International Journal of Innovative Research and Scientific Studies, 7(3) 2024, pages: 863-877

ISSN: 2617-6548

URL: www.ijirss.com

The development of a model for the threat detection system with the use of machine learning and neural network methods

Olga Ussatova[1,3], Aidana Zhumabekova[1,2*], Vladislav Karyukin[1,2], Eric T. Matson[4], Nikita Ussatov[1,5]

[1]*Institute of Information and Computational Technologies, Almaty, Kazakhstan.*
[2]*Al-Farabi Kazakh National University, Almaty, Kazakhstan.*
[3]*Almaty University of Power Engineering and Telecommunications named after G. Daukeyev, Almaty, Kazakhstan.*
[4]*Purdue University, West Lafayette, USA.*
[5]*Turan University, Almaty, Kazakhstan.*

Corresponding Author: Aidana Zhumabekova (*Email: zhumabekova2702@gmail.com*)

Abstract

This study examines the development of a model for the threat detection system with the use of machine learning and neural network methods. The fast development of Internet technologies has led to the appearance of many digital systems and platforms. However, despite the impressive technological progress, another side also emerged in the spread of a massive number of different cyber threats. Although various ways have been created to detect and prevent them, the threats are also developing and becoming more complex each year. Therefore, new system defense and data protection methods using machine and deep learning approaches have been proposed recently. The methods based on these approaches have proved to be especially effective in the wave of new Artificial Intelligence applications. In this paper, a threat detection system has been designed to disclose different kinds of threats while maintaining the security, confidentiality, and availability of the computer system. The development of machine learning models for detecting DDoS and man-in-the-middle attacks, Structured Query Language (SQL) injections, phishing, and malware was examined. The data scaling, feature selection, feature extraction, and classification steps were also thoroughly described. Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, XGBoost, CatBoost, and Deep Neural Network algorithms were utilized for training the cyber threat detection models. The experimental results evaluated all the models using accuracy, precision, recall, and F1-score metrics. The best models achieved scores in the range of 0.90 to 1.00.

Keywords: Artificial intelligence, Cyberattacks, DDoS, Defence system, Machine learning, Malware, Man-in-the-Middle, Neural networks, Phishing, SQL injection.

DOI: 10.53894/ijirss.v7i3.2957
Funding: This research is supported by the Institute of Information and Computational Technologies (Grant number: AP19675957).
History: Received: 13 October 2023/**Revised:** 3 January 2024/**Accepted:** 12 March 2024/**Published:** 28 March 2024

Authors' Contributions: Formed the main research objectives of the paper, prepared the text, and edited its final version, O.U.; did the research on threat and attack identification with the use of machine learning algorithms, prepared the text, and edited the paper, A.Z.; prepared the experiments' datasets, conducted them, and described them in the paper, V.K.; proposed the steps of applicability of machine learning models to the threat and attack identification system, checked the final version of the paper, and corrected it, E.T.M.; checked the experimental results, edited the configuration of the programs and systems, and checked the final version of the paper, N.U.All authors have read and agreed to the published version of the manuscript.

Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Institutional Review Board Statement: Not applicable.
Publisher: Innovative Research Publishing

1. Introduction

Today, digital systems and technologies are widespread in all aspects of life. Therefore, cyberattacks [1] occur very often. Detecting dangerous programs and protecting confidential information is always an urgent problem and a key asset for cybersecurity experts. The number of cybercriminals and types of threats [2] has grown significantly recently. Moreover, cyberattacks are becoming more challenging and complex than ever. In order to provide protection against such cyberattacks [3],cybersecurity [4] actions are aimed at protecting users, their information systems, networks, and programs.

Different measures have been taken to develop cybersecurity in Kazakhstan, including the concept called "Cyber Shield of Kazakhstan," which is implemented to solve the problem of cyberattacks. In addition, governmental and non-governmental organizations are working together to develop and enhance cybersecurity methods [5]. As part of the information security event, Kaspersky Lab experts studied Kazakhstan's most common cyber threats in 2022. Last year, the company's defense Web Solutions blocked 109,183,489 unique malicious objects. This year's analysis shows that the most common cyber threats include spam and malware attacks, such as phishing and malicious documents [6], spyware, and crypto miners.

Different types of threats and attacks exist. Among the most widespread are Denial of Service (DoS) and Distributed Denial of Service (DDoS), Man-in-the-Middle (MiTM), SQL injections, phishing, and malware. DoS and DDoS attacks are prevalent types of attacks. A DoS attack is a cyberattack that crashes a single computer or device by sending malicious files to the system, overloading the network, and making it almost entirely unavailable. By flooding the website with traffic, it is possible to achieve this goal while preventing it from responding to other legitimate users. A DDoS attack is carried out by simultaneously sending malicious data to the system through several devices. This type of attack is difficult to control and block because the attacker quickly sends a stream of traffic from multiple devices to the victims [7]. These attacks pose a significant risk for multiple services, as the assaults utilize various legitimate channels to send hundreds and thousands of messages, making them difficult to block. DoS and DDoS attacks are shown in Figure 1.

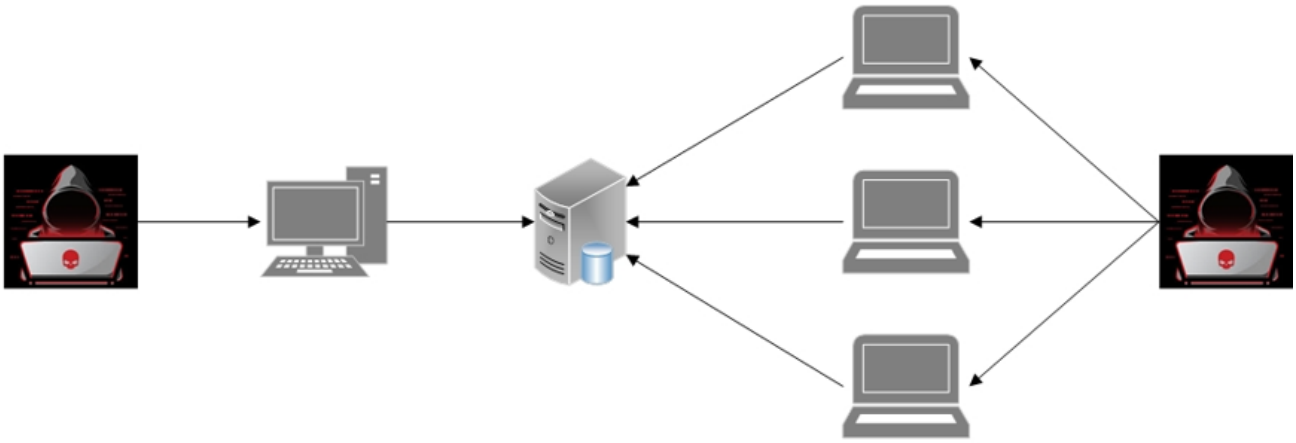

Figure 1.
Differences between DoS and DDoS attacks.
Note: Ussatova, et al. [7].

The MiTM attack is an attack where the intruder intercepts the communication between two sides, staying unnoticed by both of them. In this scenario, the attack is revealed only when the information is stolen [8]. The intruders can access information by staying in passive or active roles. Passive intruders quietly steal bank account data, bank card numbers, or other confidential information by being outside observers of the information. At the same time, an active attacker becomes a participant who emulates the system by changing the content of information, transferring illegal money, or pretending to be its legal participant. A web application or website user exchanges sensitive data without noticing the attack, pretending that a legitimate exchange of information is taking place. A MiTM attack is demonstrated in Figure 2.

