аккумуляторлардың, күн батареяларының және басқа да технологиялардың өндірісіне қолданылады. Бұл зерттеу тұрақты болашақ үшін ВРМ әлеуетіне жарық түсіреді.

Түйін сөздер: электролиз, электрохимия, химия өндірісі, гидрометаллургия, экология.

Электрохимическое получение металлов из водного раствора их солей

*[1]*А.А.Канафин, [1]К.В.Манашева*

[1]Назарбаевская Интеллектуальная Школа химико-биологического направления (Алматы, Казахстан)

Аннотация

В основе работы электролизера лежит пропускание электрического тока через водный раствор, содержащий ионные формы металлов. Этот сложный процесс приводит к окислению атомов металла на аноде и одновременному восстановлению атомов того же металла на катоде. Это контролируемое электрохимическое взаимодействие приводит к распаду компонентов раствора на их элементарные составляющие. Потенциальные возможности получения металлов в водных растворах с помощью электролизеров распространяются на производство металлических материалов, аккумуляторов, солнечных батарей и другие технологии. Электролизеры являются перспективной областью исследований для получения металлов в водных растворах, предлагая эффективное использование возобновляемых источников энергии и уменьшение воздействия на окружающую среду. В данной статье рассматривается множество существующих технологий электролизеров, предназначенных для получения металлов в растворе, и их ключевая роль в современном ландшафте производства металлических материалов. В статье представлен обзор текущего состояния и будущих перспектив электрорафинирования свинца в Китае с акцентом на мембранные и биполярные технологии. Теоретическое моделирование, имитация процессов и экономический анализ Статистический анализ, анализ затрат и выгод Изучение хода исследований и потенциала применения технологии электролиза, включая энергоэффективность и воздействие на окружающую среду. Кроме того, исследовательские усилия направлены на разработку эффективных, экологически чистых систем электролиза, использующих возобновляемые источники энергии, такие как солнечная или ветровая энергия. Потенциальные возможности применения получения металлов с помощью электролиза распространяются на производство металлических материалов, аккумуляторов, солнечных батарей и других технологий. Это исследование проливает свет на потенциал ВРМ для более устойчивого будущего.

Ключевые слова: электролиз, электрохимия, химическое производство, гидрометаллургия, экология.

Поступила в редакцию: 10.02.2024
Одобрена: 11.03.2024
Первая публикация на сайте: 05.01.2025

МРНТИ: 81.93.29 https://doi.org/10.65247/3105-3432-2025-1.03

INTEGRATED COMPUTER NETWORK SECURITY SYSTEM: SVM-BASED INTRUSION DETECTION AND THREAT PREDICTION USING MACHINE LEARNING ALGORITHMS

*[1]*M.S.UTARBAYEVA, [1]M.A.MUKANOVA*

[1]Nazarbayev Intellectual School of Chemistry and Biology
(Almaty, Kazakhstan)
**utarbaeva_m0512@hbalm.nis.edu.kz, mukanova_m@hbalm.nis.edu.kz*

Abstract

With the rapid growth in the use of computer networks and the significant expansion of related applications, cybersecurity issues are becoming increasingly relevant. This paper will provide an overview of solutions to growing network security problems, followed by developing a tool for detecting and preventing cyber threats by analyzing network traffic data from the Security Information and Event Management System (SIEM). Using various machine learning algorithms, including SVM, KNN, Decision Tree, Random Forest, Gaussian Naive Bayes, XGBoost, and neural networks, the study provides accurate traffic classification and identifies potential threats. The neural network increases the accuracy of detecting complex threat models. The study uniquely combines a targeted application in the field of cybersecurity, a comprehensive comparison of models, and practical implementation to obtain accurate data. The results demonstrated using histograms and tables show the effectiveness of Random Forest and PCA Random Forest, emphasizing their accurate traffic classification. Finally, the efficacy of diverse experiments conducted on cyber-security data sets featuring multiple cyber-attack categories will be assessed. Additionally, the effectiveness of performance metrics such as precision, recall, and accuracy will be evaluated. Applying a multi-level approach aligned with the latest trends in machine learning in cybersecurity facilitates swift and precise threat analysis and response, consequently elevating the system's overall effectiveness.

Keywords: cybersecurity, machine learning, machine learning algorithms, intrusion detection, artificial intelligence, cyber-attack prediction.

Introduction

The escalating utilization of computer networks and the substantial proliferation of associated applications have led to rapid growth, making network security concerns progressively pertinent. Many systems have security weaknesses, which can lead to increased attacks with negative impact. Thus, the exact vulnerabilities discovered this way first become an essential factor. This research will be based on a developed and trained model using an SVM classifier to determine the correspondence in a network packet.

The work is aimed at reviewing the classifiers of cyber threats and then creating a tool for detecting and preventing cyber threats by analyzing network traffic based on data obtained from SIEM. The study employs machine learning algorithms to categorize network traffic and effectively discern potential attacks. The neural network improves the accuracy and reliability of detecting complex threat models. Research helps to enhance cybersecurity by providing fast and accurate threat analysis and response tools.

The uniqueness of the study is as follows:

Targeted application in the field of cybersecurity

The survey focuses on cybersecurity and is related to intrusion detection on the SIEM database, anomaly detection, or similar areas. This emphasis is significant in the digital era, given the rising importance of network security.

Comprehensive model comparison

It utilizes machine learning models such as logistic regression, K-nearest neighbors, the naive Bayesian method, decision trees, random forest, and XGBoost and meticulously compares their performance. Such a broad comparison is only sometimes standard in many projects focusing on more than one or two models.

Detailed performance analysis

A detailed assessment of the model's performance using indicators such as accuracy, precision, recall, ROC-AUC, and the importance of functions provides a deep understanding of the strengths and weaknesses of each model.

Custom visualization methods

Custom graphs and visualizations for comparing model performance indicators are informative and improve the interpretability of the results, which is crucial for stakeholders who may need to become more familiar with the intricacies of machine learning.

Data preprocessing and function development

The work includes specific data preprocessing steps and developing crucial functions in machine learning pipelines. How these steps are performed can significantly affect the model's performance.

Potential integration of best practices

The paper includes advanced techniques such as ensemble methods, hyperparameter tuning, or new feature selection methods, further contributing to its uniqueness.

Practical implementation for accurate data

The paper is designed to work with accurate, dynamic cybersecurity data. Such practical applicability adds significant value, especially in an area where data is constantly evolving.

