHPAŞA

ORIGINAL ARTICLE
Electrica 2022; 22(2): 313-320

Multiple Classification of Cyber Attacks Using Machine Learning

Ebu Yusuf Güven[1], Sueda Gülgün[2], Ceyda Manav[2], Behice Bakır[1], Zeynep Gürkaş Aydın[1]

[1]Department of Computer Engineering, İstanbul University-Cerrahpaşa, İstanbul, Turkey
[2]Department of Computer Engineering, İstanbul Commerce University, İstanbul, Turkey

Cite this article as: E. Yusuf Güven, S. Gülgün, C. Manav, B. Bakır and G. Zeynep Gürkaş Aydın, "Multiple classification of cyber attacks using machine learning," *Electrica.*, 22(2), 313-320, 2022.

ABSTRACT

With the rapid growth of technology, the Internet's use and the number of devices connected to it are growing at a breakneck pace. As a result of this development, network traffic has increased in volume and has become more vulnerable. The focus has been on the development of learning intrusion detection systems in order to detect sophisticated and undetected threats. Because machine learning-based models achieve great accuracy in a short amount of time, they are commonly utilized in intrusion detection systems. Multiple classifications were made in this study to detect assaults on network traffic using machine learning. The model was created using the CICIDS2017 data set, which comprises both current and historical attacks. The high-performance computer was used to rapidly conduct tests on the CICIDS2017 data set, which contains around 2.8 million rows of data. We improved the performance of the machine learning models we developed by cleaning, normalizing, oversampling for an unbalanced number of labels, and reducing the size of the data set using feature selection methods. The random forest, decision tree, logistic regression, and Naive Bayes classifiers were all implemented on the pre-processed data set, and it was observed that the random forest classifier had the highest accuracy of 99.94%.

Index Terms—Artificial intelligence, intrusion detection systems, cyber security, CICIDS2017, pre-processing, HPC

I. INTRODUCTION

Intrusion detection systems (IDS) are systems used to detect unauthorized interventions on traffic. The IDS can be a hardware or software system that monitors, detects, and notifies the computer or network to an attack or intrusion [1]. Machine learning and artificial intelligence (AI) technologies are critical for automated cyber defense techniques, including monitoring, control, threat detection, and alarm systems [2, 3].

Intrusion detection data sets are representations of specific types of network-based attacks that have been identified. It is critical to choose the right datasets to reflect today's assault situations and to make real-time applications more useful [4]. We used the publicly available CICIDS2017 data set to conduct the experiments due to the aforementioned. There are 2.8 million traffic flow records in the CICIDS2017 data set. In 2018, Sharafaldin and others demonstrated that it was the most comprehensive and up-to-date data set available [5]. CICIDS2017 contains 11 criteria to be met in order to provide accurate data set that includes updated denial of service (DoS), distributed denial of service (DDoS), Brute Force, XSS, SQL Injection, Infiltration, Portscan, and Botnet attacks [6]. When the CICIDS2017 data set was evaluated, it was discovered that pre-processing was a required step. The missing and infinite values were cleared after the 5-day data set was combined into a single record. Standard normalization was chosen for the normalization of local outlier factor (LOF) and values in outlier regulation. Sampling methods were needed due to the unbalanced distribution of the records in the data set. Random oversampling (ROS), one of the oversampling methods, has been preferred for the attacks with low records to perform good learning. Linear discriminant analysis (LDA), a feature selection method, was used to reduce the difficulties of conducting repeated trials with many participants and to achieve high accuracy with minimal resources.

The CICIDS2017 data set was pre-processed to create new data set samples. We organized our data with several normalization methods, and we selected the most useful features for training using LDA, ROS, and LOF. Finally, the success of random forest (RF), decision tree (DT), logistic

Corresponding author:
Zeynep Gürkaş Aydin

E-mail: zeynepg@iuc.edu.tr

Received: March 3, 2022

Accepted: April 4, 2022

DOI: 10.54614/electrica.2022.22031

regression, and Naive Bayes methods was compared in this study, which used the CICIDS2017 data set to process huge number of rows of data on a high-performance computer (HPC). Performance metrics were obtained by optimizing hyperparameters using various normalization and sampling methods. It was also aimed to find the effect of pre-processing on our model's success.

The rest of this paper is organized as follows: section II discusses similar studies conducted by other researchers in related work. The methods and experimental settings used are explained in section III. In section IV, performance metrics for the methods and attack detection rates as a result of the applicable methods are presented. The last section concludes our work with a discussion.

II. RELATED WORK

Specialized IDSs are being developed to identify typical cyberattacks that can be visible in heterogeneous network traffic. There are two types of IDSs: rule-based and AI-based. These systems have been shown to produce proper results in less time than rule-based systems [7]. On the CICIDS2017 data set, Ahmetoglu and Daş tested six different feature selection techniques and compared their performance in deep learning-based classification models. To implement this system, the authors reduced the number of variables from 78 to 25 for multiple classifiers and used eight attributes for binary classifications. According to the test results, the success rate in all applications is greater than 92%. The authors of this study reached the conclusion that eight parameters are sufficient for IDSs to generate an attack alarm without knowing the type of attack [8].

