Development of intelligent intrusion detection systems based on intelligent data analysis methods*

Serhii Toliupa[1,†], Serhii Buchyk[1,*,†], Yurii Shcheblanin[1,†], Anastasiia Shabanova[1,†], and Andrii Kulko[1,†]

[1] *Taras Shevchenko National University of Kyiv, 60 Volodymyrska str., 01033 Kyiv, Ukraine*

Abstract

The dynamic expansion of cyber influence and the emergence of new types of threats in the digital space necessitate the implementation of proactive strategies for the development of intelligent systems. The creation of such methods is a fundamental prerequisite for high-quality data monitoring and strengthening cybersecurity. A priority area of scientific research is the development of innovative algorithms capable of effectively processing Big Data arrays. The research is aimed at improving the precision of analysis and ensuring the promptness of obtaining critically important information, which is necessary to meet the needs of modern industries for reliable information processing tools. Rapid progress in the IT sphere has led to a sharp increase in the number of attacks on the integrity of information structures. This has highlighted the role of intrusion detection systems (IDS) as a key element of protection. Intelligent data analysis technologies (Data Mining) open up significant prospects for their implementation in the field of information security. This article explores the potential of data mining methods within the framework of intrusion detection systems. The authors systematize IDS according to key characteristics and analyze the mathematical basis of the corresponding approaches. The most popular intelligent analysis algorithms used to neutralize cyber risks are reviewed. Particular attention was paid to a detailed study of three approaches: the support vector machine (SVM) method, cluster analysis, and the principal component analysis (PCA) method. Based on their combination, a hybrid attack detection model was synthesized. The effectiveness of the proposed solutions was evaluated by analyzing the true positive rate (TPR) and false positive rate (FPR). The created hybrid algorithm is based on the concept of layered defense: it integrates the accuracy of SVM in identifying typical threats with the high sensitivity of clustering and PCA to anomalies, which allows achieving an optimal balance between the sensitivity and specificity of the system in real traffic conditions.

Keywords

cybersecurity, network attack, IDS, cluster analysis, SVM, data mining, databases, recognition metrics, PCA, model performance

1. Introduction

Today, critical information networks face a wide range of cyberattacks, requiring the development of a large number of diverse protection methods and tools. In particular, large-scale and coordinated cyberattacks serve as a catalyst for the creation of specialized technical solutions and powerful countermeasures. To identify network intrusions, modern models, software, and comprehensive technical solutions integrated into intrusion detection and prevention systems are actively used. These systems are capable of maintaining high efficiency even in the face of new or modified cyber threats. Detecting them is based on known signatures or behavioral patterns. However, practice shows their limitations: when new anomalies or attacks with unknown or

* *CPITS 2026: Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2026, Kyiv, Ukraine*
* Corresponding author.
† These authors contributed equally.
tolupa@i.ua (S. Toliupa); buchyk@knu.ua (S. Buchyk); shcheblanin.yurii@knu.ua (Y. Shcheblanin), shabanovaa@knu.ua (A. Shabanova); kulko452@gmail.com (A. Kulko)
0000-0002-1919-9174 (S. Toliupa); 0000-0003-0892-3494 (S. Buchyk); 0000-0002-3231-6750 (Y. Shcheblanin); 0009-0008-4962-569X (A. Shabanova); 0009-0006-1185-0774 (A. Kulko)

unclear characteristics arise (for example, zero-day attacks), existing tools are not always effective. The solution to this problem is to use intelligent data analysis and artificial intelligence methods.

The cybersecurity situation in Ukraine remains critical: in 2025, over a million cyber threats were recorded. Of particular concern is the use of information technology in cyberspace for military-political and forceful confrontation, terrorist activities, and hacker attacks. Experts predict radical changes in the structure and methods of cyber operations in the coming year. Attacks are expected to become more covert, allowing them to more effectively bypass existing detection systems and firewalls.

In view of this, the scientific and practical task today is to develop comprehensive methods for detecting cyberattacks using intelligent intrusion prevention systems (IPS) in the information systems of government agencies. Successfully solving this task will help eliminate the aforementioned contradiction between the growing complexity of attacks and the limited effectiveness of current defense mechanisms.