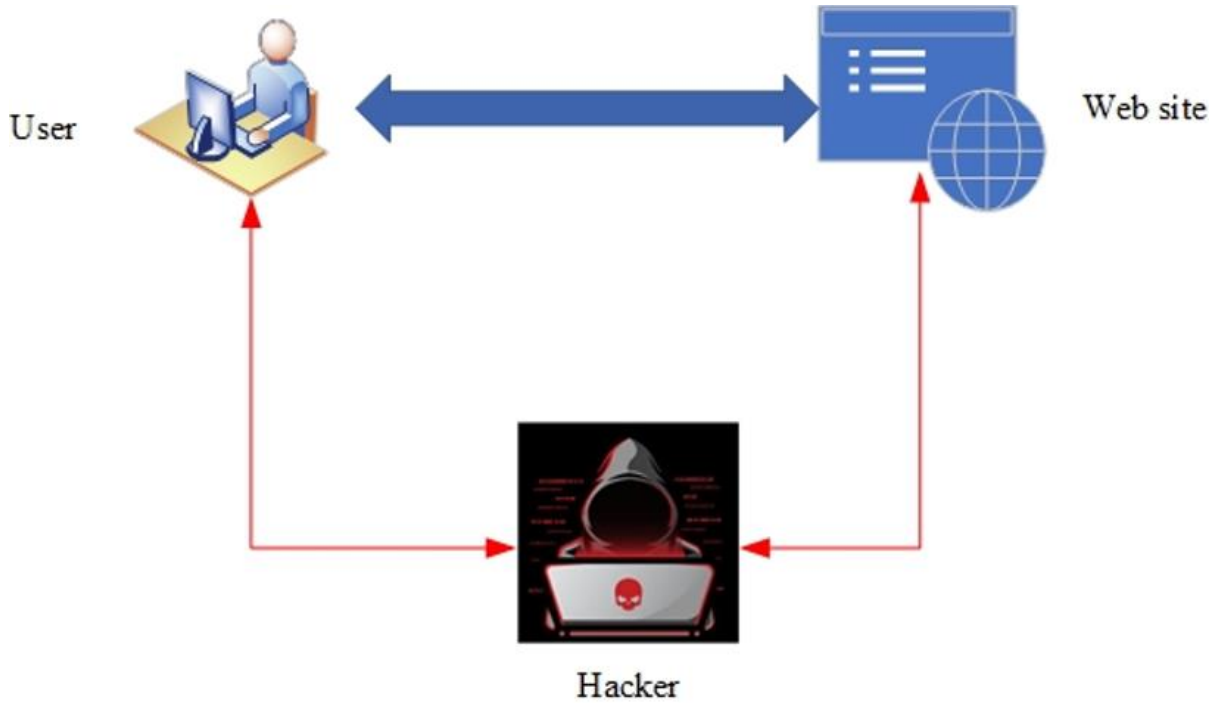

Figure 2.
A MiTM attack.
Note: Sivasankari and Kamalakkannan [8].

A phishing attack is one of the most common types of fraud used to access users' confidential data. In this type of attack, an intruder creates a malicious website that looks very similar to the legitimate one and sends the links via different communication channels. The forms of distribution of phishing attacks are various. In electronic phishing, intruders send emails andShort Message Service(SMS) with malicious links [9]. In search phishing, intruders design a non-legitimate website, creating a link leading to it. The links are also promoted in search engines with the use of common indexing mechanisms. If users open such links, they are directed to the specific website where scammers achieve their goal by gaining access to users' valuable data [10].

The SQL injection attack is one of the most common and dangerous cyberattacks carried out by cybercriminals for unauthorized access to the database management systems of web applications [11]. Intruders create dangerous SQL codes to access and manage sensitive information [12]. This type of attack can affect both the base structure and the data itself, including its consequences: disclosure, theft, modification, destruction of confidential data, and complete system hacking [13].

A program or piece of code that damages a computer system is called malware. Malware programs are usually spread via the Internet and removable devices, such as flash drives. They influence the systems by dropping the computer's performance, reducing itshard disk drive(HDD) and solid-state drive(SSD) free space, and popping up various advertisements on the screen. This situation obviously shows that the user's computer system is infected with malware. Dangerous malware continues to perform malicious actions, stealing files with sensitive data and hiding them inside the computer.

Generally, the number of cyberattacks is growing daily, and new effective methods and models must be developed to detect and prevent them successfully. This paper focuses on the machine learning (ML) approach for detecting cyber threats in a system. Section 2 observes related literature that describes the methods for detecting cyber threats. Section 3 describes the architecture of the system and the steps for developing ML models for detecting DDoS and MiTM attacks, SQL injections, phishing, and malware. Section 4 shows the experimental results of the ML model's development. Finally, Section 5 observes the full content of the paper and proposes the outlines for future works.

2. Related Works

Many works are devoted to developing effective Internet threat detection and information defense methods. In Biswas and Roy [14], there was a study of the detection of botnet threats using Deep Learning (DL) approaches, such as Artificial Neural Networks (ANN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) models. The experiments have shown that the GRU model is effective for processing large amounts of data, achieving excellent results with an accuracy of around 99%. In Zarandi and Sharifi [15], the authors used a deep neural network (DNN) to test the detection of cyberattacks in cyber-physical systems. In the experimental results, the 2-layer neural network (NN) allowed for an accuracy score of 80%-90%. The DL approach can make cybersecurity much simpler by detecting various threats. The authors of Poonguzhali, et al. [16] studied the method of revealing malicious software using convolutional neural networks (CNN). The dangerous code was transformed into grey-scaled images and classified with CNN, reaching an accuracy score of about 94.01%. In Cai and Li [17], the industrial Internet threat identification system was explored, where the main focus

was on analyzing the natural gas pipeline and wastewater treatment plant datasets. The implemented Recurrent Neural Network (RNN) model got an accuracy score of 99%. The paper bySstla, et al. [18] explored the use of the Support Vector Machine (SVM) and DNN models in Intrusion Detection Systems. SVM used different kernels in the designed models, and DNN implemented various activation functions. The radial basis function(RBF) Kernel for SVM achieved an accuracy score of 85%, while the sigmoid activation function for DNN gave an accuracy score of 83% to 97%. In Al Razib, et al. [19], the authors experimented with and conducted research on a DL software system to detect attacks in IoT. The proposed DNN-LSTM model showed high performance scores, reaching an accuracy value of 99.55%. The authors of the article,Safat, et al. [20], conducted an empirical analysis to detect and predict crime using various ML and DL methods. A lot of machine learning algorithms were used to sort the crime datasets from Chicago and Los Angeles. These included Naïve Bayes (NB), SVM, Decision Tree (DT), Random Forest (RF), Multilayer Perceptron and more. This helped get high scores for accuracy, precision, recall, and F1. Najafimehr, et al. [21]present a complex hybrid method that uses unsupervised and supervised ML approaches for detecting DDoS attacks. Thedensity-based spatial clustering of applications with noise(DBSCAN) clustering algorithm is applied to find benign and malicious traffic. When the data is labeled, the dataset is classified using such ML algorithms as RF and SVM. There was a performance improvement compared to NB, DT, RF, and SVM ML algorithms. The paper byGao, et al. [22] focuses on building an intrusion detection system using DL and association analysis. The binary classification models on the NSL-KDD dataset showed an accuracy score of around 80%, and multiclass classification models demonstrated an accuracy score of 76%. The authors of the paper [23], used different ML and DL models to detect DDoS attacks. The designed models allowed getting accuracy scores above 95%. In Karim, et al. [24] the combined hybrid Logistic Regression,SupportVector Machine, Decision Tree(LSD)model, which implemented Logistic Regression (LR), SVM, and DT algorithms, was proposed to prevent phishing attacks. This model reached a high accuracy score above 92%. The authors of the paper,Awajan [25] showed an intrusion detection system for IoT devices utilizing DL approaches. It detected various attacks, such as DDoS, sinkholes, and workholes, with a score of around 93%. Table 1 presents specifications, advantages, and drawbacks of other significant research papers in the field of threats and attack analysis with ML models.

Table 1.
Research works and their features.

Study	Specifications	Advantages and drawbacks
Nguyen and Le [26]	In this research, DoS and DDoS attacks on three datasets were evaluated. The classification was implemented with the newly developed Soft-ordering CNN (SOCNN) deep neural network.	The experimental results showed impressive F1-score values of 98.94%, 91.68%, and 96.07% for these three datasets. Despite the strength of the proposed model in detecting DoS and DDoS attacks, it is useful to check its performance on other types of attacks.
Elsayed, et al. [27]	This paper presents a secure automated system that uses a Long Short-Term Memory (LSTM) network. It focuses on the classification of DoS, DDoS, MitM, Password, and SQL Injection attacks. The scores of accuracy, precision, recall, and F1-score achieved values of 96.56%, 97.3%, 97.35%, and 97.4%, correspondingly.	The experiments of this research explored a large number of different threats, but they lacked focusing on real-world Internet of Things networks composed of mobile devices.
Bhayo, et al. [28]	The research of this paper analyzes the machine learning DDoS detection module. Three machine learning algorithms, Naïve Bayes, Decision Tree, and Support Vector Machine, were used for the dataset classification. They allowed to achieve accuracy score rates of 97.4%, 96.1%, and 98.1% for NB, SVM, and DT, respectively.	The classification models built in this research work received good accuracy scores, but the experiments could be strengthened with neural network models.

3. Materials & Methods

An Internet threat detection system is addressed to reveal various threats to the computer system, severely compromising its security, confidentiality, and availability. It uses the host and network devices and their configurations to detect different kinds of suspicious traffic incoming to the network. The system is aimed at being deployed for monitoring malicious attacks on the system's infrastructure. This system is designed to prevent various threats such as DDoS [29, 30]MiTM attacks [31], SQL injections [12, 32], phishing [33, 34] and malware [35] and requires training corresponding ML models built on high-quality datasets. The scheme of threat detection with ML models in server and network infrastructure is shown in Figure 3. In this system, threats from Internet sources try to breach the system by avoiding its firewall. When these attacks reach the web and database servers, ML models analyze the traffic and define whether it is benign or malicious.