Literature review

In cybersecurity, much literature has surfaced, highlighting the increasing importance of integrating machine learning algorithms into security models. Yavanoglu O. (2017) underscores the significance of utilizing real datasets in «Review on Cyber Security Datasets for Machine Learning Algorithms». This review emphasizes the pivotal role that authentic datasets play in enhancing the efficacy of security models.

Alqahtani H. (2020) delves into the escalating challenges posed by cyber threats in an interconnected world in the study titled «Cyber Intrusion Detection Using Machine Learning Classification». The author employs various machine learning classification algorithms to address the complex landscape of intrusion detection, providing insights into combating evolving cybersecurity issues. Handa A. (2019) contributes to the discourse with their work, «Machine Learning in Cybersecurity: A Comprehensive Review». Acknowledging the critical role of machine learning and deep learning methodologies, they categorize and elaborate on different algorithms to facilitate a better understanding and application of these techniques in cybersecurity. Dasgupta D. (2022) offers a comprehensive survey in their work titled «Comprehensive Survey of Machine Learning in Cybersecurity». Covering the basics of cyber attacks, defenses, commonly used ML algorithms, and proposed ML and data mining schemes, their overview provides a holistic understanding of the evolving landscape of cybersecurity. Buczak A.L. (2015) explicitly focuses on applying machine learning and data mining techniques in cyber analytics in the work «Survey of ML and DM Methods for Cyber Analytics». The survey explores utilizing these techniques to fortify intrusion detection capabilities, contributing insights into the evolving field of cyber analytics. Natarajan J. (2020) work, «Cyber-Secure Man-in-the-Middle Attack Intrusion Detection», hones in on network communication security. The chapter addresses concerns related to cybersecurity network attacks, with a specific emphasis on countering man-in-the-middle attacks using machine learning algorithms, contributing to the ongoing efforts to secure network communications effectively.

Methodology

A. Intrusion Detection System. An intrusion detection system (IDS) is designed. It accomplishes this by actively monitoring and analyzing the daily activities of computer systems, aiming to detect security issues and threats, including denial-of-service (DoS) attacks [Saker I.H., 2020]. The capability to identify various types of cyber-attacks and abnormalities in network behavior is imperative for enhancing overall system security. Therefore, developing an efficient Intrusion Detection System (IDS) is paramount in today's network security landscape.

Upon detecting suspicious behavior, an intrusion detection system (IDS) is activated to scrutinize network traffic at the specific location and subsequently issue an alert. This application has recently breached security policies or brought competitive activity to the network or system. The administrator is notified of priority actions or violations, or a centralized security and event management (SIEM) system will visit the site. In order to separate activity from false positives, the SIEM system gathers data from various sources and applies filtering procedures.

- Host-Based IDS (HIDS): Designed to defend an endpoint against both internal and external threats, a host-based IDS is installed on a specific endpoint. Such an IDS can watch active processes, examine system logs, and monitor network traffic to and from the device. Although the host machine is the only context in which a host-based intrusion detection system (IDS) can be seen, it nevertheless has extensive internal visibility into the host computer.

- Network-Based Intrusion Detection System (NIDS): A network-based IDS solution can observe an entire protected network. It can see every bit of data going over the network and decides what to do by looking at the contents and metadata of each packet. Though these systems lack internal visibility into the endpoints they defend, their broader perspective gives additional context and the capacity to detect widespread threats.

Detection Method of IDS:

- Signature-based Method: This particular intrusion detection system (IDS) discerns attacks by scrutinizing distinctive patterns within network traffic, including characteristics such as the volume of bytes, occurrences of 1s, or occurrences of 0s. Furthermore, it performs detection by analyzing the known malicious instruction sequences of malware. *Signatures* representing identified patterns are integral to intrusion detection systems (IDS). Signature-based IDSs promptly detect attacks with pre-identified patterns (signatures). However, they face challenges recognizing novel malware attacks, as their patterns (signatures) still need to be identified [Saranya T., 2020].

- Anomaly-based Method: Since new malware is created quickly, anomaly-based intrusion detection systems (IDS) were developed to detect unknown malware attacks [Saranya T., 2020]. A trustworthy activity model is created using machine learning in anomaly-based intrusion detection systems. Any new information is compared to this model and deemed suspicious if it does not match it. Machine learning-based techniques offer a universal feature set rather than signature-based IDS because their models can be tailored to specific hardware and application configurations.

B. Dataset. The data set taken from the SIEM (Security Information and Event Management) system contains 43 columns and 125971 rows. These results are intended to serve as a basis for future research in cybersecurity. Table 1 demonstrates the features of the dataset.

The complete data set, including attack type labels and difficulty level, is presented in text (.txt) format. The dataset possesses several advantages when compared to the original SIEM dataset:

The training set lacks redundant entries, preventing classifiers from exhibiting bias towards more commonly appearing records. The proposed test sets are accessible from duplicates, ensuring that the efficacy of learning algorithms remains unaffected by improved methods for identifying frequently occurring records. Records are chosen from each difficulty level group in inverse proportion to their representation in the original SIEM dataset percentage. This approach introduces a broader spectrum of the classification efficacy of different machine learning methods. This approach enhances the efficiency of evaluating different learning methods. The size of records in both the training and test sets is judiciously determined, enabling experiments to be conducted on the entire data set without resorting to random sampling of a smaller subset. This methodology ensures that the evaluation results across different research papers remain consistent and comparable.

Table 1

Features of dataset

№	Column	Non-Null Count	Dtype	№	Column	Non-Null Count	Dtype
1	duration	125972 non-null	int64	23	count	125972 non-null	int64
2	protocol_type	125972 non-null	object	24	srv_count	125972 non-null	int64
3	service	125972 non-null	object	25	serror_rate	125972 non-null	float64