A. Well-known approaches to solving the problem.

In [1], a methodology is proposed for synthesizing models of intelligent control and security systems for critical infrastructure facilities that are fundamentally different in their properties. In [2], the problems of intrusion detection and the development of methodological foundations for the analysis and synthesis of functionally stable distributed information systems against cyber threats based on multi-agent technology, which will ensure the timely detection and blocking of cyber influences, are considered. In [3], the main issues of the theory and practice of data mining are considered: algorithms, models, classification tasks, cluster analysis, search, deep data analysis, complex networks theory, as well as information necessary for mathematical and computer modeling and analysis of complex systems and networks in the field of cybersecurity. In [4], based on control theory and a general mathematical approach, the evaluation of the creation and functioning of cyber security systems in modern conditions is considered, as well as methods of synthesis, without which it is impossible to build a developed and adequate scientific basis for cybersecurity.

In [5], the current state and prospects for the development of security mechanisms are considered: cybersecurity, information security, information security, and information technology. The list can be quite extensive, which indicates that this issue is one of the key ones in the protection of critical infrastructure.

Currently, a large number of methods and means have been developed to protect information objects from cyber attacks. The intensity and scale of cyber attacks stimulate the constant creation of specialized technical solutions and countermeasures. To identify network intrusions, modern methods [6-8], models [9], tools [10-11], software [12], and comprehensive technical solutions for intrusion detection and prevention systems (IDPS) [13-14] are used. These tools often prove effective even when new or modified types of cyber threats emerge.

However, practical experience shows their limitations: when unknown threats or anomalies caused by attacks with unclear characteristics arise, existing tools may lose their effectiveness. In addition, adapting them to new conditions requires a significant amount of time. Therefore, intrusion detection systems (IDS) require constant research and improvement to maintain continuous and effective operation.

Global experience shows that the most effective methodological approach to building innovative intelligent cyberattack monitoring systems is to create hierarchical multi-level structures. These structures are designed for early detection of cyberattacks at the initial stages of their implementation.

When developing and conducting experimental studies on the effectiveness of intrusion detection systems, one of the key tasks is to select the data sets on which testing will be performed.

Large development companies primarily rely on their own databases, which are specialized for specific tasks and areas of application.

Training databases and testing intrusion detection systems. Selecting a high-quality training database with attack samples is a non-trivial task. Widely used databases often contain outdated types of attacks. More modern databases, although relevant, have a specific structure that requires complex preprocessing of data. As a result, such databases are used only by a narrow circle of researchers, which complicates the comparison of the performance and quality indicators of different systems.

B. Choosing a training base

The most comprehensive and widely used training databases (datasets) with samples of cyberattacks, traditionally used for the development and evaluation of intrusion detection systems (IDS), are KDDCup 1999 and NSL-KDD. The KDDCup 1999 (KDD99) database has a huge amount of data (about 5 million records) and contains 4 main categories of attacks (DoS, Probe, R2L, U2R). The disadvantage of the database is an excessive number of duplicate records, which leads to a bias in the results in favor of frequent attacks (e.g., DoS) and an overestimation of model accuracy.

The NSL-KDD database is a refined and improved version of KDD99. It retains the same four categories of attacks. The advantage is that it removes redundant and duplicate records, which provides a fairer evaluation of algorithms and allows models to learn better from rare attacks (R2L and U2R). Thanks to this correction, it is more complete and balanced for academic research than the original KDD99.

The other databases, CICIDS2017 and CICDDoS2019, represent a new generation of datasets that are more relevant to modern networks and have fewer methodological problems than KDD99. To develop a comprehensive, modern IDS, it is recommended to use CICIDS2017 or UNSW-NB15 for training on modern attack vectors, and NSL-KDD for comparing results with classical studies. We will use the latter in our research.

Among the leaders in vulnerability detection, the following developers of relevant vulnerability databases can be noted: MITRE and its Common Vulnerabilities and Exposures (CVE) database [16]; the National Institute of Standards and Technology and the National Vulnerabilities Database (NVD) [17]; the United States Computer Emergency Readiness Team and its Vulnerability Notes Database (VND) [18]; IBM and its X-Force vulnerability database [19-20]; and others [21].

To build effective intrusion detection systems, it is proposed to apply intelligent data analysis methods that include the intelligent properties of artificial intelligence, one of which is neural networks. However, it should be noted that solving this type of problem using a single method is problematic, so other methods should be used alongside neural networks to increase the effectiveness of intrusion detection systems. Therefore, a combination of clustering and neural network methods is proposed [22]. The first method will group similar cyber attacks into a specific cluster, and neural networks can be used for recognition [13]. In works [14-16], neural network hierarchies are used to detect anomalies. Neural networks are trained using data that covers normal space and are capable of recognizing unknown attacks. Work [17] reports on a real-time solution for detecting known and unknown attacks using uncontrolled neural networks.

