Cybersecurity Attack Detection Model, Using Machine Learning Techniques

İsa Avcı[1] and Murat Koca[2]

[1]Department of Computer Engineering, Faculty of Engineering, Karabuk University, Kılavuzlar Mahallesi 413. Sokak No: 7, 78000, Merkez, Karabuk, Turkey, isaavci@karabuk.edu.tr

[2]Department of Computer Engineering, Faculty of Engineering, Van Yuzuncu Yil University, Kampüs, 65080, Tuşba, Van, Turkey, muratkoca@yyu.edu.tr

Abstract: Millions of people use the web every day, in this age of technology and the internet. Protecting the privacy and security of these users is a significant challenge for cybersecurity developers. With tremendous technological advancements, there is a noticeable improvement in the cyber-attackers' capabilities. At the same time, traditional Intrusion Detection Systems (IDS) are no longer effective at detecting intrusions. After the tremendous competences achieved by Artificial Intelligence (AI) techniques in all fields, great interest has developed in its use in the field of cybersecurity. There have been many studies that use Machine Learning (ML)-based intrusion detection systems. Despite the strong performance of ML techniques in detecting malicious activities, some challenges still reduce accuracy of performance. Knowing the proper technique, as well as knowing the features, is essential for effective intrusion detection. Therefore, this study proposes an effective network intrusion detection system based on ML and feature selection techniques. The performance of four ML techniques, the Random Forest (RF), K-Nearest Neighbors (KNN), Support Vector Machine (SVM) and the Decision Tree (DT) systems for intrusion detection are explored. In addition, feature selection techniques are employed for the selection of important features. Among the techniques used, the RF technique achieved the best performance, outperforming other techniques, with an accuracy of 99.72%. This study elaborates on the detection of malicious and benign cyber-attacks, with a new-level, high accuracy.

Keywords: cybersecurity; intrusion detection; DDoS attacks; machine learning; feature selection techniques

1 Introduction

All technological developments, including smartphones, computers, communication systems and IoT devices lead to the development of internet networks, throughout the world [1]. Recent studies estimate more than 5 billion smart devices and 3 billion Internet users worldwide [2]. As a result of this great use of Internet networks, massive amounts of data are generated every second, which led to the creation of significant security challenges to protect data from the many challenges facing cybersecurity developers [3]. Cybersecurity protects computer systems and networks from unauthorized access [4]. Cybersecurity is a backbone for all types of companies, governments, and even people to secure data and maintain privacy. People send and receive data over the internet, which can be hacked and manipulated by strangers [5]. Cyber-attacks in 2017 caused damages of $5 billion and will only increase in the future; for example, damages were estimated to reach $6 trillion, annually, by 2021 [6].

Distributed Denial of Service (DDoS) attacks are among the most common cybersecurity attacks. DDoS is a lethal weapon that overwhelms a server or network by sending large amounts of packets, which floods the servers and causes the service to stop [7]. In recent years, DDoS attacks have witnessed an alarming increase. In February 2020, Amazon Web Services (AWS) customers suffered a severe crash when a DDoS attack targeted Amazon Simple Storage (S3) and other services, shutting them down for approximately 8 hours [8]. It is one of the most significant DDoS attacks, with a capacity of 2.3 terabytes per second. According to a Security Week article, researchers have discovered that the average number of separate DDoS attacks infecting the internet daily is about 28,700 [9]. Therefore, there was a need to create a reliable system for detecting cybersecurity attacks. Cybersecurity developers aim to create an effective IDS that can identify known and new attacks and threats with high accuracy and a low false alarm rate [10]. Although many methods are available for intrusion detection, the increase in the effectiveness of recent attacks and the evolution of attack methods, especially DDoS attacks, requires effective intelligent methods to detect them. Whereas, traditional IDSs are no longer effective in intrusion detection [11].

The use of artificial intelligence techniques in the field of cyber security has become mandatory with great success in every field. AI and ML techniques provide a tremendous ability to explore hidden models in big data, allowing them to help decision-making. ML techniques help detect and monitor attacks on network traffic activity. Many studies used different ML techniques to detect intrusion. However, there are still some shortcomings, including determining the appropriate ML technique for the intrusion detection process [12]. Basic and effective features have been selected to improve the performance of ML techniques in the detection of unauthorized inputs [13].