Using the CICIDS2017 data set, the authors created a system based on the AdaBoost algorithm, which was found to be the most efficient among other systems for detecting DDoS attacks [9]. They attained precision, recall, and F1 score values of 0.77, 0.84, and 0.77, respectively, with this model. They applied synthetic minority oversampling technique (SMOTE) technique on the data set's minority class to boost these values and the model's sensitivity. They then chose 25 features from the data set using the ensemble approach and 16 features using the principal component analysis (PCA) method. They attained an accuracy of 0.81 by applying the AdaBoost classifier to the data set generated after these operations.

Kurniabudi et al. [10] classified the attributes in the CICIDS2017 data set into three subgroups based on their weights using the information gain feature selection. They used 20% of the data set for training and 30% for testing, with 70% used for training and 30% for testing. Each group was assigned one of five possible classifiers. When the results were analyzed, the model utilizing the RF classifier with 22 features had the highest accuracy value of 99.86%, and the model utilizing the J48 classifier with 52 features had the most remarkable accuracy value of 99.87%. When the processing time is compared, the model using the J48 classifier is significantly slower than the model using the RF.

Pelletier and Abualkibash [11] used the Boruta library in R language to perform data pre-processing and analysis on the CICIDS2017 data set. The data set was classified using artificial neural networks and the machine learning-based RF method. When the RF model's test results were reviewed, it was discovered that an average accuracy rate of 96.24% was achieved for the identification of various attack types in a 68.35-hour process. When the 500-repetitive artificial neural network and the RF model are compared, the RF algorithm is found to be more consistent in various attack types.

Aamir et al. [12] worked on Friday—working hours—afternoon data set, which is part of the CICIDS2017 data set and includes labels Benign, PortScan, and DDOS. In the first stage, 12 features containing monovalent and non-computable (infinite) values were removed from the data set. Secondly, the correlations of the features were calculated and the features whose results were below 20% were also eliminated. Standard scalar normalization was performed on the new data set with the remaining 21 attributes. About 70% of the obtained data set was divided for education and 30% for testing. Decision tree, discriminant analysis, support vector machine (SVM), nearest neighbor, and some community classifiers were used as classifiers, and 60.6%, 97.1%, 99.0%, 68.7%, and 85.5%, respectively, accuracy values were obtained.

Yi and Aye [13] constructed a model for detecting DoS and PortScan attacks on the CICIDS2017 data set by using six different classifiers. The RF classifier has a maximum accuracy of 99.799% for DoS attacks, while SVM and JriP classifiers have a 100% accuracy for PortScan attacks. The authors of [14] used the CICIDS2017 and NSL-KDD data sets to build a deep learning-based model. According to the results of the classifier utilized, the data set CICIDS2017 is 99.43% accurate and the dataset NSL-KDD is 99.63% accurate. The highest accuracy value has been achieved when the suggested deep neural network (DNN) model is compared with recent studies.

The CSE-CIC-IDS 2018 data set has also been used to run IDS models. Karaman et al. [15] used one of machine learning's methodologies, artificial neural networks, to analyze their data set. From this data set, they built five separate sub-data sets. They created models to identify whether a packet containing these data sets is a DDoS, BruteForce, Botnet, or DoS attack and what type of attack it is. Features are selected to match each data set based on trial and error. They identified 99.11%, 99.31%, 99.26%, 93.23%, and 92.26% accuracy for each sub-data set. Using the CIC-IDS 2018 data set, the authors of [16] created a convolutional neural network (CNN) model with two convolution and two max-pooling layers, presenting the data in visual form for intrusion detection. They created a recurrent neural network (RNN) model using the same data set and compared the outcomes of the two models to determine the model's performance. With the data acquired, it is found that the CNN model produces more accurate findings.

Gonzalez-Cuautle et al. [17] developed one of the models for IDS modeling that makes use of many data sets. Using the ISCX-Bot-2014 and CIDIDS-001 data sets, they selected 11 and 8 attributes from the data sets, respectively. They balanced the data distribution in both data sets using SMOTE. They selected hyperparameters for five distinct machine learning-based classification algorithms using grid search optimization. Among the models that incorporate these weights and biases, the model that combines SMOTE and grid search has the best success rate for both data sets. Sarnovsky and Paralic [18] evaluated the attack detection system model using the KDD99 data set, which they created by combining a hierarchical ensemble model with a knowledge model. The test results for detecting DOS, Probe, R2L, and U2R attacks are compared for models utilizing C4.5, RF, ForestPA, and ensemble method classifiers. It was found that the model utilizing deep learning produced superior values [19].