Intelligent data analysis methods (Data Mining) offer significant advantages in intrusion detection systems (IDS) because they can automatically process huge amounts of network and system data, revealing hidden patterns and anomalies that are not accessible to traditional signature-based methods.

The main advantages of these methods are:

- detection of unknown (zero-day) attacks. Traditional signature-based intrusion detection systems can only detect known attacks. Data mining methods such as clustering and anomaly detection (e.g., using PCA, autoencoders, or isolation forests) create a model of normal behavior. Any significant deviation from this model (even if it is a completely new attack) generates an alert;

- reduction of false positives (FP). The use of supervised learning (e.g., Random Forest, SVM classifiers) allows the model to be trained on large amounts of labeled data. These models can distinguish between genuine but harmless events and real threats. This significantly improves accuracy and reduces the number of false alarms that "clutter" the work of security administrators.

Automation and adaptation. Data mining methods allow SVWs to be adaptive. Models can automatically retrain on new data to: adapt to changes in normal network behavior; automatically update detection rules without human intervention; reduce dimensionality and resource efficiency.

Data mining techniques, in particular dimensionality reduction (such as PCA or feature selection), are used to identify the most important features of network traffic (e.g., number of packets, session duration, number of errors); removing redundant information and noise; ensuring faster data processing and saving computing resources, which is critical for analyzing high-speed network traffic in real time.

Based on an analysis of scientific publications on intrusion detection systems (IDS), particularly on standard datasets (NSL-KDD), Table 1 was created showing the indicators of correct recognition of cyberattacks TPR (True Positive Rate) – the proportion of correctly identified cyberattacks among all actual attacks and false positives FPR (False Positive Rate) – the proportion of normal traffic mistakenly classified as an attack.

Table 1
Indicators of correct recognition of cyberattacks TPR (True Positive Rate) and false positives FPR (False Positive Rate)

Method	TPR (%)	FPR (%)
k-means clustering	99.80	0.50
SVM + Genetic algorithms	98.80	1.20
SVM + Fuzzy logic	98.50	1.50
C4.5 (Decision tree)	99.50	1.00
Multilayer perceptron (MLP)	98.00	1.00
Support vector machine (SVM)	97.80	8.60
k-nearest neighbors (k-NN)	95.00	5.00
Hidden Markov model (HMM)	92.00	8.00
Genetic algorithms (GA)	90.00	10.00
Neural networks + MGK	99.00	0.80
C4.5 + MGK	99.20	0.70
C4.5 + Hybrid neural networks	99.40	0.60
ɣ–algorithm	93.00	7.00
Y–means clustering	94.00	6.00

It can be concluded that the best performance (high TPR, low FPR) is usually achieved by hybrid and optimized methods, as well as decision trees (C4.5) and k-means (when used for anomaly detection). For example, k-means clustering and hybrids based on C4.5 or neural networks demonstrate TPR≈99.0% and FPR<1.0%. Hybrid models, such as SVM + genetic algorithms and SVM + fuzzy logic, often outperform their basic counterparts (pure SVM) because optimization algorithms (such as GA and Fuzzy Logic) help to better tune parameters or select features, significantly reducing FPR. Basic SVM in some unoptimized implementations can have a fairly high false positive rate (FPR≈8.6%), although its TPR remains high.

Thus, data mining is a key tool in modern cyber threat response systems, as it allows detecting hidden patterns and anomalies in large volumes of network traffic, system logs, and other data that indicate cyber attacks.

In our research, we will consider three methods of intelligent data analysis: support vector machines, clustering, and principal components, and model a hybrid method based on them. We will evaluate the effectiveness of these methods using TPR (True Positive Rate) indicators (the percentage of successfully detected attacks among all actual attacks) and FPR (False Positive Rate), or false positives: the percentage of legitimate traffic mistakenly classified as an attack (false alarm).

The Support Vector Machine (SVM) method is one of the most effective supervised machine learning algorithms, widely used in intrusion detection systems (IDS). Its mathematical apparatus is based on finding the optimal separating hyperplane with the maximum distance (gap) between data classes, which increases its ability to generalize.