This study aimed to create an effective IDS based on machine learning and feature selection techniques. In our new approach, critical features that affect the classification result are selected as a basis for a more accurate classification process. The performance of four different machine learning techniques Random Forest (RF), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) are compared to find the appropriate technique for intrusion detection. One of the most effective ways to select features is to use a DT technique to determine the feature's importance. The feature importance feature of the DT technique was found useful to determine the importance of each feature and its effect on the classification result. This proposed study is designed to detect malicious or benign traffic in DDoS cyber-attacks with a new high accuracy. In addition, the outliers in the data's normalization steps were standardized using regression methods. Data robustness and confidence intervals were also examined in detail.

The research article is organized as follows and related studies are discussed in Section 2. It describes the general methodology preprocessing steps and the application of machine learning techniques in Section 3. Section 4 deals with the results obtained and their discussion. Section 5 provides conclusions. Performance experiments are performed using the NSL-KDD dataset to determine whether the activity is innocuous [14]. This study evaluated the model's performance using the confusion matrix. The performances of the ML models were calculated in terms of accuracy, precision, sensitivity, specificity, and F1-Score.

2 Related Work

The significant development in technology and Internet of Things (IoT) devices leads to increasing use of Internet networks, which in turn requires effective protection methods to protect the privacy and security of the user. Artificial intelligence is one of the most promising approaches to countering cybersecurity threats. Many studies have used IDSs based on ML and Deep Learning (DL) techniques. This section discusses a set of studies that use ML and DL techniques to detect cybersecurity attacks.

Bindra and Sood explored 6 ML techniques LR, KNN, RF, NB, Linear SVM, and Linear Discriminant Analysis (LDA) to find out the best technique for detecting DDoS attacks [15]. ML techniques were tested on the CIC IDS dataset, where RF technology achieved the best performance with an accuracy of 96.5%, superior to the rest of the techniques. Also, Chavan et al. studied the performance of four ML techniques KNN, SVM, DT, and LR for detecting DDoS attacks [16]. Of all the techniques used, LR achieved the best accuracy with 90.4%, outperforming the rest. The ensemble method often produces a better accuracy rate than a base classifier. Therefore Das, Saikat, et al. proposed an ensemble model that combines 4 base machine learning ML techniques Multilayer Perceptron (MLP), SVM, KNN, and DT [17]. The performance experiments were tested on the NSL-KDD data set, where the ensemble classifier achieved better results than the individual classifiers used in the same study.

Kasim suggested using an Auto-Encoding (AE) method for selecting features and reducing dimensions to effectively classify traffic [18]. The AE is used to identify essential features, and the SVM classifier then detects a DDoS attack.

Performance experiments were performed on CICIDS2017 and NSL-KDD datasets, and the results showed the model's effectiveness for classifying traffic. Bhardwaj et al. introduced a method that combines well-stacked sparse AE for feature learning using a Deep Neural Network (DNN) to detect potential DDoS attacks [19]. The performance of AE and DNN was tuned by adjusting parameters to improve detection accuracy. Performance experiments were conducted on the CICIDS2017 and NSL-KDD datasets. The results showed that AE + DNN was superior to AE + SVM in the study with the NSL-KDD dataset, while the results were competitive, when using CICIDS 2017.

The high-efficiency DL techniques achieved in discovering big data have been many efforts to explore it in the field of cybersecurity. Al-Emadi et al. explored the performance of DL techniques in Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) for network intrusion detection [20]. The performance experiments showed that the CNN technique is superior to the RNN technique when tested on the NSL-KDD dataset. Also, Abu Abu Al-Haija and Zein-Sabatto proposed CNN techniques for intrusion detection [21]. Performance experiments were conducted on the NSL_KDD dataset, whereby the CNN technique achieved a classification accuracy of 99.3% for intrusion detection. The performance of ML and DL techniques is highly dependent on data quality. Therefore, Xavier Larriva-Novo et al. explored various preprocessing techniques to classify traffic using the DNN technique [22]. This study showed, that by using preprocessing techniques, accuracy could be improved by up to 45%.

3 Proposed Methodology

This study proposes an effective network intrusion detection system based on ML and feature selection techniques. The performance measures are conducted four ML techniques RF, KNN, SVM, and DT for intrusion detection. In addition, feature selection techniques are used to identify essential features.

3.1. Research Design

Machine learning algorithms used in this part of this study are briefly summarized. In addition, in this section, performance criteria used in machine learning algorithms are given.