4	flag	125972 non-null	object	26	srv_serror_rate	125972 non-null	float64
5	src_bytes	125972 non-null	int64	27	rerror_rate	125972 non-null	float64
6	dst_bytes	125972 non-null	int64	28	srv_rerror_rate	125972 non-null	float64
7	land	125972 non-null	int64	29	same_srv_rate	125972 non-null	float64
8	wrong_fragment	125972 non-null	int64	30	diff_srv_rate	125972 non-null	float64
9	urgent	125972 non-null	int64	31	srv_diff_host_rate	125972 non-null	float64
10	hot	125972 non-null	int64	32	dst_host_count	125972 non-null	int64
11	num_failed_logins	125972 non-null	int64	33	dst_host_srv_count	125972 non-null	int64
12	logged_in	125972 non-null	int64	34	dst_host_same_srv_rate	125972 non-null	float64
13	num_compromised	125972 non-null	int64	35	dst_host_diff_srv_rate	125972 non-null	float64
14	root_shell	125972 non-null	int64	36	dst_host_same_src_port_rate	125972 non-null	float64
15	su_attempted	125972 non-null	int64	37	dst_host_srv_diff_host_rate	125972 non-null	float64
16	num_root	125972 non-null	int64	38	dst_host_serror_rate	125972 non-null	float64
17	num_file_creations	125972 non-null	int64	39	dst_host_srv_serror_rate	125972 non-null	float64
18	num_shells	125972 non-null	int64	40	dst_host_rerror_rate	125972 non-null	float64
19	num_access_files	125972 non-null	int64	41	dst_host_srv_rerror_rate	125972 non-null	float64
20	num_outbound_cmds	125972 non-null	int64	42	outcome	125972 non-null	object
21	is_host_login	125972 non-null	int64	43	level	125972 non-null	int64
22	is_guest_login	125972 non-null	int64				
dtypes: float64(15), int64(24), object (4) memory usage: 41.3+ MB							

C. Preprocessing the data. Data preprocessing involves transforming a specific dataset's values to optimize the information acquisition and process. Typically, a significant disparity exists between the maximum and minimum values within the dataset. Normalizing the data reduces the algorithmic complexity associated with its processing [Larriva-Novo X., 2021]. Data preprocessing is a fundamental step in the data analysis and machine learning process, wherein raw data is readied for subsequent utilization. This stage includes various activities to clean, transform, and standardize data to improve subsequent analysis or model training quality and efficiency.

D. Principal component analysis. Principal Component Analysis (PCA) is a statistical technique that identifies the most impactful features for extracting maximal information from a dataset, facilitating the transformation of high-dimensional data into a lower-dimensional representation [Sanober S., 2022]. The features are chosen based on how much variance they produce in the result. The first principal component is the feature that contributes the most variance. The second principal component is the characteristic that accounts for the second-highest variance, and so on. It is significant to note that there is no correlation between the principal components.

The use of PCA for dimensionality reduction has primary benefit:

With fewer features, the algorithms' training times drop off dramatically. High-dimensional data analysis is only sometimes feasible. For example, let us say a dataset contains 100 features. To visualize the data, 100(100-1)2 = 4950 scatter plots would be needed. It is impossible to analyze data in this manner.

E. Machine Learning Algorithms for Traffic Classification. The modeling process involves training a machine learning algorithm to predict labels from features, refining it to meet business requirements, and validating it using holdout data. The outcome of the modeling process is a trained model suitable for inference and predicting outcomes for new data points. A machine learning model is a trained file designed to identify and recognize specific patterns.

Training a model encompasses providing an algorithm to reason over and learn from a dataset. Subsequently, the trained model can be utilized to reason over unfamiliar data and make predictions. For instance, consider the scenario where there is a desire to develop an application capable of discerning a user's emotions based on facial expressions. The model can be trained by supplying images of faces, each associated with a particular emotion. Once trained, the model can be employed in an application to recognize the emotions of any user.

Logistic Regression

This statistical model, the logit model, is frequently employed for classification and predictive analytics. Logistic regression serves as a method for forecasting the outcomes of binary trials, where events have two possible results [Das A., 2021]. The model calculates the probability of a particular event, such as voting or not voting, by leveraging a dataset of independent variables. As the predicted outcome is a probability, the dependent variable is constrained from 0 to 1. Logistic regression involves applying a logit transformation to the odds, representing the probability of success divided by the probability of failure. This transformation is called the log odds or the natural logarithm of odds. The logistic function, represented by the following formulas (1), encapsulates this transformation.

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}},$$

$$g(z) = \frac{1}{1+e^{-z}} \tag{1}$$

In the logistic regression equation, 'h' represents the dependent or response variable, and 'x' denotes the independent variable. Within this model, the beta parameter, known as the coefficient, is typically estimated using maximum likelihood estimation (MLE). This method involves testing various beta values through multiple iterations to optimize the fit of log odds. These iterations generate the log-likelihood function, and logistic regression aims to maximize this function to determine the most accurate parameter estimate.

Once the optimal coefficient (or coefficients, in the case of multiple independent variables) is identified, the model can calculate conditional probabilities for each observation. These probabilities are then logged and aggregated to generate a predicted probability. In binary classification, a probability less than 0.5 predicts 0, whereas a probability greater than 0 predicts 1. Following the computation of the model, it is advisable to assess its predictive performance regarding the dependent variable, a process known as goodness of fit.

Binary logistic regression:

In this methodology, the dependent variable is binary, indicating it possesses solely two potential outcomes (for instance, 0 or 1). Prominent instances of its application involve forecasting whether an email falls into the category of spam or determining the malignancy status of a tumor. This specific approach is widely utilized within logistic regression and, more broadly, stands out as one of the prevailing classifiers for binary classification tasks.

Multinomial logistic regression:

In this variant of the logistic regression model, the dependent variable exhibits three or more potential outcomes, yet these values lack a predetermined order. Consider, for instance, the scenario where movie studios aim to anticipate the probable film genre a moviegoer might choose, enhancing their marketing effectiveness. Employing a multinomial logistic regression model enables the studio to assess the impact of factors like age, gender, and relationship status on individuals' film preferences. Consequently, the studio can tailor advertising campaigns for specific movies to target groups that are more likely to show interest.

K-nearest neighbors

The k-nearest neighbors algorithm is a non-parametric, supervised learning classifier, alternatively recognized as KNN or k-NN. It relies on proximity to formulate classifications or predictions of categorizing a singular data point. Despite its applicability to regression or classification problems, it is more commonly employed as a classification algorithm, grounded in the notion that comparable data points tend to cluster nearby.

In order to ascertain the proximity of given query points to a designated query point, it becomes imperative to calculate the distances between the query mentioned above point and the remaining data points [Boateng E. Y., 2020]. Distance metrics are pivotal in delineating decision boundaries and segregating query points into distinct regions. The visualization of decision boundaries is frequently realized through the representation of Voronoi diagrams.

Euclidean distance

The KNN algorithm uses Euclidean distance metrics to locate the nearest neighbor [Boateng E. Y., 2020]. The Euclidean distance is calculated using equation (2).