In the context of intrusion detection systems, SVM solves the problem of binary classification: it assigns network events or traffic characteristics to one of two classes: "+1" (normal activity) or "–1" (attack/anomaly).

Let's consider the mathematical apparatus of SVM for binary classification. We are faced with the main goal of SVM: to find the optimal hyperplane that maximally separates the two classes of data.

A. Linearly separable case (Hard Margin SVM).

For a training data set

$$D=\{(x_i,y_i)\}_{i-1}^{N}, \tag{1}$$

where x_i is a feature vector (e.g., network packet characteristics) and $y_i \in \{-1,+1\}$ is a class label (normal or attack), the hyperplane is defined by the equation:

$$w \cdot x+b=0, \tag{2}$$

where w -is the normal vector to the hyperplane, b - is the bias of the hyperplane.

In order for the hyperplane to separate classes, the following condition must be satisfied for all points:

$$y_i(w \cdot x_i+b)\geq 1. \tag{3}$$

The task of finding the optimal hyperplane boils down to minimizing the norm of the vector w to maximize the margin $2/\|w\|$:

$$\min_{w,b}\frac{1}{2}\|w\|^2 \text{ subject to: } y_i(w \cdot x_i+b)\geq 1,\ i=1,\dots,N. \tag{4}$$

B. Non-linearly separable case (Soft Margin SVM).

Network data in cybersecurity is rarely linearly separable. To account for noise, outliers, or situations where data cannot be perfectly separated, Soft Margin SVM is used with the introduction of slack variables ξ_i, $\xi_i>1$:

$$y_i(w \cdot x_i+b)\geq 1-\xi_i. \tag{5}$$

The optimization problem is transformed into minimization:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \text{ under the conditions: } y_i(w \cdot x_i + b) \geq 1 - \xi_i,\ \xi_i \geq 1, i = 1, \ldots, N, \quad (6)$$

where C is a regularization parameter that controls the trade-off between maximizing the margin and minimizing classification errors (the sum of ξ_i).

C. Kernel Trick.

SVM uses a kernel trick for nonlinear data separation. This allows input feature vectors to be mapped to a higher-dimensional space where classes can be linearly separable:

$$K(x_i, y_i) = \Phi(x_i) \cdot \Phi(y_i), \quad (7)$$

where $K(x_i, y_i)$ is the kernel function.

The most common kernels used in IDS are: polynomial kernel and radial basis function (RBF), or Gaussian kernel (the most popular).

The kernel trick allows solving the optimization problem in high-dimensional space while avoiding explicit computations with these high-dimensional vectors, which is key for processing multidimensional network traffic data.

In IDS, the SVM method is most often used to classify network connections, system calls, or log entries as normal or abnormal/malicious.

Advantages of SVM in IDS: high classification accuracy. SVM has a high generalization ability due to the principle of structural risk minimization, which allows it to work well on unseen (new) data and reduce the false positive rate.

Efficiency in multidimensional space: Using the SVM kernel trick, effects-in processes network data with a large number of features (e.g., 41 features in the NSL-KDD dataset), transforming nonlinearly separable problems into linearly separable ones.

Support vector determination: the solution depends only on a small subset of training points called support vectors. This reduces computational complexity and ensures model stability.

Overfitting resistance: by controlling the gap between classes, SVM is less prone to overfitting compared to some other methods. Based on scientific research (using the NSL-KDD and UNSW-NB15 datasets), data on the effectiveness of the support vector method (SVM) for detecting various types of cyberattacks are presented: denial of service (DoS/DDoS) attacks, Probe (Probing), R2L (Remote to Local), U2R (User to Root).

Table 2 shows the effectiveness of SVM against different types of cyber attacks in terms of TPR and FPR. The figure visualizes the relationship between true positive rate (TPR) and false positive rate (FPR) of the SVM method for different categories of attacks. It can be seen that for attacks with a high network signature (DoS, Probe), TPR is high, while for rare, internal, or unknown attacks (R2L, U2R, Zero-Day), TPR drops significantly and FPR increases.