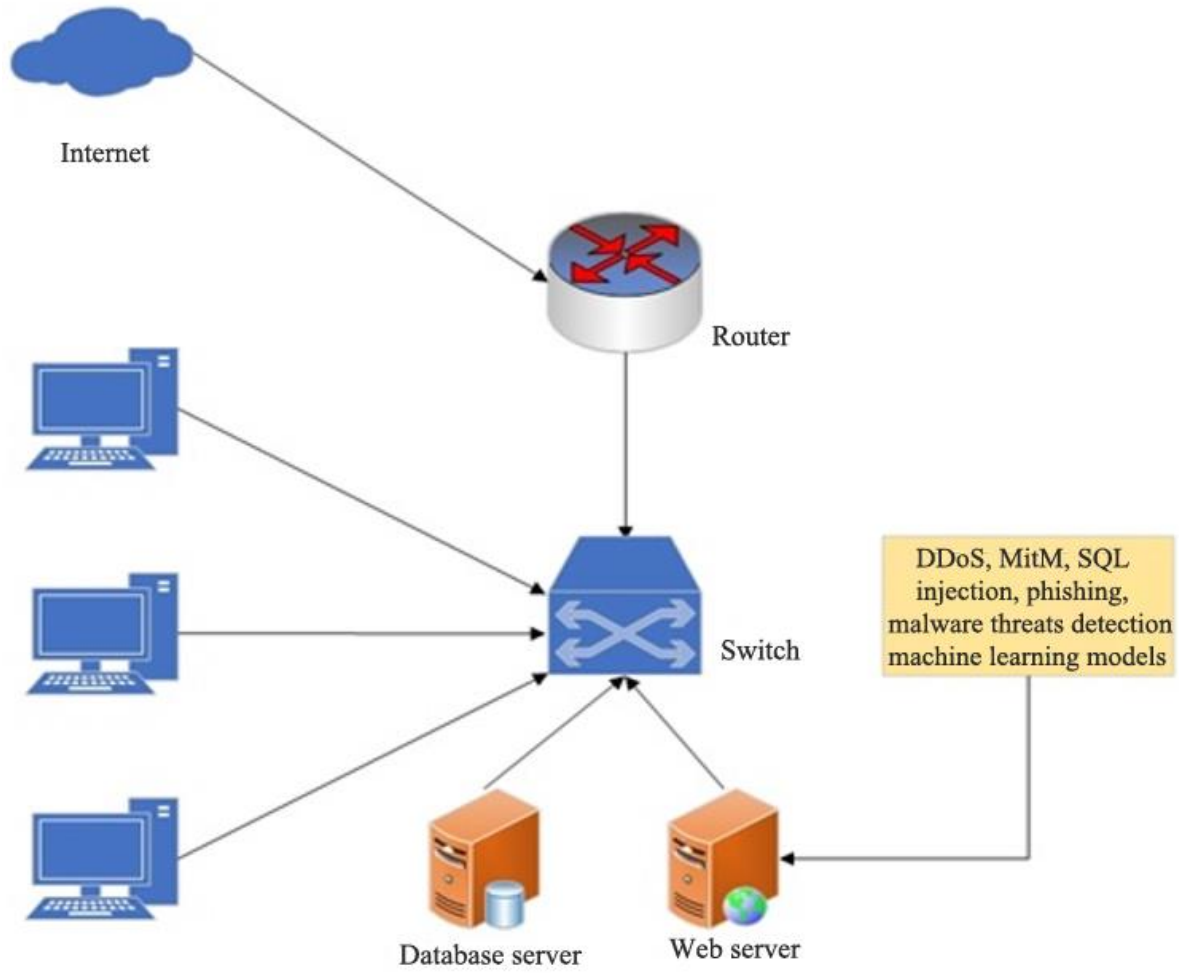

Figure 3.
Threat detection scheme.

The following steps are taken to train the models: dataset processing, data scaling, feature selection and feature extraction, and classification with ML algorithms (NB, LR, DT, RF, extreme gradient boosting (XGBoost), CatBoost, and DNN). The algorithm of the system is shown below, and the scheme of the steps is reflected in Figure 4.

1. Loading the dataset– df_data .
2. Scaling the dataset to make values of the dataset's features in the same range.

$$df\_data\_scaled = MinMaxScaler(df\_data)$$

3. Using the Chi-square feature selection technique to get the most important features for the DDoS, MiTM, phishing, and malware datasets models.

$$bestfeatures = SelectKBest(score\_func = chi2, k = 20)$$

Using the *tf-idf* feature extraction measure to vectorize sentences for the SQL injection dataset.

$$vectorizer = TfidfVectorizer()$$

4. Assigning a set of features and labels to variables.
Getting the best features for the DDoS, MiTM, phishing, and malware datasets models.

$$X = df\_data\_scaled[bestfeatures]$$

Extracting features for the models of the SQL injection dataset.

$$X = vectorizer.fit\_transform(df\_data\_scaled['Sentence'])$$

$$y = df\_data\_scaled['Label']$$

5. Splitting the dataset into training and testing parts.

$$x\_train, x\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.3)$$

6. Classifying with the ML algorithms.

$$ml\_classifier.fit(x\_train, y\_train)$$

$$y\_pred = ml\_classifier.predict(x\_test)$$

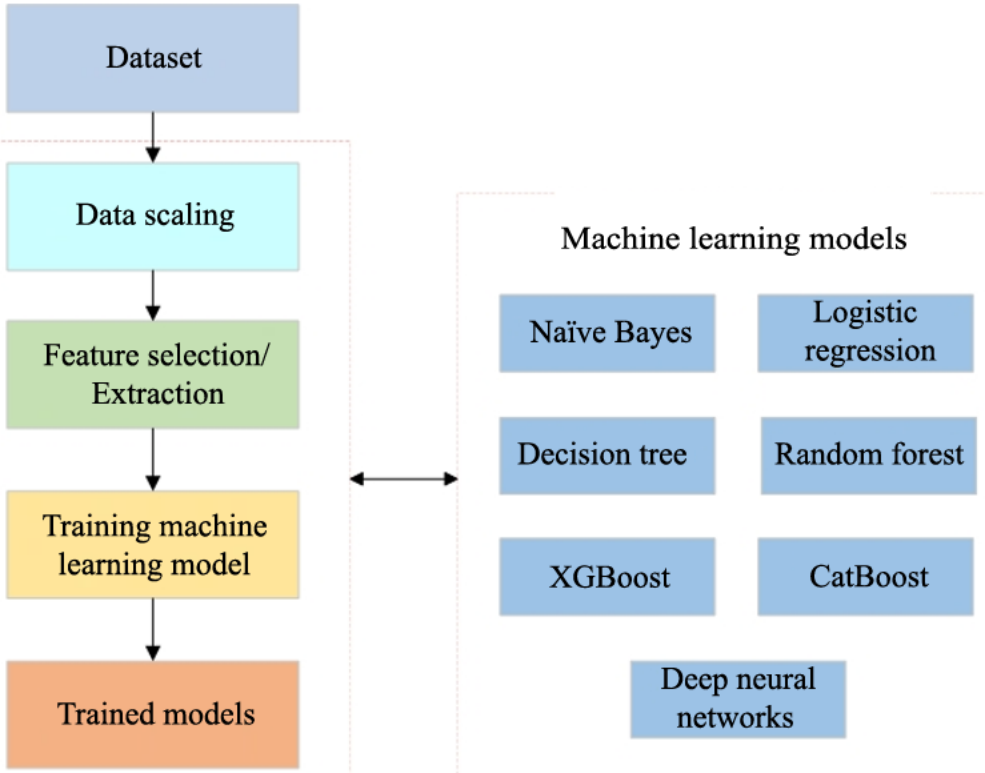

Figure 4.
Training ML models.

3.1. Datasets

Five datasets on the described threats were taken from the Kaggle website. A dataset containing many attacks that were received using various devices for detecting and generating DDoS traffic was used to determine DDoS attacks. This dataset includes 6,373,397 data points and 83 features. Among the most significant features are "Src Internet Protocol(IP)," "Src Port," "Dst IP," "Dst Port," "Protocol," and others. The dataset for detecting the MiTM attacks contains traffic from large commercial systems. As in the case of DDoS attacks, this dataset also includes 2,504,267 data points and 115 features. The dataset of phishing threats includes 5,000 legitimate and 5,000 phishing web pages, and a set of 48 features characterizes it.

The SQL injection dataset comprises safe and dangerous SQL commands, enabling data extraction from websites. It consists of 68,553 commands, each represented as a feature and a corresponding class label indicating its safety level (dangerous or safe). The malware dataset encompasses details about software specifically crafted to interfere, cause harm, or illicitly access computer systems. This dataset includes information on 216,351 programs, categorizing them as benign or hazardous, and is characterized by 53 features. Table2 presents the distribution of classes.

Table 2.
Threat class distribution.