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

Manhattan distance

The Manhattan distance is calculated using equation (3).

$$d=\sum_{i=1}^{n}\left(x_i-y_i\right) \qquad (3)$$

Naive Bayes

Naive Bayes classifiers encompass a set of classification algorithms grounded in Bayes' Theorem. Rather than a singular algorithm, it constitutes a family of algorithms unified by a common principle. The fundamental assumption underlying Naive Bayes is the independence of every pair of classified features, a condition only sometimes met in real-world scenarios. Despite the inherent inaccuracy of the independence assumption, it frequently yields effective results in practical applications. Consequently, Naive Bayes is a foundational probabilistic technique that assesses the probability of classifying or predicting the cyber-attack class within a provided dataset [Alqahtani H., 2020].

Bayes’ Theorem

Bayes' Theorem is a mathematical principle employed to ascertain the probability of an event transpiring, considering the probability associated with the occurrence of another previously occurring event. Bayes' Theorem is expressed as the following equation (4):

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \quad (4)$$

Bayes' Theorem computes the probability of an event (A) occurring based on the probability of another event (B) where P(B) ≠ 0. In this context, A and B are events, with P(B) representing the probability of event B, which serves as evidence. The objective is to determine the probability of event A, given that event B is accurate. P(A) denotes the prior probability of A, r presenting the likelihood of an event before evidence is observed. P(A|B) is A's posterior probability, indicating an event's probability after considering evidence (event B).

Support Vector Machines

The Support Vector Machine (SVM) is a relatively simple supervised machine learning algorithm utilized for classification and, to some extent, regression tasks. Although it is primarily applied in classification, there are instances where it also proves beneficial for regression. SVM identifies a hyperplane that acts as a boundary between different data types, typically appearing as a line in two-dimensional space. In SVM, each data point in the dataset is represented in an N-dimensional space, where N corresponds to the number of features or tributes in the data. The goal is to ascertain the optimal hyperplane that separates the data [Cervantes J., 2020].

It is crucial to emphasize that SVM is inherently geared towards binary classification, indicating its capability to make decisions between two classes. Nevertheless, when confronted with multi-class problems, diverse strategies can be applied. In the SVM framework for multi-class see areas, the strategy entails establishing a binary classifier for each class. The outcomes of each classifier yield one of two possibilities:
- The data point belongs to that class OR;
- The data point does not belong to that class.

To illustrate, in a class of fruits, we can establish a binary classifier for each specific fruit, enabling multi-class classification. For instance, a binary classifier dedicated to the 'mango' class is designed to predict whether a given instance is a mango. The cla sifier with the highest score among all binary classifiers is then selected as the output of the SVM for the multi-class classification. Regarding SVM for complex (non-linearly separable) data, SVM naturally performs well without any adjustments for linearly separable data. Linearly separable data refers to any dataset that can be graphically represented and divided into classes using a straight line.

Kernelized SVM is employed for handling non-linearly separable data. When the data exhibits nonlinear separability in one dimension, a transformation is applied to elevate it to two dimensions, rendering it linearly separable in the transformed space. This transformation involves mapping each one-dimensional data point to a corresponding two-dimensional ordered pair. Consequently, for any non-linearly separable data in any dimension, it is feasible to map it to a higher dimension, thus achieving near separability. This transformation is both compelling and universally applicable.

In the realm of Kernelized SVM, a kernel serves as a measure of similarity between data points. The kernel function within a kernelized SVM offers insights into the likeness of two data points in the original feature space, considering their relationship in the newly transformed feature space [Guo Y., 2021]. Various kernel functions are available, but two have gained prominence:

- Radial Basis Function Kernel (RBF): This kernel computes the similarity between two locations in the converted feature space using an exponentially fading function of the distance between the vectors in the original input space. The RBF kernel is commonly used as the default kernel in SVM.
- Polynomial Kernel: The Polynomial kernel introduces an additional parameter, 'degree,' which governs the model's complexity and the computational cost of the transformation.

Decision Tree

The Decision Tree is a widely used and effective tool for classifying and predicting data [Charbuty

B., 2021]. It is structured like a flowchart, with internal nodes representing attribute tests, branches indicating test outcomes, and leaf nodes holding class labels.

Building a decision tree involves splitting the source set into subsets based on the test of attribute values. This splitting occurs recursively on each derived subset, a method known as recursive partitioning. The recursion continues until the subset at a node possesses the same target variable value or further splitting ceases to enhance predictive value.

Decision trees categorize instances by traveling them down from the root to a specified leaf node containing the instance's classification. The classification procedure begins at the root node, checking the attribute supplied by that node before progressing along the tree branch corresponding to the attribute's value. This step is repeated for the subtree rooted at the new node. In the provided figure, the decision tree classifies a particular morning based on its suitability for playing tennis and returns the classification associated with the specific leaf (in this case, Yes or No).

Random forest

The Random Forest is a supervised learning method, especially a classifier composed of decision trees that act as an ensemble in the context of ensemble learning [Breiman L., 2001]. The "forest " constructed by Random Forest is essentially an ensemble of decision trees, typically trained using the "bagging" method. The core concept behind bagging is that combining multiple learning models enhances overall performance. A noteworthy advantage of the random forest is its applicability to classification and regression problems, which are prevalent in contemporary machine learning systems. Additionally, it effectively mitigates overfitting issues often encountered in individual decision trees.

XGBOOST Regressor

XGBoost (Extreme Gradient Boosting) Regressor is a machine learning algorithm to predict threat level and regression tasks. It is an extension of the original XGBoost algorithm for classification problems. XGBoost Regressor is particularly powerful and effective for predicting continuous numerical values, making it well-suited for regression problems.

Artificial Neural Networks

Artificial Neural Networks (ANNs) constitute a category of machine learning methodologies that form the foundation of deep learning paradigms. ANNs function as models for information processing, drawing inspiration from the operational principles of biological nervous systems observed in the human brain [Abiodun O. I., 2018]. Consisting of layers of nodes, artificial neural networks comprise an input layer, one or more hidden layers, and an output layer. Each node, representing an artificial neuron, is interconnected with other nodes, allocated distinctive weights, and governed by a predetermined threshold.

Activation of a node occurs if the output exceeds the specified threshold value, leading to the transmission of data to the subsequent layer in the network. Conversely, no data is forwarded to the next network layer if the output falls below the threshold.