Table 2

SVM performance table against different types of cyberattacks

Attack type	TPR (%)	FPR (%)	Note
DoS (denial of service)	98.5	1.0	Attacks are massive and easily noticeable, SVM demonstrates high efficiency
DDoS (distributed DoS)	96.0	3.0	More difficult due to traffic distribution, but still high TPR
Probe (scanning)	95.0	2.5	Noticeable abnormal activity, SVM detects well
R2L (remote to local)	70.0	6.0	Difficult to detect. The attack resembles legitimate remote access, resulting in lower TPR

U2R (user to root)	65.0	7.5	The most difficult to detect among known types. Requires analysis of user actions on the host, rather than network traffic

The second method chosen is clustering. Clustering in intrusion detection systems (IDS) is used as a method of anomaly detection. The mathematical apparatus of clustering is aimed at grouping similar objects (network connections, log entries) into clusters. Any new data point that deviates significantly from existing "normal" clusters is identified as a potential intrusion or anomaly.

Let's look at the main clustering algorithms and their mathematical apparatus.

The following algorithms are most commonly used in IDS:

A. K-means.

K-means is the most popular distance-based partitioning clustering algorithm.

Problem statement: divide N data points $\{x_1, x_2, \dots x_N\}$ into K clusters $C = \{C_1, C_2, \dots, C_K\}$.

Loss function (objective function): the algorithm aims to minimize the sum of squared distances (Squared Euclidean Distance) from each point to the center (centroid) of its cluster:

$$J = \sum_{j=1}^{K} \sum_{x_i \in C_j} \left\| x_i - \mu_j \right\|^2, \tag{8}$$

where μ_j is cluster centroid C_j.

Update step: centroid μ_j calculated as the arithmetic mean of all points in the cluster C_j:

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i. \tag{9}$$

The application of this method in IDS consists in first determining K clusters corresponding to different types of normal behavior (e.g., web traffic, FTP traffic, DNS queries). A new event x_{new} is classified as an anomaly if its distance to the nearest centroid μ_{min} exceeds a certain threshold T.

B. DBScan (Density-Based Spatial Clustering of Applications with Noise).

DBScan is a density-based clustering algorithm that identifies clusters of arbitrary shape well and effectively separates "noise" (anomalies).

Mathematical apparatus. The algorithm uses two key parameters: *Eps* (radius) - the maximum distance between two points for one to be considered "reachable" from the other; *MinPts* - the minimum number of points required to form a dense set.

Points are classified as: Core Point; a point that has a minimum of *MinPts* of points (including itself) within the radius *Eps*.

Border Point: a point that is within *Eps* of a core point, but is not itself a core point.

Noise Point: A point that is neither a core nor a boundary.

In this context, noise points x_{noise} (those that do not belong to any cluster) are directly interpreted as anomalies (intrusions). This method is very powerful because it does not require prior knowledge of the number of clusters and is robust to outliers.

Clustering-based intrusion detection systems work on the principle of learning normality. Data preprocessing: network traffic, system logs, or user data are transformed into feature vectors (e.g., number of packets, connection duration, ports used). Training: A clustering algorithm (e.g., *K*-Means or DBScan) is applied to a large dataset that is considered normal activity. Result: models of normal behavior are formed (e.g., *K* centroids or dense cluster regions). Detection: new input data x_{new} is compared to the trained model: if x_{new} is close to the center (or inside a dense region) of a normal cluster $\Longrightarrow$ normal activity. If x_{new} is far from all centers (or classified as a noise point) $\Longrightarrow$ anomaly/intrusion. Unlike SVM (supervised learning), clustering is an anomaly detection method. Its advantage is the ability to detect unknown attacks, since it looks for deviations from the

"normal" cluster, rather than matching pre-known patterns. The disadvantage is often a higher false positive rate. Table 3 shows the effectiveness of the clustering method against different types of cyberattacks in terms of TPR and FPR.

Table 3
Effectiveness of the clustering method in terms of correct (TPR) and false positive (FPR) for different categories of attacks

Attack type	TPR (%)	FPR (%)	Note
DoS	90.0	10.0	Massive anomalies are easy to detect, but high traffic volumes increase FPR
DDoS	85.0	12.0	Similar, but less concentrated traffic makes the anomaly less pronounced
Probe	80.0	8.0	Detected as frequent, unusual requests (anomaly)
R2L	55.0	18.0	Low-frequency attacks that mimic legitimate remote access are difficult to distinguish from a normal cluster
U2R	45.0	20.0	Very rare events on the host that are difficult to separate from legitimate system administrator actions

Principal Component Analysis (PCA) is a powerful statistical tool widely used in intrusion detection systems (IDS) to reduce the dimensionality of data and detect anomalies. These methods transform the original feature space into a new space of lower dimension while preserving the most important information. The new features (components) are a combination of the original ones.