Dataset	Legitimate	Malicious
DDoS	3,137,090	3,236,307
MiTM	1,358,995	1,145,272
SQL injection	44,632	23,921
Phishing	5,000	5,000
Malware	140,849	75,502

3.2. Data Scaling

Data scaling techniques are approaches that enable the normalization of differences among feature values, ultimately enhancing the performance of ML algorithms. Observing the scenario where two features exhibit ranges of values from 0 to 1 and 0 to 120,000, respectively, the utilization of these values can adversely affect the models, as one feature may disproportionately influence the outcomes. Applying scaling techniques addresses this problem, resulting in more accurate and balanced model predictions.

There are multiple scaling techniques called min-max scaling, mean scaling, and standardization. In the conducted experiments, the features of the datasets undergo min-max scaling, which involves applying a formula to obtain values within the range of [0,1].

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}, \quad (1)$$

Where x is an initial value and x' is a normalized value.

3.3. Feature Selection and Extraction

The input data on which ML models are trained is essential. However, when large datasets are gathered, they contain some features that do not significantly influence the performance of the models. Moreover, a massive amount of data can even delay the training process. The feature selection techniques are designed to solve this issue by reducing the number of dataset features and eliminating irrelevant data. There are different methods of feature selection, grouped into categories: Filter methods (Chi-Square, Information Gain, Fisher's Score), Wrapper methods (Forward Selection, Backwards Elimination), Intrinsic methods (Lasso and Ridge Regression), etc.

The Chi-square method is used as a feature selection technique in the experimental part. This method is a statistical test utilized to determine if there is a statistically significant association between two variables. The Chi-square method is well suited for selecting categorical features, especially when the target variable is categorical. It has the following characteristics:

- Ease of Use: The Chi-Square method is easy to apply to a dataset and does not require complex calculations or additional data processing.
- Selection of features based on their importance: The Chi-square method evaluates the importance of each feature by calculating the statistical relationship between the feature and the target variable. In this way, the most important features can be selected and the less significant ones excluded, which helps improve the model's performance.
- Model Independence: The Chi-Square method can be applied regardless of the ML model, making it versatile and easy to use.

Other feature selection techniques could be chosen. However, the following changes can occur in this way:

- Detection of new important features: Some methods can detect essential features that the Chi-square method missed. For example, methods based on mutual information can detect important interactions between features.
- Exclusion of less important features: If the Chi-square method includes features unrelated to the target variable, another method can exclude them, thereby improving the model's performance.
- More or less model complexity: Using a different feature selection method may result in a model with more or fewer features. It can affect the performance and complexity of the model.
- Changing data requirements: Some feature selection methods may require a specific data type or preprocessing.

It is important to note that no best feature selection method exists for all situations. Its choice depends on the nature of the data and the specific solved problem.

The Chi-square metric is calculated as follows:

$$x_c^2 = \sum \frac{(B_i - A_i)^2}{A_i}, \quad (2)$$

Where B is an observed value, A is an expected value, and c is a degree of freedom.

In the phishing dataset, the query sentences are required to be vectorized. For this purpose, the *tf-idf* measure is utilized. This measure is one of the most efficient and commonly used vectorization methods. *Tf* covers the occurrence of a single word in an SQL sentence command, and it includes two components: *tf* (term frequency) and *idf* (inverse document frequency). Thus, the importance of a word t_i within a single sentence is evaluated:

$$tf(t,s) = \frac{n_i}{\sum_{i=1}^{k} n_i}, \quad (3)$$

Where n_i is the number of occurrences of a word in a sentence, and the denominator is the total number of words in a sentence.

Idf is the inverse of the frequency where a certain word occurs in sentences. The implementation of *idf* reduces the weight of commonly used words. There is only one *idf* value for each unique word within a given set of sentences:

$$idf(t,S) = \log \frac{|S|}{|(s_i \supset t_i)|}, \quad (4)$$

Where $|S|$ is the number of sentences in corpora; $|(s_i \supset t_i)|$ is the number of sentences where t_i occurs.

After calculating the values of *tf* and *idf*, both parts are multiplied:

$$tf - idf = tf \times idf \quad (5)$$

3.4. Machine Learning Algorithms

Attack identification is usually associated with anomaly detection and classification tasks. ML models are utilized for classifying attacks and threats. The choice of the most appropriate ML model depends on the data's nature and the task's specific requirements. This research tests ML models such as NB, LR, DT, RF, *Extreme Gradient Boosting* (XGBoost), CatBoost, and DNN in the threat detection task. These algorithms proved to be effective in building advanced cybersecurity models in many research projects.

An NB [36] is one of the simplest ML algorithms. It considers each feature independent of others and implements the Bayesian formula to compute the conditional probability. The Bayesian formula is calculated as follows:

$$p(B \mid A) = \frac{P(A \mid B) \times P(B)}{P(A)}, \quad (6)$$

Where P(B) is a prior probability that the label is observed; P(A) is a prior probability that a feature has occurred; P(A|B) is a prior probability that a feature is classified as a label.

An LR [37] is another ML algorithm that uses a sigmoid function to predict the values of labels between 0 and 1. The formula for an LR is shown below:

$$p(x) = \frac{1}{1 + e^{-f(x)}}, \quad (7)$$

Where, $f(x) = w_0 + w_1 x_1 + \ldots + w_r x_r$ is a function and $w_0, w_1, \ldots, w_r$ are the corresponding weights.

DT and RF algorithms work well with large data sets and can handle categorical and numerical features. They also provide good model interpretability.

A DT [38] is one of the most efficient ML algorithms. It uses a tree structure with N nodes that contain conditions for classifying data points according to their features. In this structure, some feature is chosen on the first step, and all points that include it are put on one side, and points that do not include it are put on the other side. This procedure continues in cycles until the leaf nodes are reached. A DT is shown in Figure 5.

An RF [39] algorithm, designed as an ensemble learning mechanism, includes multiple decision trees. It proves to be very effective because not only does a single tree decide which class a new data point is assigned to, but a whole group of trees does it by the majority vote of all of them. An RF algorithm is shown in Figure 6.

Ensemble models such as (XGBoost) can perform very well by combining the power of several "weak" models. XGBoost [40] is an ensemble gradient-boosting algorithm that implements boosting classifiers where the succession modes lower the values of loss functions, reducing the errors of early-used models. The predictions of an ensemble algorithm are computed at each iteration. CatBoost [41] is another high-performance gradient-boosting algorithm based on decision trees. Yandex created it, and many ML tasks showed it to be effective.

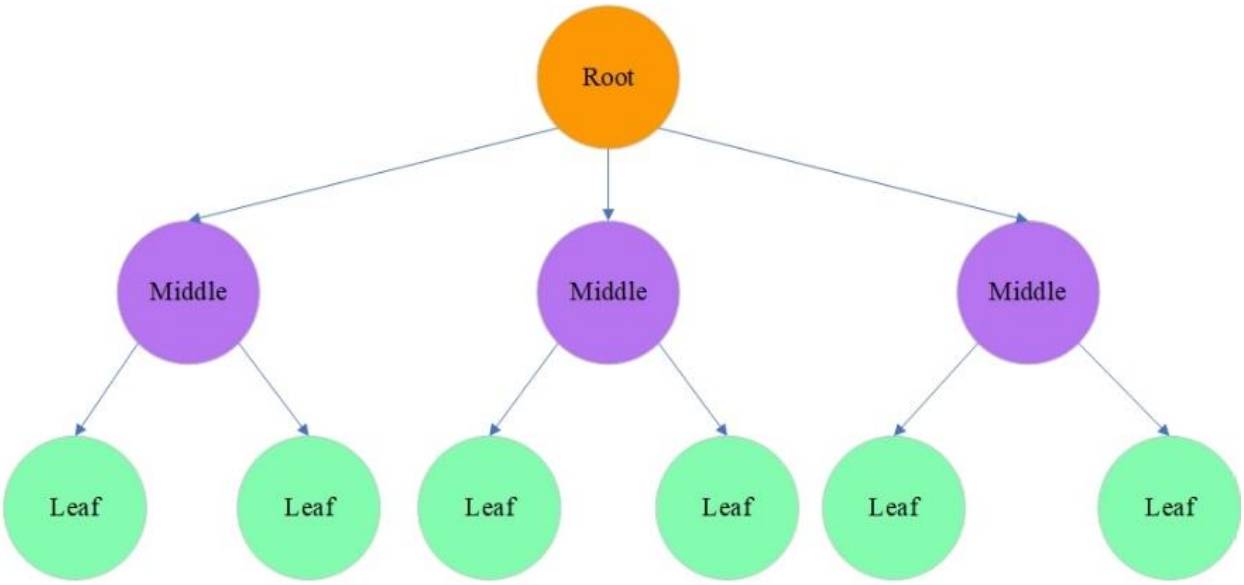

Figure 5.
A decision tree algorithm.
Note: Alshathri, et al. [38].