Neural networks employ training datasets to acquire knowledge and enhance precision through iterative refinement processes. Once these learning algorithms are refined for accuracy, they become potent tools in computer science and artificial intelligence, enabling swift classification and clustering of data. Tasks such as speech or image recognition can be accomplished within minutes, a notable improvement compared to the hours required for manual identification by human experts. Google's search algorithm is one of the most renowned examples of a neural network.

Results and discussion

The section demonstrates the effectiveness of machine learning classification techniques for detecting intrusions on both Train and Test data. Various popular classification techniques are analyzed, including Logistic Regression, K-nearest neighbors, Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Artificial Neural Networks, with a comparison of the effectiveness of these popular classification techniques. Accuracy, precision, and recall are calculated as defined above to assess the potentiality of models.

1. Accuracy: This is a general indicator of how often the model correctly classifies data. Values close to 1, or 100%, indicate that the model accurately determines whether network traffic is normal or abnormal (an attack).

2. Precision: This indicator reflects how accurately the model predicts positive classes. For example, when the model predicts an attack, the accuracy shows how likely it is that it is indeed an attack.

3. Recall: This measures the model's ability to detect all real-world attacks. A high completeness score indicates that the model can detect and not miss attacks.

The resulting graph (figure 1) is a histogram that visually compares each machine-learning model's training and testing accuracy. This comparison is crucial for evaluating the performance and generalization ability of models. A model that works well with training and testing data is generally considered to have a good balance between bias and variance, indicating effective learning and generalization.

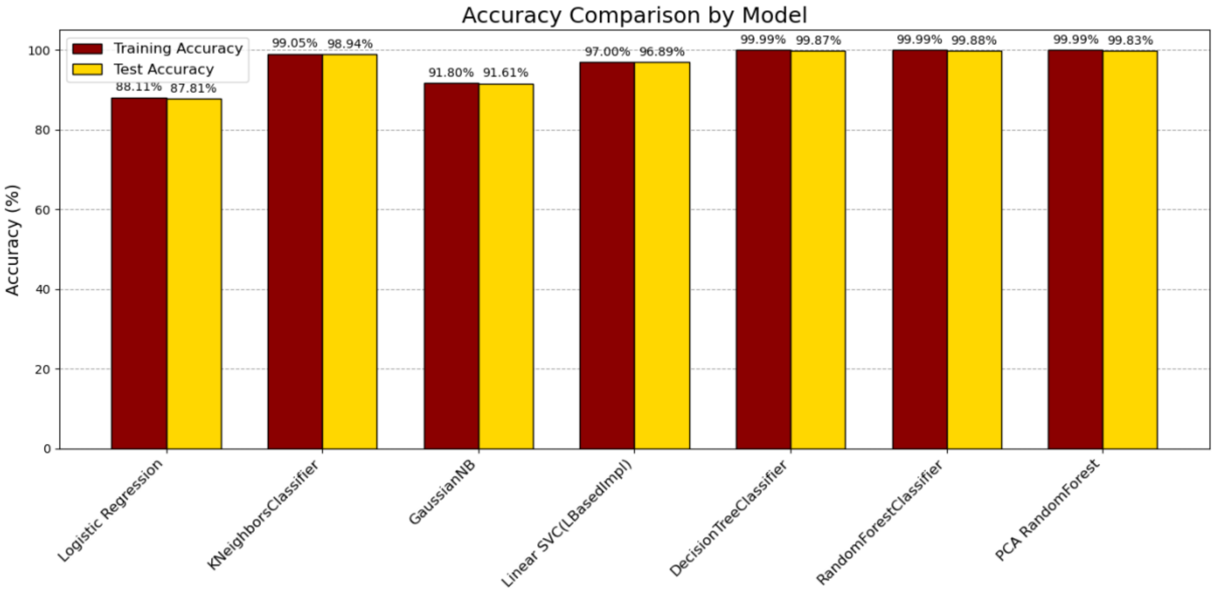

Figure 1. Accuracy Comparison by Model

The resulting histogram (figure 2) visually represents how accurately each model correctly classifies the res lts. Accuracy is an essential classification indicator, especially in cases where false positive results are expensive. The location of each model next to each other makes it easy to compare the difference in accuracy of each model during the training and testing stages. This can help identify models that are potentially overfitting (high learning accuracy but significantly lower testing accuracy) or those that maintain a consistent level of accuracy in both training and testing datasets.

The resulting histogram (figure 3) visually compares each machine-learning model's recall in training and testing scenarios. This is the most critical indicator in classification tasks, especially when skipping a positive instance (false negative) is expensive. This diagram helps assess models' ability to identify positive examples correctly. The arrangement of the strips of each model next to each other makes it easy to compare their performance during training and testing, which can help identify models that may be overfitted or insufficiently adapted.