PCA is a linear method that uses an orthogonal transformation to transform a set of possibly correlated variables into a set of values of linearly uncorrelated variables, called principal components. It is solved in several steps.

1. Calculate the covariance matrix for the data X (where X is the data matrix centered around the mean):

$$C = \frac{1}{m-1} X^T X, \tag{10}$$

where m is the number of observations. The covariance matrix describes the relationship and variance between features.

2. Calculation of eigenvalues and eigenvectors of the covariance matrix. The principal components are defined by the eigenvectors vi of the covariance matrix C, and the amount of variance explained by each component is defined by the corresponding eigenvalue λi:

$$C(v_i) = \lambda_i v_i, \tag{11}$$

The eigenvectors vi are sorted in descending order of eigenvalues λi. The vector corresponding to the largest eigenvalue is the first principal component (the direction of maximum variance), the next is the second, etc.

3. K eigenvectors (principal components) corresponding to the largest eigenvalues are selected (usually k is chosen to explain 90-95% of the total variance). These k vectors form the transformation matrix W:

$$W = [v_1, v_2, \ldots, v_k]. \tag{12}$$

4. Projection: The original data X is projected onto the subspace formed by these k eigenvectors (projection matrix W):

$$Z = XW, \tag{13}$$

where Z is the reduced-dimensional data matrix.

To detect anomalies, the Reconstruction Error is used. It measures how much the observation deviates from the subspace defined by the normal data:

Data reconstruction: the reduced Z data is projected back into the original space:

$$\widehat{X} = Z\,W^T + \mu. \tag{14}$$

Error calculation: The reconstruction error E for each observation is calculated as the square of the distance between the original x_i and reconstructed $\hat{x}_i$ vector:

$$E(x_i) = \left\| x_i - \hat{x}_i \right\|^2. \tag{15}$$

If $E(x_i)$ exceeds a certain threshold, the observation x_i is classified as an attack (anomaly).

The principal component analysis (PCA) is primarily a dimensionality reduction technique or a statistical method for anomaly detection (via reconstruction error analysis), rather than a classifier for specific types of attacks.

When PCA is used for intrusion detection, it works as an anomaly detector: the model is trained on normal traffic and any significant deviation from this "normal" state (measured by a large reconstruction error) is flagged as an anomaly (an attack).

Thus, the results of the PCA method for specific attack categories (DoS, R2L) will be similar to the results of unsupervised clustering - high efficiency for unknown attacks and low accuracy for rare, subtle attacks. Table 4 shows the results of the efficiency of the principal component analysis (PCA) method in terms of correct TPR and false FPR recognition.

Table 4

Efficiency of the principal component analysis (PCA) method in terms of correct TPR and false recognition FPR

Attack type	TPR (%)	FPR (%)	Note
DoS	92.0	7.0	Easy to detect, as massive traffic significantly changes the covariance matrix (large anomalies)
DDoS	88.0	8.5	Similar to DoS, but the distributed nature makes detection difficult, increasing FPR
Probe	78.0	11.0	Anomalies in port and protocol distribution. FPR is higher due to the complexity of “normal” scanning
R2L	40.0	18.0	Similar to legitimate remote access, so PCA does not separate them well from normal space
U2R	35.0	20.0	Rare events on the host that are not usually detected by network flow analysis methods such as PCA

Principal component analysis (PCA) is most commonly used as an anomaly detection technique by measuring the reconstruction error. This allows it to detect unknown attacks, but compromises accuracy for known, rare attacks.

The figure illustrates that PCA performs well in detecting anomalies that are very different from normal traffic (DoS), but as the complexity and stealth of the attack (R2L, U2R) increases, its performance drops sharply and the number of false positives increases significantly.

Hybrid methods that combine the strengths of supervised (SVM) and unsupervised (clustering, PCA) learning tend to show the best overall results, as they are able to effectively detect both known attacks and anomalies while maintaining a relatively low false positive rate.

Table 5 shows the performance data for a hybrid method that uses these three (clustering + SVM + PCA) methods.