3.1.1 Decision Tree (DT)

The DT is a non-parametric supervised learning technique and one of the most influential classification techniques, which can be used for both classification and regression problems. The decision tree structure is like the tree structure but from top to bottom, where the highest node in the tree represents the root. Each internal node represents a test on a feature, each branch indicates the result of the test, and each leaf node indicates a class label [23]. A Classification and Regression Tree (CART) is used to detect cyberattacks that generate binary trees and uses a Gini index function as a method for feature selection for classification problems in Equation (1).

$$Gini = 1 - \sum_{i=1}^{n} (p_i)^2 \tag{1}$$

3.1.2 Random Forest (RF)

RF is a supervised ML technique that can be used for both classification and regression. Since it grows many decision trees rather than a single decision tree in the model, RF is an ensemble learner. It means more trees which generates a more robust classifier. RF generates several CART, in which each tree is trained on a randomly selected subset of the original data set. The decisions of all the decision trees generated within the forest are aggregated, and a vote makes the classifying decision of most of the trees [24].

3.1.3 Support Vector Machine (SVM)

SVM is one of the most potent supervised ML models used for classification and regression problems, but it is commonly used in classification. The work of the SVM technique is to classify data by defining a hyperplane or a line separating two classes within a data set. To find the best line to separate the data, SVM calculates the distance between the points of the two different classes and determines the points closest to each hyperplane class, which are called support vectors, where the most significant margin separating the hyperplane and the support vectors are chosen [25].

3.1.4 K-Nearest Neighbors (KNN)

The KNN is one of the most straightforward ML techniques that can be used for both classification and regression problems. The KNN technique assumes that convergent objects are the same. In other words, similar things are close to each other. To classify a new condition KNN technique calculates the distance between the item to be classified and all the training data items. Then the best value of K is determined, which is the number of nearest neighbors of the element to be classified [26]. Usually, several values are tried to determine the optimal value of k. The majority vote of the neighbors determines the result of the classification.

To measure the distance between two points, the KNN technique used Euclidean distance in Equation (2).

$$Euclidean\ distance(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + \cdots + (x_{in} - x_{jn})^2} \tag{2}$$

3.1.5 Performance Measurements

The performance of ML techniques was evaluated using five quality measures that are Accuracy, Precision, Sensitivity, Specificity, and F1-Score. Malicious Samples are considered positive and represented by ‘1’. While benign samples are considered negative and represented by '0’ [27]. All performance measure formulas are given in Equations (4-8).

- True Positives (TP): Malicious Samples have already been detected as malicious.
- True Negatives (TN): Benign Samples have already been detected as benign.
- False Positives (FP): Benign Samples have already been detected as malicious.
- False Negatives (FN): Malicious Samples have already been detected as benign.

$$\text{Accuracy} = \frac{(TN+TP)}{(TN+TP+FN+FP)} \tag{4}$$

$$\text{Precision} = \frac{TP}{(TP+FP)} \tag{5}$$

$$\text{Sensitivity} = \frac{TP}{(TP+FN)} \tag{6}$$

$$\text{Specificity} = \frac{TN}{(TN+FP)} \tag{7}$$

$$\text{F1 Score} = \frac{2*(\text{Precision} * \text{Sensitivity})}{(\text{Precision} + \text{Sensitivity})} \tag{8}$$

3.2. Data

In this study, the NSL-KDD dataset was used which is a clean and refined version of the University of New Brunswick KDD'99 dataset. A large amount of network traffic was collected to create the KDD dataset [28]. The NSL-KDD dataset consists of 42 features and 148,517 samples, these features are given in Table 1.

Table 1

Describes the features of the NSL-KDD data set

No	Features Names	No	Features Names	No	Features Names
1.	duration	**15.**	su attempted	**29.**	same_srv_rate
2.	protocol_type	**16.**	num_root	**30.**	diff_srv_rate
3.	service	**17.**	num_file_creations	**31.**	srv_diff_host_rate
4.	flag	**18.**	num_shells	**32.**	dst_host_count
5.	src_bytes	**19.**	num_access_files	**33.**	dst_host_srv_count
6.	dst_bytes	**20.**	num_outbound_cmds	**34.**	dst_host_same_srv_rate
7.	land	**21.**	is_host_login	**35.**	dst host_diff_srv_rate
8.	wrong fragment	**22.**	is_guest_login	**36.**	dst host same_src_port_rate
9.	urgent	**23.**	count	**37.**	dst_host_srv_diff_host_rate
10.	hot	**24.**	srv_count	**38.**	dst_host_serror_rate
11.	num_failed_logins	**25.**	serror_rate	**39.**	dst_host_srv_serror_rate
12.	logged_in	**26.**	srv_serror rate	**40.**	dst_host_rerror_rate
13.	num compromised	**27.**	rerror_rate	**41.**	dst host srv rerror rate
14.	root_shell	**28.**	srv_rerror rate	**42.**	class (malicious and benign)