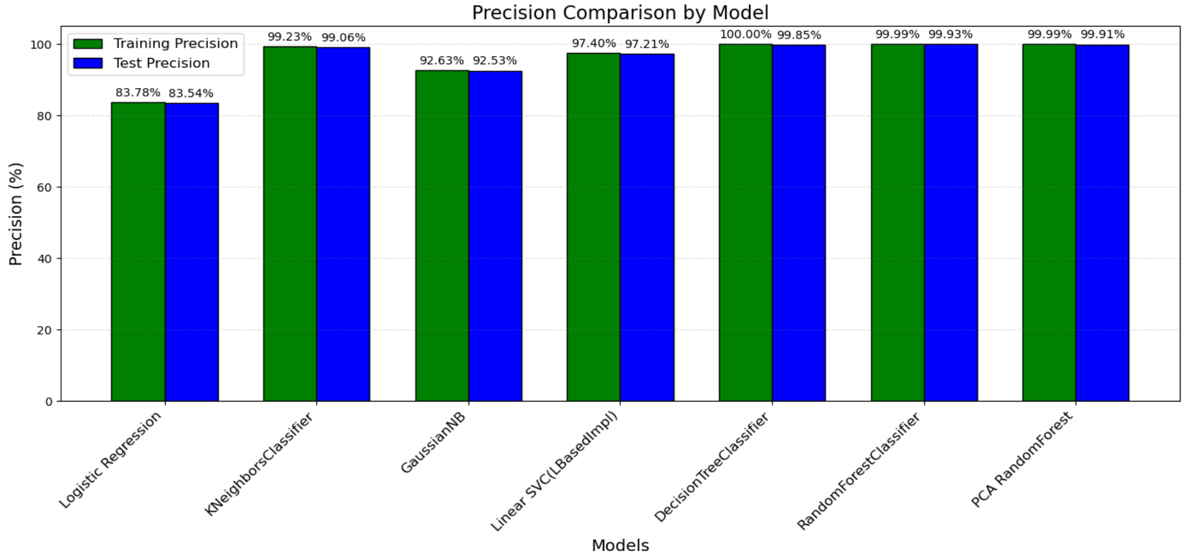

Figure 2. Precision Comparison by Model

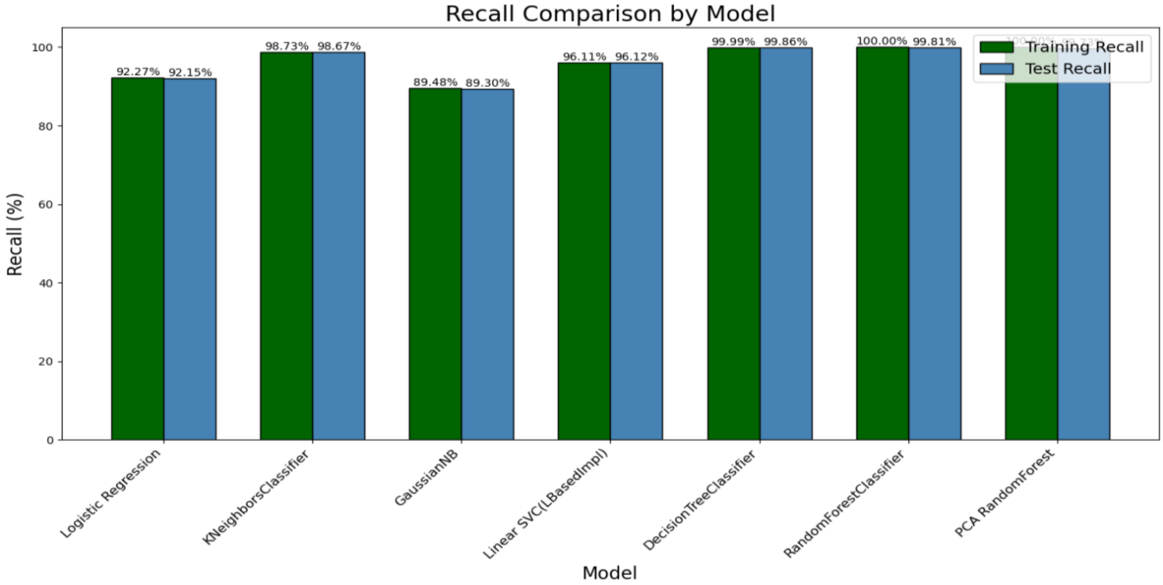

Figure 3. Recall Comparison by Model

The provided table (table 2) illustrates the model's effectiveness based on both Train and Test data across Accuracy, Precision, and Recall metrics.

Table 2

Summary of the performance

	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall
Logistic Regression	0.881104	0.878111	0.837803	0.835406	0.922727	0.921501
K neighbors classifier	0.990524	0.989363	0.992251	0.990564	0.987313	0.986705
GaussianNB	0.918027	0.916055	0.926266	0.925325	0.894791	0.892963
Linear SVC (L Based Impl)	0.963117	0.961103	0.952792	0.950270	0.968604	0.967652
Decision Tree Class	0.999940	0.998770	1.000000	0.998561	0.999872	0.998814
Random Forest Classifier	0.999940	0.998730	0.999872	0.999322	1.000000	0.997968
PCA Random Forest	0.999940	0.998254	0.999915	0.999151	0.999957	0.997121
Optimized Random Forest	0.992607	0.992181	0.999350	0.998883	0.984729	0.984419

Based on the study's findings, the following may be concluded:

Random Forest and PCA Random Forest have proven very effective on all three metrics: training, test data, and test data. This means that they classify traffic very accurately and reliably.

The Decision Tree model also showed promising results, indicating the model's reasonable ability to distinguish between regular traffic and attacks.

Logistic regression, KNN classifier, Gaussian, and Linear SVC performed well, but their performance is slightly lower than that of ensemble models (such as random forest).

Conclusion

As a result of the survey, an Intrusion Detection System was developed for detecting and preventing cyber threats based on network traffic using data obtained from a Security Information and Event Management (SIEM) system.

Various machine learning algorithms were employed for accurate traffic classification and the identification of potential attacks:

1. Support Vector Machine (SVM): Chosen for effectively separating data into classes, especially in complex multidimensional spaces. SVM is well-suited for classification and aids in threat detection.

2. K-Nearest Neighbors (KNN): Used for classification based on proximity to neighboring data points. This allows the detection of anomalies in network traffic.

3. Decision Trees and Random Forest: Help identify critical factors and patterns in data leading to threats. Random Forest enhances stability and prediction accuracy.

4. Gaussian Naive Bay s: Applied for classification using probability principles, practical n uncertainty, and data variability cases.

5. XGBoost: Particularly useful for processing large volumes of data with good performance and accuracy.

A neural network is applied to improve the accuracy and reliability in detecting complex threat patterns. It is trained on data to identify intricate patterns and dependencies that may not be apparent with traditional machine learning methods. Neural networks excel in handling large and complex datasets, where they can uncover nonlinear relationships.

The work enhances cybersecurity by providing a fast and accurate means of analyzing and responding to threats. In conclusion, the neural network and traditional mac ine learning models work together, offering a multi-layered approach to data analysis and classification in cybersecurity.

References

Abiodun O.I., Jantan A., Omolara A.E., Dada K.V., Mohamed N.A., Arshad H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11) [Electronic resource]: URL: https://europepmc.org/article/pmc/626043 (date of access: 11.01.2024).

Alqahtani H., Sarker I.H., Kalim A., Minhaz Hossain S.M., Ikhlaq S., Hossain S. (2020) Cyber intrusion detection using machine learning classification techniques. In *Computing Science, Communication and Security: First International Conference, COMS2,* 2020, Gujarat, India, March 26-27, 121-131. Springer Singapore [Electronic resource]: URL: https://doi.org/10.1007/978-981-15-6648-6_10 (date of access: 11.01.2024).

Boateng E.Y., Otoo J., Abaye D.A. (2020) Basic tenets of classification algorithms K-nearest-neighbor, support vector machine, random forest and neural network: a review. *Journal of Data Analysis and Information Processing*, 8(4), 341-357 [Electronic resource]: URL: https://www.scirp.org/journal/paperinformation?paperid=104256 (date of access: 11.01.2024).