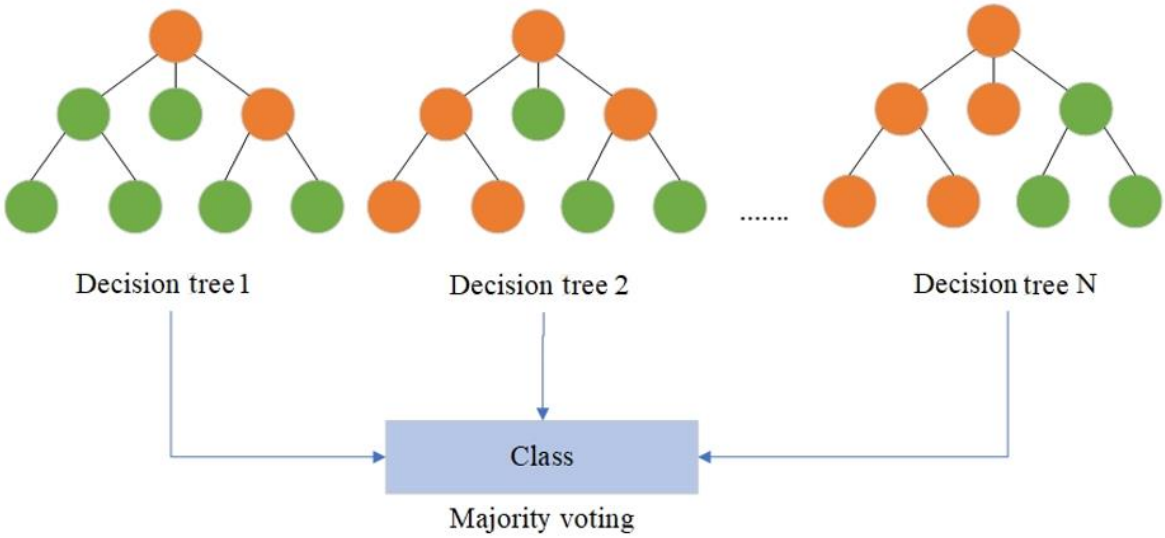

Figure 6.
A random forest algorithm.
Note: Li, et al. [39].

Artificial Neural Networks (ANN) [29] encompass a range of architectures, including Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN). These networks handle extensive datasets and uncover intricate patterns, making them particularly effective for tasks where manual interpretation of features is challenging. DNN is a specific model within the broader NN category, characterized by two or more hidden layers. This NN comprises an input layer for receiving input data, hidden layers with nodes referred to as neurons, and an output layer containing one or more neurons.DNN can be effective in detecting attacks on complex or large datasets. The scheme of DNN is shown in Figure 7.

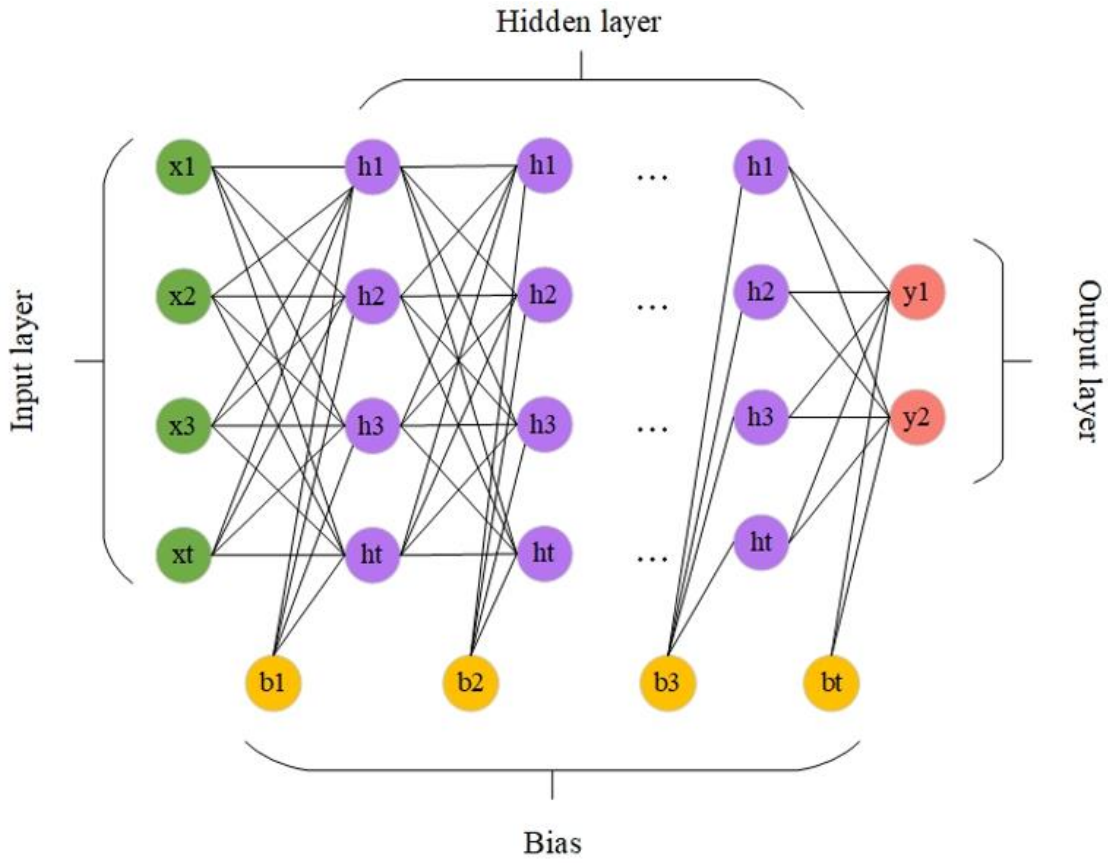

Figure 7.
Deep neural network.

In the DNN model, $x = x_1, x_2, ..., x_f$ is an input vector; $w_1, w_2, ..., w_i$ are weights connecting each layer; $b_1, b_2, ..., b_i$ are biases; y_1, y_2 are output values. The structure of the DNN model used in the experiments has the following form, shown in Table 3.

Table 3.
The structure of the DNN model.

Layer (type)	Output shape	Param #
dense_3 (Dense)	(None, 256)	5376
activation_3 (Activation)	(None, 256)	0
dropout_2 (Dropout)	(None, 256)	0
dense_4 (Dense)	(None, 128)	32896
activation_4 (Activation)	(None, 128)	0
dropout_3 (Dropout)	(None, 128)	0
dense_5 (Dense)	(None, 1)	129
activation_5 (Activation)	(None, 1)	0
Total params: 38401		
Trainable params: 38401		
Non-trainable params: 0		

It is important to consider that there is no best ML model for all tasks. Choosing the right model depends on the nature of the data, scenario, interpretability, and performance requirements.

Comparing different ML models involves looking at many different factors. Here are some key aspects that are commonly compared:

- Performance: One of the most important factors is the ability of the model to predict the results correctly. It can be measured using accuracy, precision, recall, F1-score metrics, area under thereceiver operating characteristiccurve (AUC-ROC), and others, depending on the task.
- Learning Rate and Prediction: Some models learn and make predictions faster than others. It can be critical in real-time scenarios.
- Handling missing data: Some models may be better for handling data with missing values.
- Data processing requirements: Some models require the data to be scaled or normalized before training, while others may operate on raw data.
- Interpretability: Some models, such as DT, are easier to interpret than others, such as NN.
- Overfitting resistance: Some models, such as RF, have built-in mechanisms that help prevent overfitting.
- Resilience to imbalanced classes: Classes can be highly imbalanced in some tasks, such as anomaly detection or attack identification. Some models may be better at solving these problems than others.

Depending on specific applications and requirements, some factors may be more important than others. Testing several models to determine which best suits the needs is generally recommended.

4. Results

The experiments were conducted on the workstation with the following specifications: Core i7 4790K, 32 GB RAM, 1 TB SSD, and NVIDIA GeforceRay Tracing extreme(RTX) 2070 Super.

Five datasets containing DDoS, MiTM, SQL injection, phishing, and malware threats were processed for classification models. The categorical features of these datasets were encoded, and the Min-Max scaler was applied to all features. The dimensions of the datasets were reduced with the Chi-square technique, leaving the twenty most important features. The SQL injection dataset's commands were vectorized with the *tf-idf* measure. All the datasets were split into 70% training and 30% testing parts. Then, they were classified with seven presented ML algorithms.

The performance was evaluated by accuracy, precision, recall, and F1-score measures:

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \quad (8)$$

$$precision = \frac{TP}{TP + FP}, \quad (9)$$

$$recall = \frac{TP}{TP + FN}, \quad (10)$$

$$F1\_score = 2\frac{precision \times recall}{precision + recall}, \quad (11)$$

where*TP* represents a positive instance that is correctly classified; *TN* represents a negative instance that is correctly classified; *FP* represents a positive instance that is incorrectly classified; and*FN* represents a negative instance that is incorrectly classified.

The classification results are presented in Table 4.

Table 4.
The classification of datasets.