Using PCA and RF techniques, Alhowaide et al. selected features from the NSL, NB15, BotNetIoT, and BoTIoT data sets. Decision tree, k-nearest neighbor, Gaussian Naive Bayes, and RF classifiers were used to test this model. As can be seen from the results, the PCA technique reduces the data set at the best rate, and the best classifier is RF [20]. Intrusion detection systems models can also benefit from specially obtained data sets. Sin and others used seven different machine learning-based classifiers to identify DDoS attacks on a data set derived from SDN network traffic. In the AdaBoost classifier, they obtained the highest F value with 93% [21]. Elmasry et al. used common IDS data sets in their experimental studies and their results showed a significant improvement in network intrusion detection by their proposed particle swarm optimization (PSO)-based algorithm [22]. They also investigated the KDD CUP 99, NSL-KDD, CIDDS, and CICIDS2017 data sets using various deep learning models, including DNNs, long-term memory RNNs with gated recurrent units, and deep belief networks in [23]. The authors of [24] compared deep learning models and traditional machine learning methods on different data sets for masquerade detection, which is a subset of intrusion detection.

Eskandari et al. have used the NetMate tool to monitor Internet of things network traffic, capture packets, and generate their own data sets from these packets. They determined which attributes to use by analyzing network flow statistics. For single-class classification, the proposed model employs the LOF and isolation forest (iF) techniques. The model, which aims to detect port scanning, Hypertext Transfer Protocol (HTTP) brute force, Secure Shell (SSH) brute force, and SYN flood attacks, discovered that the F1 value with the iForest was the highest at 0.99 and the lowest at 0.79 [25]. Many IDS models have been established using the CICIDS2017 data set. Authors in [26] used CNN and KNN to determine whether there was an attack on the NSL-KDD data set. Authors in [27] used a DNN classifier to evaluate four distinct attribute selection methods. The greatest Kappa value is 0.9965 for a model utilizing the proposed feature selection technique.

III. METHODS

We compared models developed using various methods on new data set samples created during pre-processing studies on the CICIDS2017 data set in this study. First, we cleaned the data set prior to data cleaning operations by removing records that could be classified as outlier, missing, and noisy values. Next, we used min-max, standard scaler, max absolute, and robust scaler normalization methods to arrange the difference between values and different data types in the data set. This way, the effects of each attribute on the training process were balanced. Then, using the LDA, ROS, and LOF methods, we selected the features that would contribute most to training from the normalized features in the feature selection stage. Finally, we developed models using new data sets that had been pre-processed using machine learning methods such as RF, logistic regression, Naive Bayes, and DT. Figure 1 depicts the data processing stages and proposed model.

A. Experimental Settings and Hyperparameters

The HPC was used to rapidly conduct experiments on the CICIDS2017 data set, containing approximately 2.8 million rows of data. Experiments were carried out on this computer in four different environments: Anaconda Jupyter Notebook (Anaconda3 2019.10), Spyder (Spyder 3.3.6), Rapid Miner (Rapid Miner 9.8), and Python (Python 3.7.4).

TABLE I COMPARISON OF NORMALIZATION TECHNIQUES

Normalization Methods	Classifier Techniques	MAE	Time (s)
Non-normalized data set	Decision tree	0.0032	3345
	Random forest	0.0031	317.2
	Naive Bayes	2.9660	223.9
	Logistic regression	0.1912	3303.9
Robust	Decision tree	0.0033	311.9
	Random forest	0.0029	371.6
	Naive Bayes	3.2160	194.3
	Logistic regression	0.3832	1867.4
Min-max	Decision tree	0.0033	246.0
	Random forest	0.0034	307.3
	Naive Bayes	1.1979	193.9
	Logistic regression	0.1177	850.1
Maximum absolute	Decision tree	0.0033	249.8
	Random forest	0.0038	312.25
	Naive Bayes	1.1980	192.7
	Logistic regression	0.1177	828.5
Standard	Decision tree	0.0032	297.7
	Random forest	0.0032	327.9
	Naive Bayes	1.2070	194.1
	Logistic regression	0.0741	5334.9

MAE, mean absolute error.

We specified default settings for hyperparameters related with the classification techniques that we employed in our study. Considering the hyperparameters for RF within the scope of the study, we have selected the number of estimators as 100, minimum samples split as 2, minimum samples leaf as 1, minimum weight fraction leaf as 0, and minimum impurity decrease as 0. As hyperparameters for DT, a minimum sample split of 2, a minimum sample leaf of 1, a minimum weight fraction leaf of 0, and a minimum impurity decrease of 0 were chosen. For Naive Bayes, the hyperparameter smoothing was set to 1e-09. As hyperparameters for logistic regression, the penalty is set to L2, the tolerance is set to 0.0001, and the maximum iteration is set to 100.

B. Data Set

To test and compare machine learning methods used in IDSs, the data set must be globally recognized and comprise a diverse range of current threats [28]. Finding a valid and comprehensive data set for many researchers to perform and test their work is a major challenge

TABLE II COMPARISON OF OUTLIER DETECTION TECHNIQUES

Outlier Methods	Classifier Techniques	MAE	Time (s)
LOF	Decision tree	0.0035	168.9
	Random forest	0.0027	292.6
	Naive Bayes	1.2495	206.6
	Logistic regression	0.0701	6108.6
Isolation forest	Decision tree	0.0035	189.0
	Random forest	0.0032	296.4
	Naive Bayes	1.2142	189.7
	Logistic regression	0.0749	4210.6
Elliptic envelope	Decision tree	0.0033	195.6
	Random forest	0.0035	307.8
	Naive Bayes	1.2492	193.1
	Logistic regression	0.0723	4782.3

LOF, local outlier factor; MAE, mean absolute error.