Table 5
Efficiency table of the hybrid method (clustering + SVM + PCA) in terms of correct TPR and false recognition FPR

Attack type	TPR (%)	FPR (%)	Note
DoS	99.0	0.8	Exceptional accuracy characteristic of SVM on massive attacks
DDoS	97.0	2.0	Very high efficiency
Probe	98.0	2.5	High efficiency
R2L	85.0	5.0	Significant improvement in TPR compared to pure methods through the use of PCA to highlight important features
U2R	80.0	6.5	Significant improvement in TPR for rare attacks

The effectiveness of the hybrid method is based on the combination of the strengths of each of the three methods (table 6).

Table 6
Advantages and disadvantages of single methods

Method	Type of training	Strength (TPR)	Weak side (FPR)
SVM	Managed (Signature/Misuse)	Very high TPR for known attacks (DoS, DDoS, Probe)	Almost unable to detect R2L (TPR ≈5%)
Clustering	Unmanaged (Anomaly)	Detects U2R attacks well (TPR ≈80%)	Very high FPR (>6.5%) as anything that deviates from the norm is flagged as an attack
PCA	Unmanaged (Anomaly)	Detects Probe and massive anomalies (DoS) well	Low TPR for rare, subtle attacks (R2L/U2R); moderately high FPR

2. Further research options

A future research direction could be to explore the combinatorics of data mining and deep learning methods [23-27], as deep learning offers significant advantages over traditional data mining methods for attack detection. These advantages include automation, adaptability, and efficiency.

Automatic feature extraction. Traditional data mining methods, such as decision trees or support vectors, rely heavily on manual feature extraction. Cybersecurity experts must manually determine which characteristics of network traffic (e.g., packet count, protocol, ports) are important for identifying attacks. This process is time-consuming and prone to human error. In contrast, deep learning models (e.g., CNN, RNN) [28-30] can automatically learn hierarchical features directly from raw data. This allows them to detect non-obvious and complex patterns that may be invisible to a human analyst.

Detecting unknown attacks (zero-day attacks). Data mining methods often have limitations in detecting new, previously unknown attacks (zero-day attacks). Because their models are based on predefined features and patterns, they may not recognize attacks that do not match known patterns. In contrast, deep learning is particularly effective in anomaly analysis. It learns from normal network behavior and can detect any significant deviations that indicate a potential new threat, even if its type is unknown.

Processing big and high-dimensional data. Modern networks generate huge amounts of data that can have a very high number of dimensions (high-dimensional data). Traditional data analysis algorithms can be inefficient or scale poorly when working with such arrays. Deep neural

networks [31] are designed specifically to process large, complex, and high-dimensional data sets. They are able to scale effectively, making them ideal for monitoring high-speed networks in real time.

Reducing false positives. Due to their limited ability to extract complex features, traditional methods can often misclassify normal traffic as malicious, resulting in a high rate of false positives. This creates an additional burden on analysts. Deep learning, due to its ability to better distinguish legitimate behavior from anomalous behavior, helps reduce false positives, making the attack detection system more reliable and effective.

Table 7 clearly shows the advantages of deep machine learning over traditional data mining methods for detecting attacks, and their combinatorics will make it possible to increase the probability of correct recognition and reduce false ones.

Table 7
Comparison of deep machine learning over traditional data mining methods for attack detection

Comparison criterion	**Traditional data mining methods (SVM, Random Forest)**	**Deep learning (RNN, CNN, Autoencoders)**
Feature extraction	Manual. Requires domain knowledge and a lot of time	Automatic. The model independently detects complex hierarchical features
Detection of new (Zero-Day) attacks	Limited. Rules and templates must be constantly updated	High efficiency. The ability to perform anomaly analysis allows you to detect unknown attacks
Working with data volumes	Less scalable, performance drops on big data	Highly scalable. Work effectively with big and high-dimensional data
Precision (Accuracy)	Depends on the quality of manually selected features	Usually higher, as the model is trained on more complex and deeper patterns
False Positives	High level. Frequent misclassifications of normal traffic	Lower level. Models are better at distinguishing legitimate behavior from anomalous behavior
Computing resources	Less demanding.	More demanding on GPU/TPU for training
Comparison criterion	Traditional data mining methods (SVM, Random Forest)	Deep learning (RNN, CNN, Autoencoders)

The advantages of deep learning for attack detection using the NSL-KDD and UNSW-NB15 datasets can be presented in the form of Table 8, which demonstrates how different classes of attacks are processed more efficiently than traditional methods.