Datasets	NB	LR	DT	RF	XGBoost	CatBoost	DNN
DDoS							
Accuracy	0.90	0.98	1.00	1.00	1.00	1.00	1.00
Precision	0.89	0.97	1.00	1.00	1.00	1.00	1.00
Recall	0.91	0.99	1.00	1.00	1.00	1.00	1.00
F1-score	0.90	0.98	1.00	1.00	1.00	1.00	1.00
MiTM							
Accuracy	0.68	0.98	1.00	1.00	1.00	1.00	0.98
Precision	1.00	0.97	1.00	1.00	1.00	1.00	0.97
Recall	0.30	0.99	1.00	1.00	1.00	1.00	0.99
F1-score	0.46	0.98	1.00	1.00	1.00	1.00	0.98
SQL injection							
Accuracy	0.97	0.99	0.99	0.99	0.99	0.99	0.99
Precision	0.98	0.99	0.99	0.99	0.99	0.99	0.99
Recall	0.94	0.97	0.98	0.98	0.98	0.98	0.98
F1-score	0.96	0.98	0.99	0.99	0.99	0.99	0.99
Phishing							
Accuracy	0.83	0.91	0.96	0.97	0.97	0.97	0.96
Precision	0.91	0.89	0.96	0.97	0.97	0.97	0.96
Recall	0.73	0.93	0.97	0.97	0.97	0.97	0.97
F1-score	0.81	0.91	0.96	0.97	0.97	0.97	0.96
Malware							
Accuracy	0.65	0.92	0.98	0.99	0.99	0.98	0.96
Precision	0.43	0.91	0.98	0.99	0.98	0.98	0.95
Recall	0.02	0.87	0.97	0.98	0.98	0.97	0.94
F1-score	0.04	0.89	0.97	0.98	0.98	0.98	0.94

The graphics of histograms and AUC-ROC curves are shown in Figures 8-12.

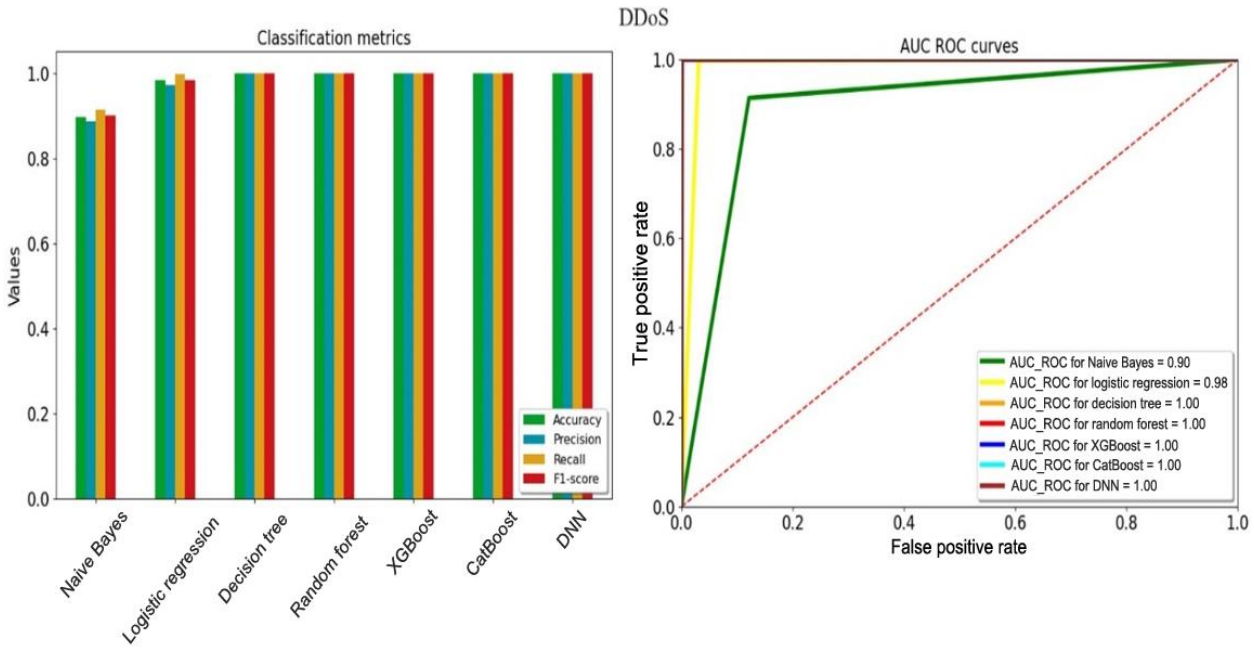

Figure 8.
Graphics for DDoS attacks.

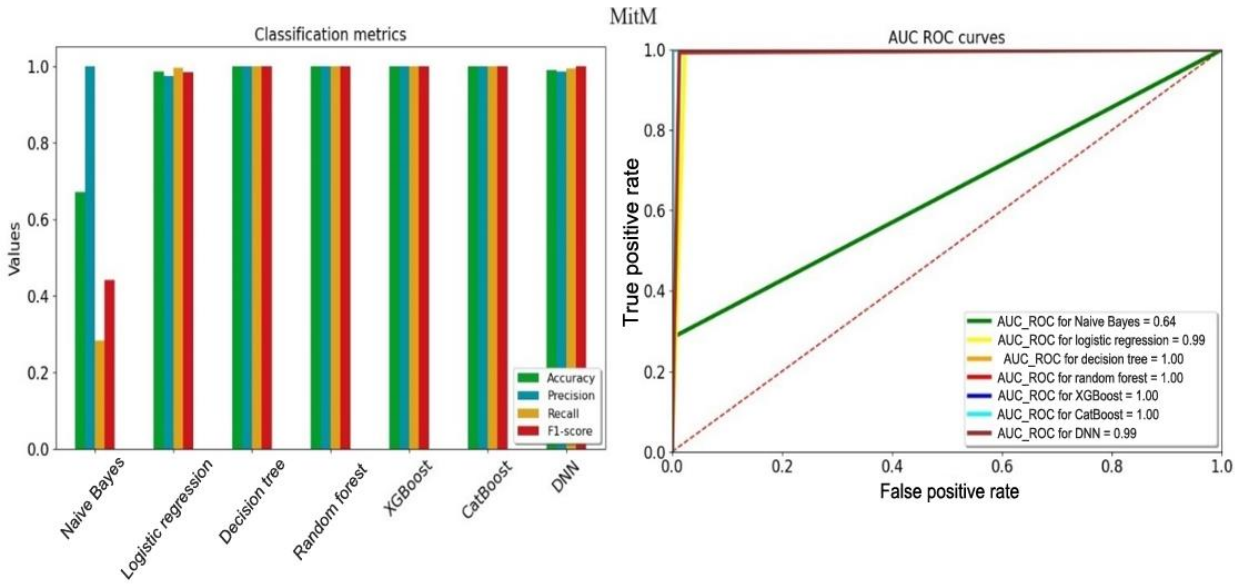

Figure 9.
Graphics for MiTM attacks.