Breiman L. (2001) Random forests. *Machine learning*, 45, 5-32 [Electronic resource]: URL: https://link.springer.com/article/10.1023/a:1010933404324 (date of access: 11.01.2024).

Buczak A.L., Guven E. (2015) A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications surveys & tutorials*, 18(2), 1153-1176 [Electronic resource]: URL: https://ieeexplore.ieee.org/abstract/document/7307098 (date of access: 11.01.2024).

Cervantes J., Garcia-Lamont F., Rodríguez-Mazahua L., Lopez A. (2020) A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408, 189-215 [Electronic resource]: URL: https://www.sciencedirect.com/science/article/abs/pii/S0925231220307153 (date of access: 11.01.2024).

Charbuty B., Abdulazeez A. (2021) Classification based on decision tree algorithm for machine learning. *Journal of Applied Science and Technology Trends*, 2(01), 20-28 [Electronic resource]: URL: https://www.jastt.org/index.php/jasttpath/article/view/65 (date of access: 11.01.2024).

Das A. (2021) Logistic regression. *Encyclopedia of Quality of Life and Well-Being Research,* 1-2. Cham: Springer International Publishing [Electronic resource]: URL: https://link.springer.com/referenceworkentry/10.1007/978-3-319-69909-7_1689-2 (date of access: 11.01.2024).

Dasgupta D., Akhtar Z., Sen S. (2022) Machine learning in cybersecurity: a comprehensive survey. *The Journal of Defense Modeling and Simulation*, 19(1), 57-106 [Electronic resource]: URL: https://journals.sagepub.com/doi/full/10.1177/1548512920951275 (date of access: 11.01.2024).

Guo Y., Zhang Z., Tang F. (2021) Feature selection with kernelized multi-class support vector machine. *Pattern Recognition*, 117, 107988 [Electronic resource]: URL: https://www.sciencedirect.com/science/article/abs/pii/S0031320321001758 (date of access: 11.01.2024).

Handa A., Sharma A., Shukla S.K. (2019) Machine learning in cybersecurity: A review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(4), 1306 [Electronic resource]: URL: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1306 (date of access: 11.01.2024).

Larriva-Novo X., Villagrá V.A., Vega-Barbas M., Rivera D., Sanz Rodrigo M. (2021) An IoT-focused intrusion detection system approach based on preprocessing characterization for cybersecurity datasets. *Sensors*, 21(2), 656 [Electronic resource]: URL: https://www.mdpi.com/1424-8220/21/2/656 (date of access: 11.01.2024).

Natarajan J. (2020) Cyber secure man-in-the-middle attack intrusion detection using machine learning algorithms. *AI and Big Data's Potential for Disruptive Innovation*, 291-316. IGI global [Electronic resource]: URL: https://www.igi-global.com/chapter/cyber-secure-man-in-the-middle-attack-intrusion-detection-using-machine-learning-algorithms/236343 (date of access: 11.01.2024).

Sanober S., Aldawsari M., Karimovna A.D., Ofori I. (2022) Blockchain Integrated with Principal Component Analysis: A Solution to Smart Security against Cyber-Attacks. *Security and Communication Networks*, 2022 [Electronic resource]: URL: https://www.hindawi.com/journals/scn/2022/8649060/ (date of access: 11.01.2024).

Saranya T., Sridevi S., Deisy C., Chung T.D., Khan M.A. (2020) Performance analysis of machine learning algorithms in intrusion detection system: A review. *Procedia Computer Science*, 171, 1251-1260 [Electronic resource]: URL: https://www.sciencedirect.com/science/article/pii/S1877050920311121 (date of access: 11.01.2024).

Sarker I.H., Abushark Y.B., Alsolami F., Khan A.I. (2020) Intrudtree: a machine learning based cyber security intrusion detection model. *Symmetry*, 12(5), 754 [Electronic resource]: URL: https://www.mdpi.com/708786 (date of access: 11.01.2024).

Yavanoglu O., Aydos M. (2017) A review on cyber security datasets for machine learning algorithms.

In *2017 IEEE international conference on big data (big data),* 2186-2193. IEEE [Electronic resource]: URL: https://ieeexplore.ieee.org/abstract/document/8258167 (date of access: 11.01.2024).

Интегрированная система безопасности компьютерных сетей: обнаружение вторжений на основе SVM и прогнозирование угроз с использованием алгоритмов машинного обучения

*[1]*М.С.Утарбаева,* [1]*М.А.Муқанова*

[1]*Назарбаев Интеллектуальная школа химико-биологического направления (Алматы, Казахстан)*

Аннотация

В связи с быстрым ростом использования компьютерных сетей и значительным расширением связанных с ними приложений вопросы кибербезопасности становятся все более актуальными. В этом документе будет представлен обзор решений растущих проблем сетевой безопасности, за которым последует разработка инструмента для обнаружения и предотвращения киберугроз путем анализа данных сетевого трафика из системы управления информацией о безопасности и событиями (SIEM). Используя различные алгоритмы машинного обучения, включая SVM, KNN, Decision Tree, Random Forest, Gaussian Naive Bayes, XGBoost и нейронные сети, исследование обеспечивает точную классификацию трафика и идентифицирует потенциальные угрозы. Нейронная сеть повышает точность обнаружения сложных моделей угроз. Исследование уникально сочетает в себе целевое применение в области кибербезопасности, всестороннее сравнение моделей и практическую реализацию для получения точных данных. Результаты, продемонстрированные с использованием гистограмм и таблиц, демонстрируют эффективность Random Forest и PCA Random Forest, подчеркивая их точную классификацию трафика. В заключение, проведена оценка эффективности различных экспериментов на наборе данных по кибербезопасности, включающего несколько категорий кибератак, а также оценка эффективности метрик производительности, таких как Accuracy, Precision и Recall. Многоуровневый подход, основанный на последних тенденциях машинного обучения в области кибербезопасности, обеспечивает быстрый и точный анализ угроз и реагирование на них, тем самым повышая их уровень.

Ключевые слова: кибербезопасность, машинное обучение, алгоритмы машинного обучения, обнаружение вторжений, компьютерная безопасность, прогнозирование кибератак.