[7]. There are many data sets for IDS systems, such as AWID-2015, Booters-2013, Botnet-2010-2014, CICDoS-2012-2017, CICIDS2017, CTU-13, DARPA-1998, DDoS2016, ISCX2012, ISOT-2010, KDD CUP 99-1998, Kyoto 2006+, LBNL-2004, NDSec-1-2016, NGIDS-DS-2016, NSL-KDD-1998, etc. [29].

CICIDS2017 is a state-of-the-art data set presented by the Canadian Cyber Security Institute, containing the latest attacks and features [29]. The CICIDS2017 data set contains approximately 2.8 million traffic flow records. It is a 5-day data set containing more than 11 different attacks [10]. When the CICIDS2017 data set was examined, several significant deficiencies were discovered. One of these issues is an imbalanced data set. In this data set, there were also missing and redundant data records [11]. CICIDS 2017 contains both benign and malicious attacks, including DoS, DDoS, brute force SSH, brute force FTP, heartbleed, infiltration, and botnet [29]. While NSL-KDD and CAIDA data sets contain limited and generically labeled attacks, the CICIDS2017 data set contains more realistic and diverse attacks [11].

TABLE III COMPARISON OF SAMPLING TECHNIQUES

Sampling Methods	Classifier Techniques	MAE	Time(s)
ROS	Decision tree	0.0013	283.6
	Random forest	0.0016	407.7
	Naive Bayes	0.9553	246.2
	Logistic regression	0.0741	8867.4
SMOTE	Decision tree	0.0021	310.9
	Random forest	0.0019	408.5
	Naive Bayes	0.9160	258.1
	Logistic regression	0.0790	11,496.1
Bootstrap	Decision tree	0.0011	233.5
	Random forest	0.0024	386.8
	Naive Bayes	0.9435	268.6
	Logistic regression	0.0760	10,120.5

MAE, mean absolute error; ROS, random oversampling; SMOTE, synthetic minority oversampling technique.

TABLE IV COMPARISON OF ACCURACY SCORES AND DURATIONS

Feature Selection Methods	Classifier Techniques	MAE	Time(s)
LDA	Decision tree	0.9992	9144.7
	Random forest	0.9994	14,266.2
	Naive Bayes	0.8731	8976.6
	Logistic regression	0.9070	9286.0

MAE, mean absolute error; LDA, linear discriminant analysis.

C. Data Set Pre-processing

In its raw form, the CICIDS-2017 data set contains 2 830 743 rows data, 79 features, and 15 labels. While the fact that the data set contains an excessive number of attack types is an advantage when building an IDS model, it is a disadvantage that it contains an excessive number of null values, outliers, and distant values, and the attack count is unbalanced. We discovered that when we did not perform detailed data pre-processing, our model had an accuracy of less than 30%. We have seen that to get higher accuracy values, the data set must first go through a good data pre-processing process. We aggregated the data into distinct days and subjected it to the following data pre-processing steps. The data set contains a variety of DoS attacks (Slowloris, SlowHttpTest, Hulk, and GoldenEye), DDoS attacks, three types of web attacks (Brute Force, XSS, and SQL Injection), FTP-Patator, SSH-Patator, PortScan, and Bot attacks. The data set contains 2 830 743 records, of which 471 454 are for attack traffic and 2 273 097 are for normal traffic. The study makes no reference of the features and description of the CICIDS2017 data set since they are discussed in length in [30, 31].

To begin, we verified the types of all the data in the data set and converted two attributes ("Flow_Bytes_s" and "Flow_Packets_s") that do not contain numerical data to numerical data. Then, because there were two similar features in the data set, we eliminated one of the "Fwd Header Length" features, leaving us with 78 features. We checked the data set for null and infinite values and removed them to make learning easier. To contribute to the data set's balance, we removed two attacks (infiltration and heartbleed) that had a little amount of data in comparison to other attacks. Fourteen different attacks with similar titles were assigned the same label in the data set. Begin, DoS, DDoS, Web, PortScan, Brute Force (for FTP and SSH Patators), and Botnet are the labels we have used. Normalization was necessary since the gap between the data in the data set is too great and the data contains both extremely large and extremely small values. We chose four distinct normalization approaches for our study since they are among the most frequently used. Standard scaler,

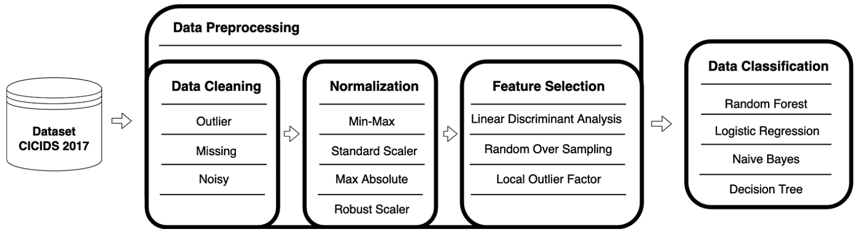

Fig 1. Stages of data processing and the development of the proposed model.