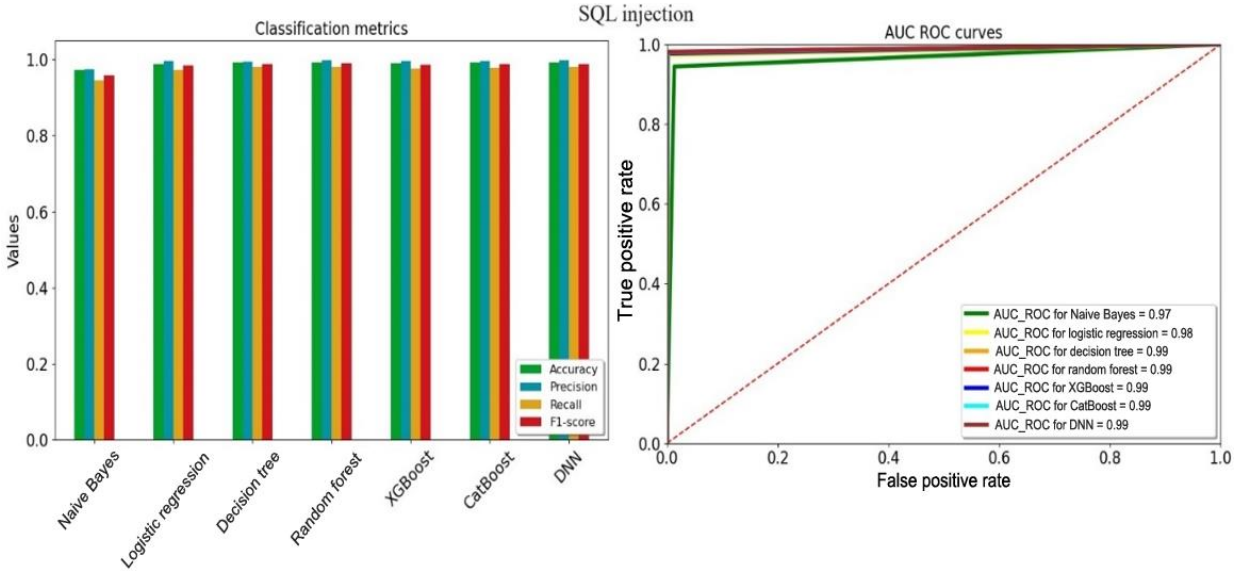

Figure 10.
Graphics for SQL injections

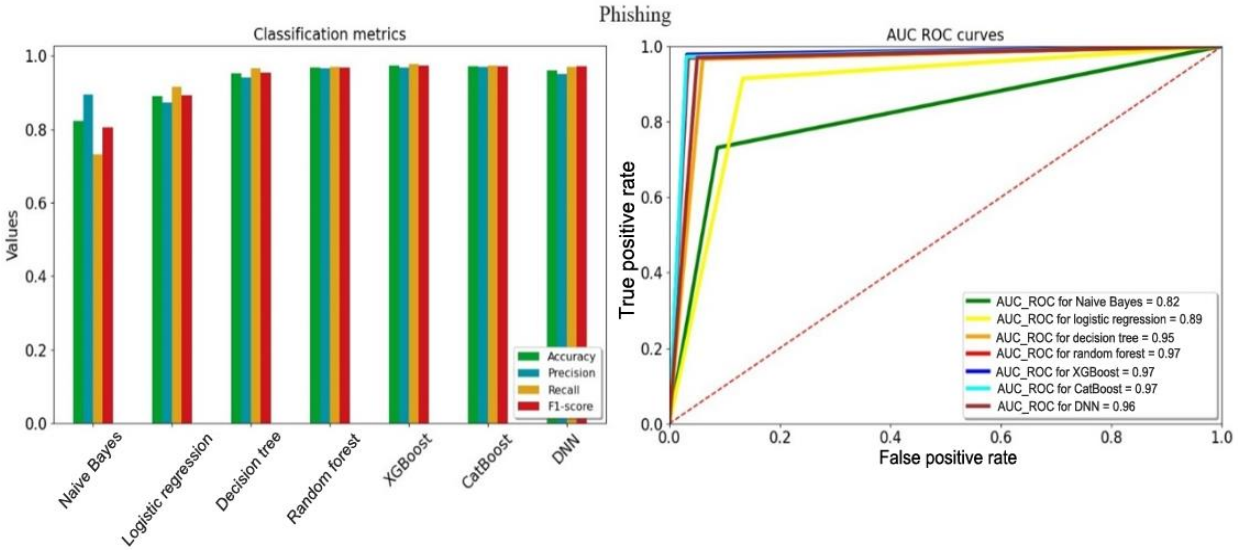

Figure 11.
Graphics for phishing.

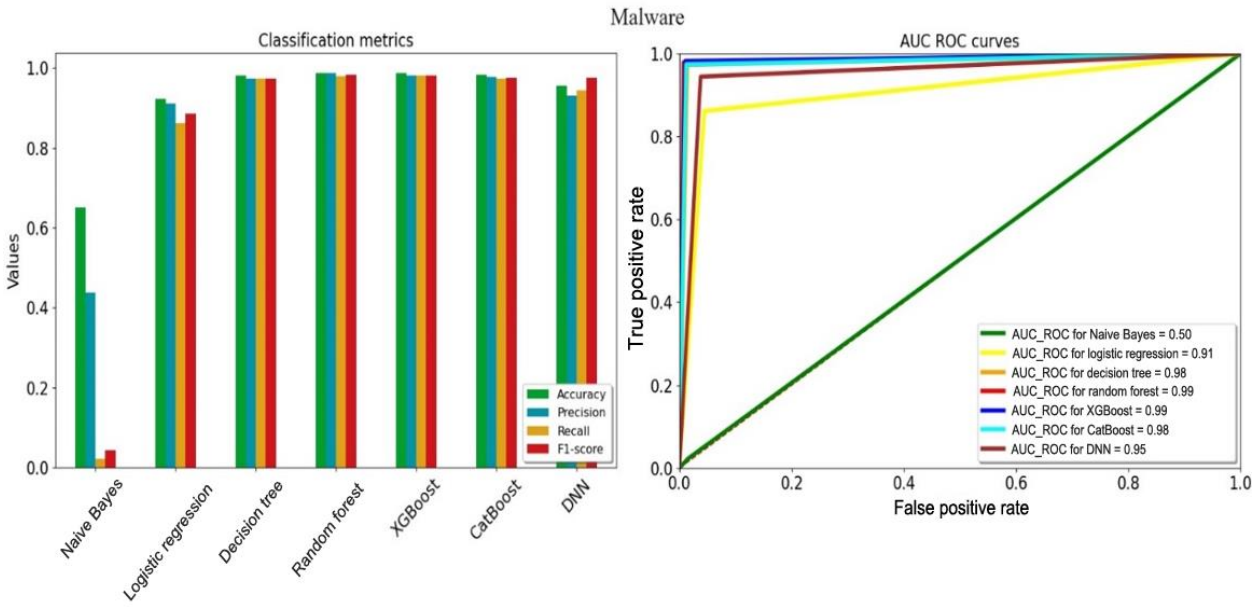

Figure 12.
Graphics for Malware.

The results of the classification models showed that most of the applied algorithms, except NB, demonstrated perfect scores, reaching values of accuracy, precision, recall, and F1-score in the range of 0.90 to 1.00. The achieved scores proved that the proposed models were effective in reaching high accuracy scores. The obtained metrics scores demonstrated the efficiency of the used ML models in the threat classification task. Although the accuracy values in the range of 95%-100% are suspicious, it must be noted that the experimental part included the results of the work of seven trained ML models that eliminated the possibility of overfitting. It also has to be stated that the datasets used in these experiments were well prepared and cleared of noise. In addition, DT, RF, XGBoost, and CatBoost mostly had the same scores. This shows that these models work well with data that has complex relationships and non-linearirties, and they can handle overfitting well by using ensemble methods or deep tree structure.

5. Conclusions

As digital systems have grown significantly in recent years, the number of threats occurring there has also multiplied. These different cyberattacks are becoming increasingly serious, bringing new challenges to defense systems. Old ways of protection are getting less relevant, requiring new intellectual approaches to build advanced ML models. Therefore, the greatest priority is directed towards building such systems that can be protected from various kinds of attacks and threats.

This paper proposes a system whose main defense is based on ML models to protect its servers from threats like DDoS, MiTM, SQL injections, phishing, and malware. The ML models were trained on a dataset containing benign and malicious data. In order to train high-quality ML models, the following steps, such as data scaling, feature selection, feature extraction, and classification with NB, LR, DT, RF, XGBoost, CatBoost, and DNN algorithms, were implemented: The developed models were evaluated using accuracy, precision, recall, and F1-score metrics. The LR, DT, RF, XGboost, CatBoost, and DNN allowed for scores of 0.90-1.00, which are excellent results for utilizing these models in the proposed system.

In future papers, it is planned to enhance security in more complex systems, using advanced intellectual approaches based on ML and NN models and extending the range of threats and attacks they can detect and prevent.

References

[1] M. Arya *et al.*, "Intruder detection in VANET data streams using federated learning for Smart City environments," *Electronics*, vol. 12, no. 4, p. 894, 2023. https://doi.org/10.3390/electronics12040894

[2] S. Mishra, A. Albarakati, and S. K. Sharma, "Cyber threat intelligence for IoT using machine learning," *Processes*, vol. 10, no. 12, p. 2673, 2022. https://doi.org/10.3390/pr10122673

[3] A. Alotaibi and M. A. Rassam, "Adversarial machine learning attacks against intrusion detection systems: A survey on strategies and defense," *Future Internet*, vol. 15, no. 2, p. 62, 2023. https://doi.org/10.3390/fi15020062

[4] N. Pattnaik, S. Li, and J. R. Nurse, "Perspectives of non-expert users on cyber security and privacy: An analysis of online discussions on twitter," *Computers & Security*, vol. 125, p. 103008, 2023. https://doi.org/10.1016/j.cose.2022.103008

[5] B. Yenlik, U. Olga, B. Rustem, and N. Saule, "Development of an automated system model of information protection in the cross-border exchange," *Cogent Engineering*, vol. 7, no. 1, p. 1724597, 2020. https://doi.org/10.1080/23311916.2020.1724597

[6] L. Á. Redondo-Gutierrez, F. Jáñez-Martino, E. Fidalgo, E. Alegre, V. González-Castro, and R. Alaiz-Rodríguez, "Detecting malware using text documents extracted from spam email through machine learning," in *Proceedings of the 22nd ACM Symposium on Document Engineering*, 2022, pp. 1-4.