The NSL-KDD dataset contains more than one type of cyber-attack, but this work focuses only on whether the traffic is malicious or benign in DDoS attacks. Fig. 2 shows the number of samples for the class type, malicious or benign. Briefly, this proposed study has been developed to detect malicious or benign traffic in DDoS attacks.

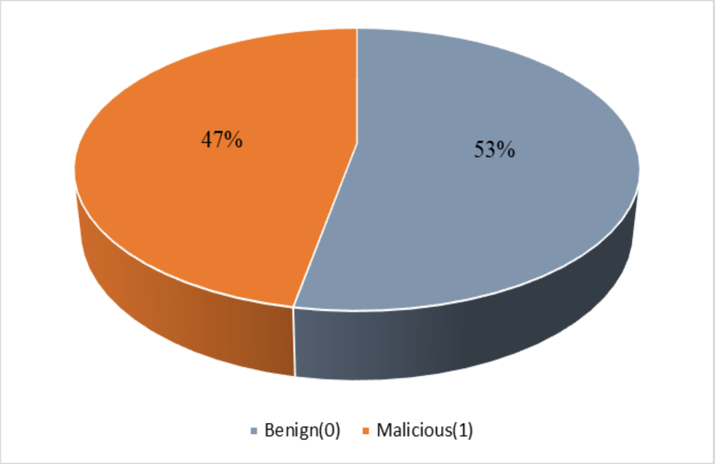

Figure 2

The percentage of each class

3.3. Proposed Methods

Data pre-processing is an important stage for ML techniques because the raw data often tends to be inconsistent and noisy and may contain missing, redundant, and irrelevant data. The efficiency of ML techniques depends mainly on the quality of the data provided [29]. So, to build a model with high performance and good accuracy, the pre-processing must be accurate. The pre-processing of NSL-KDD data in this study is summarized in the following steps. Figure 1 shows the proposed framework.

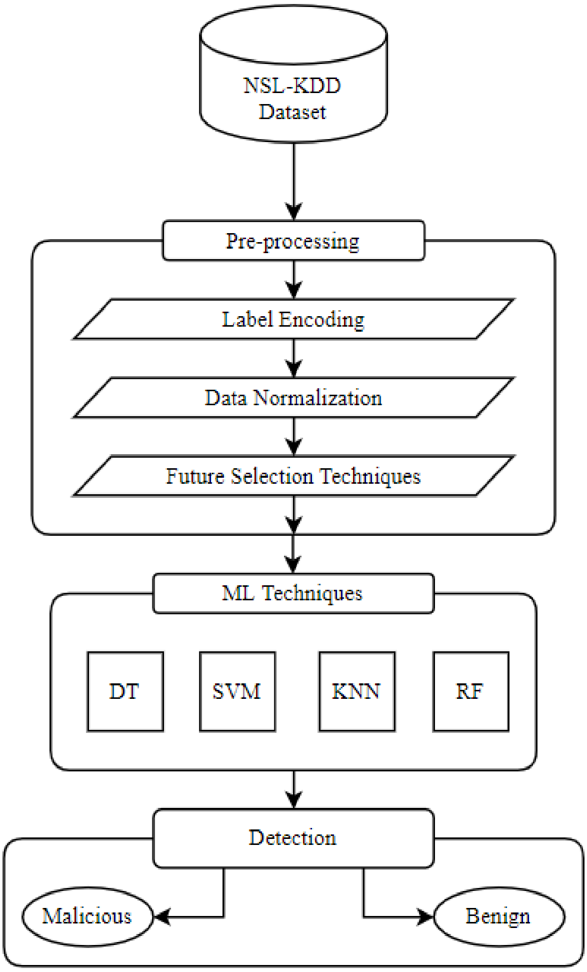

Figure 1
The proposed framework

3.3.1 Label Encoding

Label encoding converts categorical features to numbers in the NSL-KDD dataset. The NSL-KDD dataset contains three categorical features (protocol_type, service, and flag) that are converted to a number using the naming encoding method [30].