Компьютерлік желілердің интеграцияланған қауіпсіздік жүйесі: SVM негізіндегі интрузияларды анықтау және машиналық оқыту алгоритмдерін қолдану арқылы қауіптерді болжау

*[1]*М.С.Утарбаева,* [1]*М.А.Муқанова*

[1]*Химия-биологиялық бағыттағы Назарбаев Зияткерлік мектебі (Алматы, Қазақстан)*

Аңдатпа

Компьютерлік желілерді пайдаланудың тез өсуіне және онымен байланысты қосымшалардың едәуір кеңеюіне байланысты киберқауіпсіздік мәселелері өзекті бола түсуде. Бұл құжат өсіп келе жатқан желілік қауіпсіздік мәселелерінің шешімдеріне шолу жасайды, содан кейін қауіпсіздік және оқиғаларды басқару жүйесінен (SIEM) желілік трафик деректерін талдау арқылы киберқауіптерді анықтау және алдын алу құралын әзірлейді. Машиналық оқытудың әртүрлі алгоритмдерін, соның ішінде Random Forest, Gaussian Naive Bayes, XGBoost және нейрондық желілерді пайдалана отырып, зерттеу трафиктің нақты жіктелуін қамтамасыз етеді және ықтимал қауіптерді анықтайды. Нейрондық желі күрделі қауіп үлгілерін анықтау дәлдігін арттырады. Зерттеу киберқауіпсіздікті мақсатты қолдануды, модельдерді жан-жақты салыстыруды және нақты деректерді алу үшін практикалық іске асыруды бірегей түрде біріктіреді. Гистограммалар қолдану арқылы көрсетілген нәтижелер Random Forest пен PCA Random Forest тиімділігін көрсетеді, олардың трафиктің нақты жіктелуіне баса назар аударады. Соңында, бірнеше кибершабуыл санаттары бар киберқауіпсіздік деректер жиынтығымен әртүрлі эксперименттердің тиімділігі және өнімділік, дәлдік, еске түсіру және дәлдік көрсеткіштерінің тиімділігін бағалау. Киберқауіпсіздік саласындағы машиналық оқытудың соңғы тенденцияларына негізделген көп деңгейлі тәсіл қауіптерді жылдам және дәл талдауды және оларға жауап беруді қамтамасыз етеді, осылайша олардың деңгейін арттырады.

Түйін сөздер: киберқауіпсіздік, машиналық оқыту, машиналық оқыту алгоритмдері, интрузияны анықтау, компьютерлік қауіпсіздік, кибершабуылдарды болжау.

Поступила в редакцию: 17.01.2024
Одобрена: 11.03.2024
Первая публикация на сайте: 05.01.2025

МРНТИ: 20.01.45 https://doi.org/10.65247/3105-3432-2025-1.04

БАСТАУЫШ СЫНЫПҚА SCRATCH БАҒДАРЛАМАСЫН ҮЙРЕТУ

*[1]Н.А.САНСЫЗБАЙ, [1]И.О.САЙФУРОВА
Әлкей Марғұлан атындағы Павлодар педагогикалық университеті
(Павлодар, Қазақстан)
*nuraisansyzbai26@gmail.com, saifurova_indira@teachers.ppu.edu.kz

Аңдатпа
Scratch бағдарламасының негізгі мақсаты алгоритімдеуді қызықтырып беру. Қазіргі таңда цифрлық сауаттылық өте маңызды және күн сайын дамып келеді. Бұл өз тарапынан алгоритмдеуді, кодтауды, жалпы базалық тұрғыда қазіргі заманға сай цифрлық сауаттылықты ерте буынан үйретіп беруге мүмкіндіктер береді. Бағдарлманы игеру арқылы, оқушыға алда одан да қиын бағдарламаларды игеруге жақсы серпін береді. Бұл мақалада Scratch бағдарламасының дағдылары, артықшылықтары көрсетілген, бағдарламаның басты ерекшеліге оның жарқын интерфейсі мен қолжетімді программалау тілі болғандықтан, онымен қалай жұмыс жасау жайлы айтылған. Бұл бағдарламаны не себептен бастауыш сыныптан оқыту және үйретілуі жайлы айтылып кеткен. Бағдарламаның оқушыларға беретін дағдылары және олардың игере аладын щеберліктері көрсетілген. Оқушылардың бұл бағдарламаға қызығушылық таныту үшін әдістер мен мысалдар көрсетілген.

Түйін сөздер: интерактивті, визуализация, цифрлық сауаттылық, анимация; алгоритм, Scratch бағдарламасы.

Кіріспе

Scratch - бұл бастауыш мектеп оқушыларына жиі таныс болатын ыңғайлы бағдарлама. Ол мазмұнды оқу тәжірибесін қамтамасыз етеді, өйткені ол тәжірибелік және зерттеушілік оқытуға шақыратын және жаңа ойындарды жасап, ойнауға еркіндік беретін ойын ортасын қамтамасыз етеді.

Scratch көмегімен оқыту процестері логикалық ойлауды дамытуға, үлкен мәселелерді кішігірім мәселелерге бөлу арқылы шешуге, жалпы мәселелердің жалпы шешімдерін анықтауға және пайдалануға және ынтымақтастыққа мүмкіндік береді.

Жаңа мемлекеттік жалпыға міндетті стандарттардың енгізілуімен жаңа білім беру жүйесінің маңызды құрамдас бөлігі ақпараттық-білім беру ортасы болып табылады, оның негізінде әртүрлі ақпараттық ресурстарға еркін қол жеткізуді қамтамасыз ететін заманауи ақпараттық технологиялар жатыр. Қазіргі уақытта ақпараттық-білім беру ортасын құрмай жұмыс жасау, мүмкін емес ақпараттық сауаттылықты қалыптастыру саласында ғылыми зерттеулер жүргізу ерекше өзекті болып табылады, олардың орындалуы педагогикалық мақсаттарға қол жеткізуге кепілдік беруге әкеледі. Құрылған ақпараттық-білім беру ортасы бастауыш сыныптарда "ақпараттық сауаттылық" пәнін оқуда, электронды ресурстар – факультативтік сабақтарды өткізуде, әдістемелік әзірлемелер – мектепте бағдарламалау тілін оқытуда, балаларды дамыту үшін қолданылатын болады. ақпараттық-білім беру ортасы білім алушының жеке басын дамыту процесінде оқуға деген ынтасын арттырады. Оқушыларға ақпаратты белсенді іздеуге және жұмысты өз бетінше орындауға көмектеседі, жеке тұлғаны дамытудың әртүрлі кезеңдерінде оқуға деген ыньасынн арттырды [Сабырханова Л.Ш. және т.б., 2023].