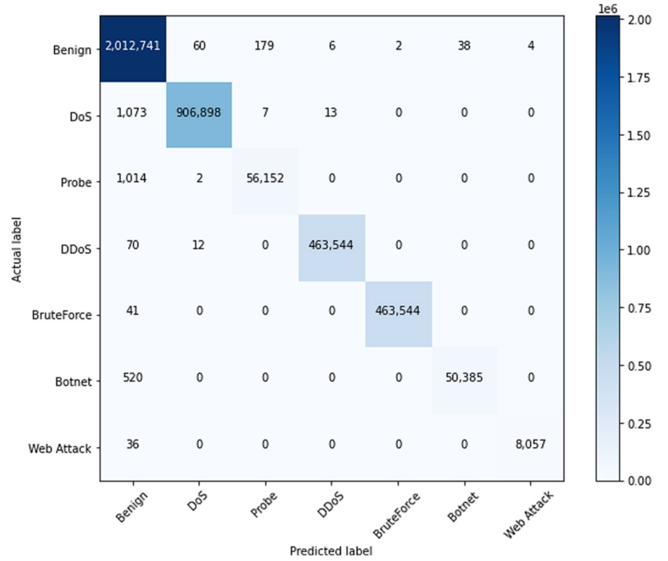

Fig 2. LDA and DT confusion matrix. LDA, linear discriminant analysis; DT, decision tree.

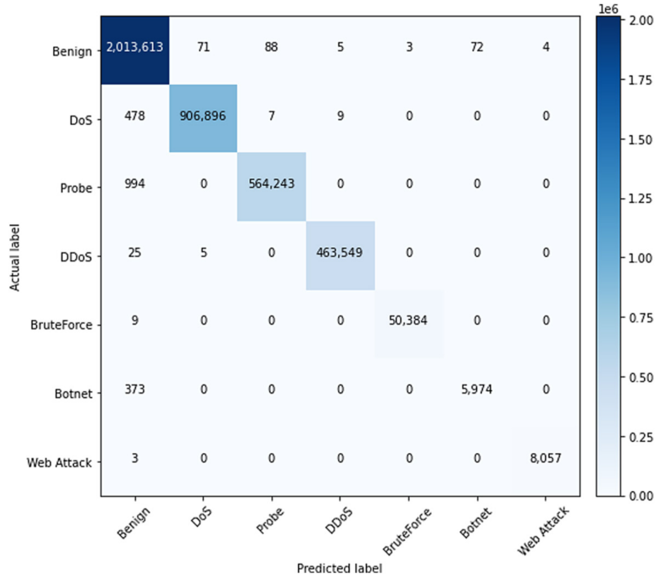

Fig 3. LDA and random forest confusion matrix. LDA, linear discriminant analysis.

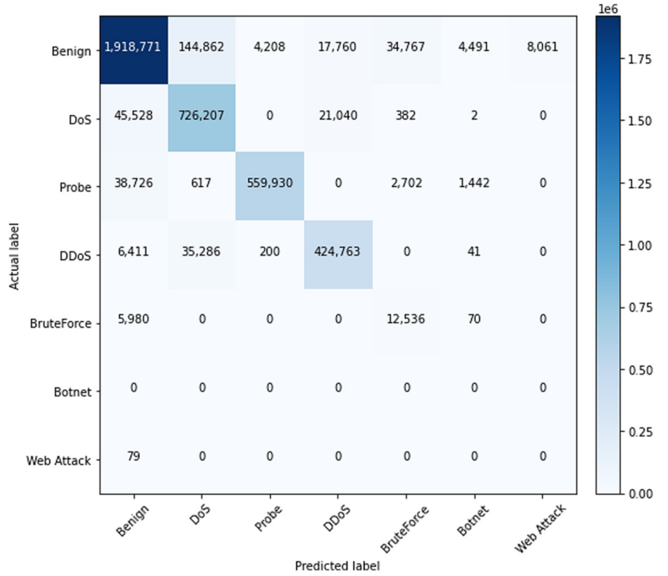

Fig 4. LDA and logistic regression confusion matrix. LDA, linear discriminant analysis.

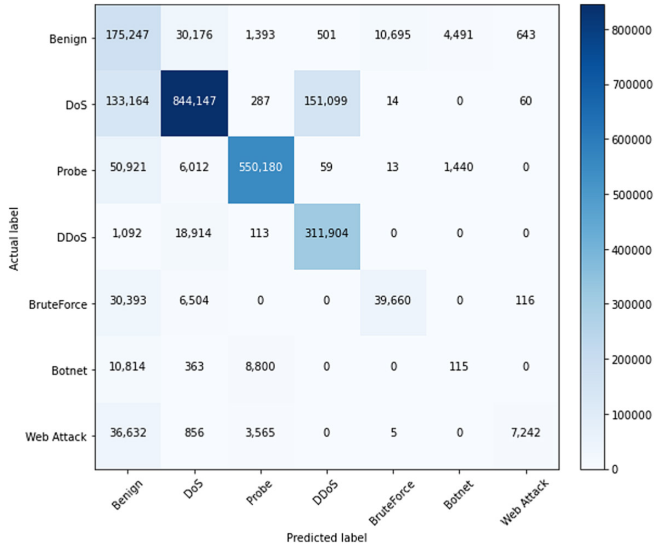

Fig 5. LDA and gaussian NB confusion matrix. LDA, linear discriminant analysis; NB, Naive Bayes.

min-max normalization, maximum absolute normalization, and robust scaler are the techniques we have chosen.