3.3.2 Data Normalization

The min-max normalization method was used to turn the numerical column values in the NSL-KDD dataset to a standard scale between 0 and 1 without distorting the value ranges. By using Equation (8) [31]:

$$Y = a - \min(a)/\max(a) - \min(a) \tag{8}$$

Where Y represents the normalized value, and *a* represents the original value.

3.3.3 Identifying Outliers in Regression (Cook's Distance)

Regression analysis helps to understand how variables in variables in groups of independent variables [32]. The aim of standardization is that the normal distribution of the values in our data set is symmetrical. Therefore, we will be able to detect whether the Mean, Mode, and Median data contain values close to each other [33]. We used Cook's distance, a successful method of detecting scaled changes in fit values, which is useful for identifying outliers (observations for predictive variables) of values in our dataset. An observation where Cook's distance is greater than three times the mean values may be an outlier. The Cook's distance of observation in Equation (9).

$$D_i = \frac{\sum_{j=1}^{n}(y_j - y_{j(i)})^2}{p^{MSE}} \tag{9}$$

Where:

- *yi* is the *j* th fitted response value
- *yj(i)* is the jth fitted response value, where the fit does not include observation *i*
- MSE is the mean squared error
- *p* is the number of coefficients in the regression model

3.3.4 Feature Selection Techniques

The proposed methodology was applied to the NSL-KDD dataset which has 41 features and one class attribute. After determining the importance value of each feature in the NSL-KDD dataset, the important features with a value greater than 0.05% were evaluated as shown in Table 2.

Table 2
The feature importance of the NSL-KDD dataset

No	**Features Names**	**Features Importance**	**No**	**Features Names**	**Features Importance**
1.	duration	1.14	**22.**	is_guest_login	<0.05
2.	protocol_type	13.44	**23.**	count	0.53
3.	service	1.93	**24.**	srv_count	0.66
4.	flag	56.96	**25.**	serror_rate	0.10
5.	src_bytes	0.74	**26.**	srv_serror rate	<0.05
6.	dst_bytes	0.68	**27.**	rerror_rate	<0.05
7.	land	<0.05	**28.**	srv_rerror rate	<0.05
8.	wrong fragment	1.38	**29.**	same_srv_rate	0.23
9.	urgent	<0.05	**30.**	diff_srv_rate	0.76
10.	hot	1.30	**31.**	srv_diff_host_rate	<0.05
11.	num_failed_logins	0.29	**32.**	dst_host_count	1.04
12.	logged_in	0.19	**33.**	dst_host_srv_count	0.80
13.	num compromised	<0.05	**34.**	dst_host_same_srv_rate	7.91
14.	root_shell	<0.05	**35.**	dst host_diff_srv_rate	0.40
15.	su attempted	<0.05	**36.**	dst host same_src_port_rate	1.73
16.	num_root	<0.05	**37.**	dst_host_srv_diff_host_rate	3.11
17.	num_file_creations	<0.05	**38.**	dst_host_serror_rate	0.06
18.	num_shells	<0.05	**39.**	dst_host_srv_serror_rate	0.18
19.	num_access_files	<0.05	**40.**	dst_host_rerror_rate	3.07
20.	num_outbound_cmds	<0.05	**41.**	dst_host_srv_rerror_rate	0.10
21.	is_host_login	<0.05			

The critical feature selection that affects the classification result is essential for a more accurate classification process. One of the most effective ways to select features is to use a DT technique to determine the feature's importance [34]. The feature importance property of the DT technique is beneficial for determining the importance of each feature and its effect on the classification result [35], [36]. This study has been developed to detect malicious or benign traffic in DDoS attacks. The NSL-KDD dataset was generated from the KDD dataset.

4 Results and Discussion

The four ML (RF, DT, KNN, and SVM) techniques are built using the Scikit Learn library, one of the powerful libraries used to build and implement ML techniques and data preprocessing in Python. With the Cook's distance method, the data found more than 3 times the threshold value were investigated and necessary adjustments were made accordingly. Finally, the significance of the data was tested using both the T-test statistic and calculating the confidence intervals. It has been observed that our data are within the confidence interval.