[7] O. Ussatova, A. Zhumabekova, Y. Begimbayeva, E. T. Matson, and N. Ussatov, "Comprehensive DDoS attack classification using machine learning algorithms," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 577-594, 2022. https://doi.org/10.32604/cmc.2022.026552

[8] N. Sivasankari and S. Kamalakkannan, "Detection and prevention of man-in-the-middle attack in iot network using regression modeling," *Advances in Engineering Software*, vol. 169, p. 103126, 2022. https://doi.org/10.1016/j.advengsoft.2022.103126

[9] X. Xiao *et al.*, "Phishing websites detection via CNN and multi-head self-attention on imbalanced datasets," *Computers & Security*, vol. 108, p. 102372, 2021. https://doi.org/10.1016/j.cose.2021.102372

[10] A. Balogun *et al.*, "Improving the phishing website detection using empirical analysis of function tree and its variants," *Heliyon*, vol. 7, no. 7, pp. e07437-e07437, 2021. https://doi.org/10.1016/j.heliyon.2021.e07437

[11] G. Aliyu, G. Thandekkattu, I. Abdulmumin, U. A. Baba, A. B. Yusuf, and M. Nasir, "Machine learning based intrusion detection on complex nested transactional SQL queries," presented at the 2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON), 2022.

[12] W. B. Demilie and F. G. Deriba, "Detection and prevention of SQLI attacks and developing compressive framework using machine learning and hybrid techniques," *Journal of Big Data*, vol. 9, no. 1, p. 124, 2022. https://doi.org/10.1186/s40537-022-00678-0

[13] M. Alghawazi, D. Alghazzawi, and S. Alarifi, "Detection of sql injection attack using machine learning techniques: A systematic literature review," *Journal of Cybersecurity and Privacy*, vol. 2, no. 4, pp. 764-777, 2022. https://doi.org/10.3390/jcp2040039

[14] R. Biswas and S. Roy, "Botnet traffic identification using neural networks," *Multimedia Tools and Applications*, vol. 80, pp. 24147-24171, 2021. https://doi.org/10.1007/s11042-021-10765-8

[15] Z. N. Zarandi and I. Sharifi, "Detection and identification of cyber-attacks in cyber-physical systems based on machine learning methods," presented at the 2020 11th International Conference on Information and Knowledge Technology (IKT), 2020.

[16] N. P. Poonguzhali, T. Rajakamalam, S. Uma, and R. Manju, "Identification of malware using CNN and bio-inspired technique," presented at the 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), 2019.

[17] J. Cai and Q. Li, "Machine learning-based threat identification of industrial internet," presented at the 2020 IEEE International Conference on Progress in Informatics and Computing (PIC), 2020.

[18] V. Sstla, V. K. Kolli, L. K. Voggu, R. Bhavanam, and S. Vallabhasoyula, "Predictive model for network intrusion detection system using deep learning," *Revue d'Intelligence Artificielle*, vol. 34, no. 3, pp. 323-330, 2020. https://doi.org/10.18280/ria.340310

[19] M. Al Razib, D. Javeed, M. T. Khan, R. Alkanhel, and M. S. A. Muthanna, "Cyber threats detection in smart environments using SDN-enabled DNN-LSTM hybrid framework," *IEEE Access*, vol. 10, pp. 53015-53026, 2022. https://doi.org/10.1109/access.2022.3172304

[20] W. Safat, S. Asghar, and S. A. Gillani, "Empirical analysis for crime prediction and forecasting using machine learning and deep learning techniques," *IEEE access*, vol. 9, pp. 70080-70094, 2021. https://doi.org/10.1109/access.2021.3078117

[21] M. Najafimehr, S. Zarifzadeh, and S. Mostafavi, "A hybrid machine learning approach for detecting unprecedented DDoS attacks," *The Journal of Supercomputing*, vol. 78, no. 6, pp. 8106-8136, 2022. https://doi.org/10.1007/s11227-021-04253-x

[22] M. Gao, L. Ma, H. Liu, Z. Zhang, Z. Ning, and J. Xu, "Malicious network traffic detection based on deep neural networks and association analysis," *Sensors*, vol. 20, no. 5, p. 1452, 2020. https://doi.org/10.3390/s20051452

[23] M. A. Al-Shareeda, S. Manickam, and M. Ali, "DDoS attacks detection using machine learning and deep learning techniques: Analysis and comparison," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 2, pp. 930-939, 2023. https://doi.org/10.11591/eei.v12i2.4466

[24] A. Karim, M. Shahroz, K. Mustofa, S. B. Belhaouari, and S. R. K. Joga, "Phishing detection system through hybrid machine learning based on URL," *IEEE Access*, vol. 11, pp. 36805-36822, 2023.

[25] A. Awajan, "A novel deep learning-based intrusion detection system for IOT networks," *Computers*, vol. 12, no. 2, p. 34, 2023. https://doi.org/10.3390/computers12020034

[26] X.-H. Nguyen and K.-H. Le, "Robust detection of unknown DoS/DDoS attacks in IoT networks using a hybrid learning model," *Internet of Things*, vol. 23, p. 100851, 2023. https://doi.org/10.1016/j.iot.2023.100851

[27] R. A. Elsayed, R. A. Hamada, M. I. Abdalla, and S. A. Elsaid, "Securing IoT and SDN systems using deep-learning based automatic intrusion detection," *Ain Shams Engineering Journal*, vol. 14, no. 10, p. 102211, 2023. https://doi.org/10.1016/j.asej.2023.102211

[28] J. Bhayo, S. A. Shah, S. Hameed, A. Ahmed, J. Nasir, and D. Draheim, "Towards a machine learning-based framework for DDOS attack detection in software-defined IoT (SD-IoT) networks," *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106432, 2023. https://doi.org/10.1016/j.engappai.2023.106432

[29] A. Mustapha *et al.*, "Detecting DDoS attacks using adversarial neural network," *Computers & Security*, vol. 127, p. 103117, 2023. https://doi.org/10.1016/j.cose.2023.103117

[30] T. E. Ali, Y.-W. Chong, and S. Manickam, "Comparison of ML/DL approaches for detecting DDoS attacks in SDN," *Applied Sciences*, vol. 13, no. 5, p. 3033, 2023. https://doi.org/10.3390/app13053033

[31] A. A. Alsulami, Q. Abu Al-Haija, A. Tayeb, and A. Alqahtani, "An intrusion detection and classification system for IoT traffic with improved data engineering," *Applied Sciences*, vol. 12, no. 23, p. 12336, 2022. https://doi.org/10.3390/app122312336

[32] R. Zuech, J. Hancock, and T. M. Khoshgoftaar, "A new feature popularity framework for detecting cyberattacks using popular features," *Journal of Big Data*, vol. 9, no. 1, p. 119, 2022. https://doi.org/10.1186/s40537-022-00661-9

[33] S. Meena and D. Pethalakshmi, "Web attack prediction using stepwise conditional parameter tuning in machine learning algorithms with usage data," *International Journal of Computer Networks & Communications*, vol. 14, no. 6, 2022. https://doi.org/10.5121/ijcnc.2022.14606

[34] R. Zieni, L. Massari, and M. C. Calzarossa, "Phishing or not phishing? A survey on the detection of phishing websites," *IEEE Access*, vol. 11, pp. 18499-18519, 2023. https://doi.org/10.1109/ACCESS.2023.3247135

[35] S. Maurya and A. Jain, "Malicious website detection based on URL classification: A comparative analysis," in *Proceedings of Third International Conference on Computing, Communications, and Cyber-Security: IC4S 2021*, 2022: Springer, pp. 249-260.

[36] V. Arya, A. A. D. Almomani, A. Mishra, D. Peraković, and M. K. Rafsanjani, "Email spam detection using naive bayes and random forest classifiers," presented at the International Conference on Cyber Security, Privacy and Networking, 2021.

[37] G. Mumtaz *et al.*, "Classification and prediction of significant cyber incidents (SCI) using data mining and machine learning (DM-ML)," *IEEE Access*, vol. 11, pp. 94486-94496, 2023. https://doi.org/10.1109/ACCESS.2023.3249663

[38] S. Alshathri, A. El-Sayed, W. El-Shafai, and E. Hemdan, "An efficient intrusion detection framework for industrial internet of things security," *Computer Systems Science and Engineering*, vol. 46, no. 1, pp. 819–834, 2023. https://doi.org/10.32604/csse.2023.034095

[39] Q. Li, L. Zhang, G. Zhang, H. Ouyang, and M. Bai, "Simultaneous detection for multiple anomaly data in internet of energy based on random forest," *Applied Soft Computing*, vol. 134, p. 109993, 2023. https://doi.org/10.1016/j.asoc.2023.109993

[40] J. B. Awotunde *et al.*, "An ensemble tree-based model for intrusion detection in industrial internet of things networks," *Applied Sciences*, vol. 13, no. 4, p. 2479, 2023. https://doi.org/10.3390/app13042479

[41] R. Zuech, J. Hancock, and T. M. Khoshgoftaar, "Investigating rarity in web attacks with ensemble learners," *Journal of Big Data*, vol. 8, no. 1, pp. 1-27, 2021. https://doi.org/10.1186/s40537-021-00462-6