We examined for outliers within our normalized data. To improve the training process, we needed to either eliminate existing outliers or integrate them into normal values. We chose to omit these two alternatives. We assembled three of the most frequently used outlier techniques and applied them to our data set once more. We had 2 545 027 data in all techniques as a result of the operations. Our data set contained data with an asymmetric distribution. The data distributions varied significantly between attacks. As a result, we chose to use sampling to obtain a high level of learning in attacks with a small amount of data. We decided to use oversampling process with three different types of sampling methodologies as SMOTE, ROS, and bootstrap methodologies. We enhanced the data for each type of attack proportionally to ensure that the data set's dispersion rate was not compromised. The data on the news table is the outcome of sampling applications depending on attack type. Due to the large size of the data set used, it consumed a large amount of memory and required an excessive amount of time to calculate. That is why we required a reduction in size. We decided that the feature selection would be appropriate as a result of our logical correlation analyses. We used LDA to reduce 78 features to 6 features and the LDA algorithm to determine the model's success using six features.

IV. RESULTS

When the CICIDS2017 data set was examined, it was discovered that it contained incomplete, outlier, and infinite values, as well as records that were unbalanced based on labels. As a result, it was determined that the data set needed to be pre-processed. At each stage, we observed the effect of pre-processing operations such as data combination, data cleaning, mapping, normalization, outlier detection, sampling, and dimensional reduction on the model's success.

To perform multiple classifications on the CICIDS2017 data set, which contains data from 5 days of network traffic, the data from all 5 days was first combined into a single data set. Incomplete and infinite records were deleted to avoid causing an error during the learning of the model. Similar types of attacks are grouped under a single category within 11 attack tags. Thus, a total of seven labels were obtained, one of which was benign. We determined data with different scales for 78 features in the data set and applied normalization. The four most preferred normalization methods for the CICIDS 2017 data set were tested in the literature review, and their effects on the model were observed in terms of mean absolute error (MAE) and time in second as shown in Table I.

When the relevance of the data of the records in the same label category was considered, an extreme difference was found between the values and the outlier detection process was applied. Local outlier factor, isolation forest, and elliptic envelope were chosen as the most well-known techniques. Their effects on the model are shown in Table II.

After the mapping process, it was seen that some categories have relatively few records when the record numbers of the categories were examined. Oversampling was employed to balance the data set and improve learning. The ratio between categories is preserved when using oversampling methods. The effects of the oversampling methods on the model are shown in Table III.

One of the size reduction approaches, feature selection, has been chosen in order to achieve high success with fewer resources and to perform operations quickly. The effect of the LDA model's accuracy scores and durations are given in Table IV. Finally, the confusion matrix of the model subjected to classification is shown in Figs. 2-5. Standard normalization was used to normalize the data set, LOF was used to outlier detection, ROS was used for sampling, and LDA was used to select features. On the processed data set, RF provided the highest accuracy value of 99.94%.

V. CONCLUSION

The contribution of machine learning model to the success of IDS used for attack detection has been examined as this has recently become a serious problem. The purpose of this study is to develop a model on an unbalanced and pre-processed data set using machine learning techniques. The effect of each operation on the data set on the model was investigated step by step.

The data set was normalized using the normalization algorithms chosen during the feature selection stage, and MAE values and run times for four classifiers were recorded. If only one classifier was to be used in the model, the best normalization technique could be a different technique based on the values in the table, but the standard normalization procedure was preferred for our model.

Three techniques for detecting outliers were tested and their results were analyzed during the outlier detection stage. The analysis revealed that the LOF technique performed admirably with RF and logistic regression classifiers. Since other algorithms increase the error margin, LOF has been the preferred technique. Three different oversampling methods were chosen during the sampling stage. When classifiers were applied to the data set with a larger sample size, the MAE value decreased significantly. In contrast, the sampling method used in the logistic regression classifier increased the MAE value. Among the techniques for decreasing the MAE value, the ROS technique provided the best results. Linear discriminant analysis was preferred for the feature selection stage because it is a technique that automatically performs the size reduction process for models aiming to make multiple classifications and increasing the success of classification.

The choice of an algorithm that provides optimal results in terms of duration, source, and MAE was achieved by selecting a model with 99.94% accuracy. We obtained an accuracy of 99.94% with RF, 99.92% with DT, 90.70% with logistic regression, and 87.31% with Nave Bayes as a result of the classifications made on the completed data set. It was concluded that the data pre-processing stages contributed greatly to the success of the model in models aiming to detect multiple attacks with unbalanced data sets.

Peer-review: Externally peer-reviewed.