The dataset used in this study was divided into 80% for training the models and 20% for testing. The performance of the four techniques was compared on only 25 features from the selected dataset using the feature selection technique. Among the techniques used, the RF technique achieved the highest accuracy, with 99.72%, superior to the rest of the techniques. While DT technique came in second with 99.51%, followed by the KNN technique with an accuracy of 99.42%, while the SVM technique achieved the lowest accuracy with 99.03%. Table 3 shows the performance comparison between the four ML techniques using 5 different performance measures. Also, Figures 3 to 6 show the confusion matrix of the four ML techniques.

Pre-processing and feature selection techniques are essential steps before implementing ML techniques. An exemplary implementation of data pre-processing and testing of critical features would increase the performance accuracy to approximately 45%. Therefore, in this study, the proposed model achieved promising results after selecting the important and influencing features and selecting the appropriate ML model.

Table 3

Performance evaluation of ML techniques

ML Techniques	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)	F1-Score (%)
DT	99.51	99.49	99.46	99.54	99.47
SVM	99.03	99.39	98.54	99.47	98.96
KNN	99.42	99.46	99.30	99.53	99.38
RF	99.72	99.84	99.56	99.85	99.70

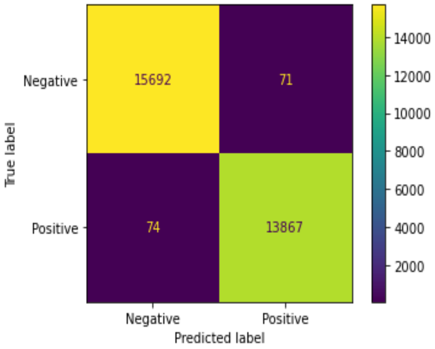

Figure 3

Confusion matrix of DT

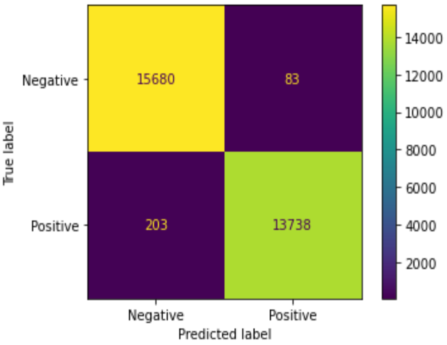

Figure 4

Confusion matrix of SVM

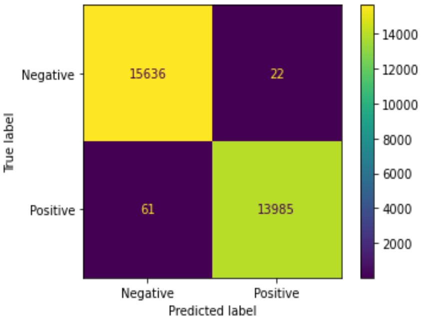

Figure 5
Confusion matrix of RF

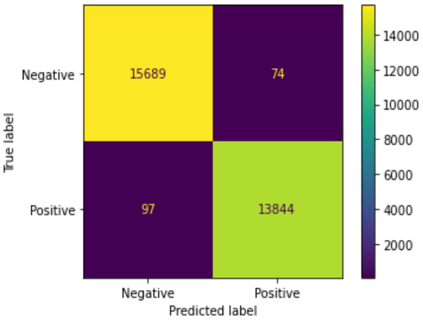

Figure 6
Confusion matrix of KNN

The RF technique used in this study was better than the results obtained from the RF technique used in other studies. In addition, the KNN, SVM, and DT techniques of this study performed better than KNN, SVM, and DT techniques. The recommended model, based on feature selection techniques and RF classifications, has achieved promising and reliable performance in the future to create identities based on ML techniques. In summary, it performed better than all studies mentioned in the study of the proposed model. Also, a higher accuracy ratio was obtained than all the studies mentioned in Table 4.

Table 4
Comparing the performance of the proposed model with related works

Ref.	Year	Dataset	ML Techniques	Best ML Technique	Best Accuracy
[15]	2019	CIC IDS	RF, LR, NB, KNN, Linear SVM, and LDA	RF	96.50%
[16]	2019	NSL-KDD	Ensemble model, MLP, SVM, KNN, and DT	Ensemble model	99.10%
[17]	2020	CICIDS2017 and NSL-KDD	AE+ SVM	AE+ SVM	96.36%
[18]	2020	CICIDS2017 and NSL-KDD	AE+DNN	AE+DNN	98.43%
[19]	2020	NSL-KDD	CNN and RNN	CNN	97.01%
[20]	2020	NSL-KDD	CNN	CNN	99.30%
[11]	2021	NSL-KDD	CNN, LSTM, and CLSTMNet	CLSTMNet	99.28%
[21]	2021	UGR16 and the UNSW-NB15, and KDD99	DNN	DNN	99.70%

[16]	2022	NSL-KDD	KNN, SVM, DT, and LR	LR	90.4%
Our Study		**NSL-KDD**	**RF, KNN, SVM, and DT**	**RF**	**99.72%**

Conclusions

There has been a significant increase in cyber-attacks, targeting organizations, institutions and even individuals. With the tremendous technological developments, the skill used by attackers, has increased and traditional IDS, can no longer detect sophisticated cyber-attacks. This required finding new, advanced tools to detect these destructive and expensive attacks. After the great successes of ML and DL techniques, in various fields, there have been many studies that use ML techniques in building IDS systems. This study presents an IDS based on feature selection techniques and ML techniques, for intrusion detection. The proposed model achieved promising results, as the RF technique achieved an accuracy of 99.72%, superior to other techniques in this work and related works. Having an intelligent system capable of detecting intrusion, helps significantly in maintaining the privacy and security of users. In this work, the focus is only on whether the traffic is malicious or benign. Future work could be developed to classify the different types of cybersecurity attacks. Classification accuracy can also be improved, by using ensemble methods, that combine more than one individual classifier.

References

[1] Dasgupta, D. et al.: Machine learning in cybersecurity: a comprehensive survey, *The Journal of Defense Modeling and Simulation*, 2022, Vol. 19, No. 1, pp. 57-106

[2] Al-Garadi, M. A. et al.: A survey of machine and deep learning methods for internet of things (IoT) security, *IEEE Communications Surveys & Tutorials*, 2020, Vol. 22, No. 3, pp. 1646-1685

[3] Salih, A. et al.: A Survey on the Role of Artificial Intelligence, Machine Learning and Deep Learning for Cybersecurity Attack Detection, *2021 7th International Engineering Conference "Research & Innovation amid Global Pandemic" (IEC)*, Erbil, Iraq, February 2021, pp. 61-66

[4] Zeebaree, S. R. et al.: Impact analysis of SYN flood DDoS attack on HAProxy and NLB cluster-based web servers, Indones, *J. Electr. Eng. Comput. Sci*, 2020, Vol. 19, No. 1, pp. 510-517

[5] Henry, A. Gautam and S.: Intelligent Intrusion Detection System Using Deep Learning Technique. In: *Computing, Communication and Learning: First International Conference, CoCoLe 2022, Warangal, India, October 27-29, 2022, Proceedings*, Cham: Springer Nature Switzerland, 2023, pp. 220-230

[6] Tong, W. et al.: A survey on intrusion detection system for advanced metering infrastructure, *In: Sixth international conference on instrumentation & measurement, computer, communication and control (IMCCC)*, IEEE, Harbin, China, July 2016, pp. 33-37

[7] Cloud Attack: Economic Denial of Sustainability (EDoS). Accessed: May 4, 2019 [Online] Available: http://www.elasticvapor.com/ 2009/01/cloud-attack-economic-denial-of.html

[8] AWS Said it Mitigated a 2.3 Tbps DDoS Attack, the Largest Ever. Accessed Jun. 30, 2020 [Online] Available: https://www.zdnet.com/ article/aws-said-it-mitigated-a-2-3-tbps-ddos-attack-the-largest-ever

[9] Academic Research Reports Nearly 30,000 DoS Attacks Per Day, Accessed Dec. 16, 2019 [Online] Available: https://www.corero.com/blog/853-academic-research-reports-nearly-30000-dos-attacks-per-day

[10] Halbouni, A. et al.: Machine Learning and Deep Learning Approaches for CyberSecuriy: A Review, *IEEE Access*, 2022, Vol. 10, pp. 19572-19585

[11] Issa, A. S. A. and Albayrak, Z.: CLSTMNet: A Deep Learning Model for Intrusion Detection, *In Journal of Physics: Conference Series*, 2021, Vol. 1973, No. 1, pp. 012244

[12] Abdullahi, M. et al.: Detecting cybersecurity attacks in the internet of things using artificial intelligence methods: A systematic literature review. *Electronics*, 2022, Vol. 11, No. 2, 198

[13] Salih, A. A. and Abdulrazaq, M. B.: Combining Best Features Selection Using Three Classifiers in Intrusion Detection System, *2019 International Conference on Advanced Science and Engineering (ICOASE)*, Zakho - Duhok, Iraq, 2019, pp. 94-99