Author Contributions: Concept – E.Y.G.; Design – E.Y.G., Z.G.A.; Supervision –E.Y.G., Z.G.A.; Data Collection and/or Processing – S.G., B.B., C.M.; Analysis and/or Interpretation – S.G., B.B., C.M.; Literature Review – S.G., B.B., C.M.; Writing – E.Y.G., Z.G.A., S.G., B.B., C.M.; Critical Review – E.Y.G., Z.G.A.

Declaration of Interests: The authors declare that they have no competing interest.

Funding: The authors declared that this study has received no financial support.

REFERENCES

1. A. Thakkar, and R. Lohiya, “A review of the advancement in intrusion detection datasets,” *Procedia Comput. Sci.*, vol. 167, pp. 636–645, 2020. [CrossRef]
2. N. Ye, X. Li, Q. Chen, S. M. Emran, and M. Xu, “Probabilistic techniques for intrusion detection based on computer audit data,” *IEEE Trans. Syst. Man Cybern. A Syst. Hum.*, vol. 31, no. 4, pp. 266–274, 2001.
3. S. Rastegari, P. Hingston, and C. Lam, “Evolving statistical rulesets for network intrusion detection,” *Appl. Soft Comput.*, vol. 33, no. C, pp. 348–359, 2015. [CrossRef]
4. S. Rajagopal, P. P. Kundapur, and H. K. S., “Towards effective network intrusion detection: From concept to creation on Azure cloud,” *IEEE Access*, vol. 9, pp. 19723–19742, 2021. [CrossRef]
5. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” In *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Vol. 1. Funchal, Madeira, Portugal: ICISSP, 2018, pp. 108–116
6. A. Gharib, I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “An evaluation framework for intrusion detection dataset,” In International Conference on Information Science and Security (ICISS), Vol. 2016, 2016. Pattaya, Thailand: IEEE Publications, 2016, pp. 1–6.
7. T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, “Performance analysis of machine learning algorithms in intrusion detection system: A review,” *Procedia Comput. Sci.*, vol. 171, pp. 1251–1260, 2020. [CrossRef]
8. H. Ahmetoğlu, and R. Daş, “Analysis of feature selection approaches in large scale cyber intelligence data with deep learning,” In 28th Signal Processing and Communications Appl. Conference (SIU). Gaziantep, Turkey: IEEE Publications, 2020, pp. 1–4.
9. A. Yulianto, P. Sukarno, and N. A. Suwastika, “Improving adaboost-based intrusion detection system (IDS) performance on CIC IDS 2017 dataset,” In Journal of Physics: Conference Series (vol. 1192, no. 1, pp. 012018). IOP Publishing, 2019.
10. Kurniabudi, D. Stiawan, Darmawijoyo, M. Y. Bin Idris, A. M. Bamhdi, and R. Budiarto, “CICIDS-2017 dataset feature analysis with information gain for anomaly detection,” *IEEE Access*, vol. 8, p. 132911–132921, 2020.
11. Z. Pelletier, and M. Abualkibash, “Evaluating the CIC IDS-2017 dataset using machine learning methods and creating multiple predictive models in the statistical computing language R,” *Science*, vol. 5, no. 2, pp. 187–191, 2020.
12. M. Aamir, S. S. H. Rizvi, M. A. Hashmani, M. Zubair, and J. A. Ahmad, “Machine learning classification of port scanning and DDoS attacks: A comparative analysis,” *Mehran Univ. Res. J. Eng. Technol.*, vol. 40, no. 1, pp. 215–229, 2021. [CrossRef]
13. H. H. Yi, and Z. M. Aye, “Performance analysis of traffic classification with machine learning,” *Int. J. Comput. Inf. Eng.*, vol. 15, no. 1, pp. 42–47, 2014.
14. H. Azzaoui, A. Z. E. Boukhamla, D. Arroyo, and A. Bensayah, “Developing new deep-learning model to enhance network intrusion classification,” *Evolving Syst.*, vol. 13, no. 1, 17–25, 2022. [CrossRef]
15. M. S. Karaman, M. Turan, and M. A. AYDİN, “Yapay sinir ağı kullanılarak anomali tabanlı saldırı tespit modeli uygulaması,” *Avrupa Bilim Teknoloji Derg.*, pp. 17–25, 2020. [CrossRef]
16. J. Kim, Y. Shin, and E. Choi, “An intrusion detection model based on a convolutional neural network,” *J. Multimed. Inf. Syst.*, vol. 6, no. 4, pp. 165–172, 2019. [CrossRef]
17. D. Gonzalez-Cuautle *et al.*, “Synthetic minority oversampling technique for optimizing classification tasks in botnet and intrusion-detection-system datasets,” *Appl. Sci.*, vol. 10, no. 3, p. 794, 2020. [CrossRef]
18. M. Sarnovsky, and J. Paralic, “Hierarchical intrusion detection using machine learning and knowledge model,” *Symmetry*, vol. 12, no. 2, p. 203, 2020. [CrossRef]
19. S. K. Dey, and M. M. Rahman, “Effects of machine learning approach in flow-based anomaly detection on software-defined networking,” *Symmetry*, vol. 12, no. 1, p. 7, 2020. [CrossRef]
20. A. Alhowaide, I. Alsmadi, and J. Tang, “Pca, random-forest and Pearson correlation for dimensionality reduction in iot ids,” In *IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, Vol. 2020. IEEE Publications, 2020, pp. 1–6.
21. S. Sen, K. D. Gupta, and M. Ahsan, “Leveraging machine learning approach to setup software-defined network (SDN) controller rules during DDoS attack,” In, *Algorithms for Intelligent Systems*, *Proceedings of International Joint Conference on Computational Intelligence*. Singapore: Springer, (pp. 49–60), 2020. [CrossRef]
22. W. Elmasry, A. Akbulut, and A. H. Zaim, “Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic,” *Comput. Netw.*, vol. 168, 2020. [CrossRef]
23. W. Elmasry, A. Akbulut, and A. H. Zaim, “Empirical study on multiclass classification-based network intrusion detection,” *Comp. Intell.*, vol. 35, no. 4, pp. 919–954, 2019. [CrossRef]
24. W. Elmasry, A. Akbulut, and A. H. Zaim, “Deep learning approaches for predictive masquerade detection,” *Sec. Commun. Netw.*, vol. 2018, 1–24, 2018. [CrossRef]
25. M. Eskandari, Z. H. Janjua, M. Vecchio, and F. Antonelli, “Passban IDS: An intelligent anomaly-based intrusion detection system for IoT edge devices,” *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6882–6897, 2020. [CrossRef]
26. A. S. Kyatham, M. A. Nichal, and B. S. Deore, “A novel approach for network intrusion detection using probability parameter to ensemble machine learning models,” In Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India: IEEE Publications, 2020, pp. 608–613.
27. G. Farahani, “Feature selection based on cross-correlation for the intrusion detection system,” *Sec. Commun. Netw.*, vol. 2020, 1–17, 2020. [CrossRef]
28. A. Makuvaza, D. S. Jat, and A. M. Gamundani, “Deep neural network (DNN) solution for real-time detection of distributed denial of service (DDoS) attacks in software defined networks (SDNs),” *SN Comput. Sci.*, vol. 2, no. 2, pp. 1–10, 2021. [CrossRef]
29. A. A. Abdulrahman, and M. K. Ibrahem, “Toward constructing a balanced intrusion detection dataset based on CICIDS2017,” *Samarra J. Pure Appl. Sci.*, vol. 2, no. 3, 2020.
30. S. Singh Panwar, Y. P. Raiwani, and L. S. Panwar, “Evaluation of network intrusion detection with features selection and machine learning algorithms on CICIDS-2017 dataset,” In International Conference on Advances in Engineering Science Management & Technology (ICAESMT). Dehradun, India: Uttaranchal University, 2019.
31. J. Li, *Detection of ddos Attacks Based on Dense Neural Networks, Autoencoders and Pearson Correlation Coefficient* [MSc. Dissertation]. Canada: Dalhousie University, 2020.