[14] Kaggle, Accessed Dec. 16, 2019 [Online] Available: https://www.kaggle.com/datasets/hassan06/nslkdd

[15] Bindra, N. and Sood, M.: Detecting DDoS attacks using machine learning techniques and contemporary intrusion detection dataset. *Automatic Control and Computer Sciences*, 2019, Vol. 53, pp. 419-428

[16] Chavan, N., et al.: "DDoS Attack Detection and Botnet Prevention using Machine Learning," *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2022, pp. 1159-1163

[17] Das, S. et al.: DDoS Intrusion Detection Through Machine Learning Ensemble, *2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, Sofia, Bulgaria, 2019, pp. 471-477

[18] Kasim, Ö.: An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks. *Computer Networks*, 2020, Vol. 180, pp. 107390

[19] Bhardwaj, A. et al.: Hyperband tuned deep neural network with well posed stacked sparse autoencoder for detection of DDoS attacks in cloud, *IEEE Access*, 2020, Vol. 8, pp. 181916-181929

[20] Al-Emadi, S. et al.: Using deep learning techniques for network intrusion detection, *In 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, Doha, Qatar, 2020, pp. 171-176

[21] Abu Al-Haija, Q., and Zein-Sabatto, S.: An efficient deep-learning-based detection and classification system for cyber-attacks in IoT communication networks, *Electronics*, 2020, Vol. 9, No. 12, pp. 2152

[22] Larriva-Novo, X. et al.: An IoT-focused intrusion detection system approach based on preprocessing characterization for cybersecurity datasets, *Sensors*, 2021, Vol. 21, No. 2, pp. 656

[23] Ghiasi, M. M. et al.: Decision tree-based diagnosis of coronary artery disease: CART model, *Computer methods and programs in biomedicine*, 2020, Vol. 192, pp. 105400

[24] Nhat-Duc, H. and Van-Duc, T.: Comparison of histogram-based gradient boosting classification machine, random Forest, and deep convolutional neural network for pavement raveling severity classification. *Automation in Construction*, 2023, Vol. 148, pp. 104767

[25] Kurani, A. et al.: A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting, *Annals of Data Science*, 2023, Vol. 10, No. 1, pp. 183-208

[26] Surinta, O. et al.: Recognition of handwritten characters using local gradient feature descriptors. *Engineering Applications of Artificial Intelligence*, 2015, Vol. 45, pp. 405-414

[27] Azrour, M. et al.: Machine learning algorithms for efficient water quality prediction, Model. Earth Syst. Environ., 2022, Vol. 8, pp. 2793-2801

[28] Tavallaee, M. et al: A detailed analysis of the KDD CUP 99 data set, 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 2009, pp. 1-6

[29] Fung, W. K. et al.: Influence diagnostics and outlier tests for semiparametric mixed models, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, Vol. 64, No. 3, pp. 565-579

[30] Jackson, E. and Rajeev, A.: Performance evaluation of different feature encoding schemes on cybersecurity logs, 2019 SoutheastCon, Huntsville, AL, USA, 2019, pp. 1-9

[31] Singh, D. and Singh, B.: Investigating the impact of data normalization on classification performance, *Applied Soft Computing*, 2020, Vol. 97, pp. 105524

[32] Kannan, K. S. and Manoj, K.: Outlier detection in multivariate data, *Applied mathematical sciences*, 2015, Vol. 47, No. 9, pp. 2317-2324

[33] Choetkiertikul, M. et al.: A deep learning model for estimating story points, *IEEE Transactions on Software Engineering*, 2018, Vol. 45, No. 7, pp. 637-656

[34] Issa, A. S. A. and Albayrak, Z.: DDoS Attack Intrusion Detection System Based on Hybridization of CNN and LSTM, *Acta Polytechnica Hungarica*, 2023, Vol. 20, No. 2, pp. 105-123

[35] Sugumaran, V. et al.: Feature selection using decision tree and classification through proximal support vector machine for fault diagnostics of roller bearing. *Mechanical systems and signal processing*, 2007, Vol. 21, No. 2, pp. 930-942

[36] Özalp, A. N. and Albayrak, Z.: Detecting Cyber Attacks with High-Frequency Features using Machine Learning Algorithms, *Acta Polytechnica Hungarica*, 2022, Vol. 19, No. 7, pp. 213-233