Ebu Yusuf Güven has received his bachelor's degree from Istanbul University Computer Engineering in Turkey. He completed his master's degree with a thesis titled "Cyber Attack Detection and Prevention Methods for Edge Computing" at Fatih Sultan Mehmet University Computer Engineering, He is currently a Ph.D. student at Istanbul University-Cerrahpasa Computer Engineering, and his thesis is entitled "Development of a New Scan Model for Cyber Threat Intelligence". He works as a research assistant at Istanbul University-Cerrahpasa and researcher at the IoT Security Test and Evaluation Center (ISTEC). He has strong interests in cyber security and related fields.

Süeda Gülgün has received her bachelor's degree from Istanbul Commerce University Computer Engineering in Turkey. She worked as an AI researcher in Internet of Things Security Test and Evaluation Center-Istanbul University after graduating. She is currently employed as a Research and Development Engineer at Crypttech Cyber Security Intelligence in Istanbul. Her job entails employing AI and a variety of modern technologies to develop cybersecurity solutions. A large portion of her profession is about developing architectural blueprints by solving tough difficulties.

Ceyda Manav has received her bachelor's degree from Istanbul Commerce University Computer Engineering in Turkey. She worked as an AI researcher in Internet of Things Security Test and Evaluation Center-Istanbul University after graduating. She is currently working as SAP Software Support Specialist at Hitsoft in Istanbul. Her job requires software support in the fields of enterprise resource planning and data management.

Behice Bakir is currently student of computer engineering Istanbul University-Cerrahpaşa. Her research interests include data science, machine learning, computer vision and autonomous technologies.

Gülsüm Zeynep Gürkaş Aydin is an Assistant Professor in the Department of Computer Engineering Cyber Security Department at Istanbul University in Istanbul, Turkey. She received her BSc and MSc in Computer Engineering from Istanbul University. In 2011 and 2014, she received PhD degrees in computer engineering from Istanbul University and computer science from Université Pierre-Et-Marie-Curie: Paris VI / Telecom SudParis. Her current areas of expertise include data and computer communications, wireless and mobile networks, the internet of things, and cyber security. She is also a member of Istanbul University-Cerrahpaşa's Internet of Things Security Test and Evaluation Center (ISTEC). Additionally, she has authored publications on mobility management, indoor localization, content delivery networks, and wireless communications.