Let's take the same four main attack categories from the NSL-KDD and UNSW-NB15 datasets: DoS (Denial of Service), Probe (Probing), R2L (Remote to Local), U2R (User to Root).

As the table shows, deep learning is particularly effective for detecting rare and complex attacks (R2L and U2R), where traditional methods often fail due to their reliance on manual feature extraction and data imbalance. For more common attacks (DoS, Probe), deep learning provides higher accuracy and reduces the number of false positives, making intrusion detection systems significantly more reliable.

Table 8
Benefits of deep learning in detecting attacks from NSL-KDD, UNSW-NB15 datasets

Attack type	Characteristics of the attack	Problems for traditional methods	Advantages of deep learning
DoS	Large volume of traffic in a short time (for example, Smurf, Neptune)	It is easy to detect, but traditional methods can have a high false positive rate due to inaccurate threshold values	High accuracy and low false positives. CNN and RNN can analyze time sequences of traffic, accurately distinguishing DoS from normal activity spikes
Probe	Port and host scanning, often at low intensity (e.g. Portsweep, Nmap)	It is difficult to distinguish from normal network scanning. Rule-based methods can be too strict or too lenient	Automatic pattern extraction. Neural networks can learn subtle patterns that indicate a scan, even if it is stretched over time, eliminating manual adjustments
R2L	Small amount of traffic, but malicious actions in the data (e.g. Guess_passwd, FTP_write)	The most difficult class to detect. Traditional methods based on packet headers are ineffective because the attack occurs at the content level. The low number of samples in the KDD99 dataset makes training difficult	Efficient content analysis: Recurrent neural networks (RNNs) and autoencoders can analyze sequences of bytes in packets, detecting anomalies in the data, not just the metadata
U2R	Rare events (e.g. Buffer_overflow, Rootkit) that change access rights	The rarest and most complex class. Only detected after an attack has already occurred. The low number of samples in the dataset leads to a strong class imbalance and poor accuracy	Ability to work with unbalanced data. Deep learning, especially when combined with unbalanced data processing methods, can learn rare U2R patterns, significantly improving their detection accuracy

Table 9 shows a comparison of the effectiveness of deep learning and traditional data mining methods in detecting attacks from the DoS, Probe, R2L and U2R categories.

Table 9
Comparison of attack detection using data mining and deep learning methods

Attack type	Characteristics of the attack	Efficiency of traditional methods (e.g. SVM)	Deep learning efficiency (e.g. RNN, CNN)	Benefits of deep learning
DoS	Attacks with large traffic volumes (e.g., SYN flood)	High because the attack has clear signs (large number of packets). There may be false positives due to incorrect threshold settings	Very high, as models can analyze traffic time series and accurately distinguish DoS from normal activity spikes	Lower false positive rate; better distinction between an attack and a normal traffic spike
Probe	Low-intensity reconnaissance attacks (e.g., port scanning)	Medium, depends on configured rules and signatures. May skip slow scans or detect regular administrative scans as a threat	High, due to the ability to automatically extract subtle patterns. Models can detect anomalies that are "stretched" in time	Automation and detection of hidden patterns. Better distinction between legitimate and malicious scans
R2L	Attacks that use packet content for remote access	Low, as the attack does not depend on traffic volume. Methods that analyze only metadata are ineffective	High, because models (e.g., RNN) can analyze packet content and byte sequences, detecting anomalies in the content, not just in the traffic	Analyze data content, not just metadata. Better detect rare data-based attacks
U2R	Attacks to elevate privileges are very rare events	Very low, due to the severe class imbalance and rarity of attacks. Traditional methods can completely ignore such rare events	Medium/High, models can be trained on unbalanced data, detecting even rare events that indicate a change in privileges	The ability to work with unbalanced data, which allows you to identify the rarest and most

Table 10 shows the rates of correct and incorrect recognition of SVRs built on the basis of data mining and deep learning methods.

Table 10
The rates of correct and incorrect recognition of SVRs built on the basis of data mining and deep learning methods

Attack type	Method	Accuracy of correct recognition (%)	False Positives (%)
DoS	Traditional (SVM, Random Forest)	98-99	2-5
	Deep learning (RNN, CNN)	99-99.8	< 1
Probe	Traditional (SVM, Random Forest)	95-97	3-6
	Deep learning (RNN, CNN)	> 98	< 2
R2L	Traditional (Decision Trees)	70-85	10-20
	Deep learning (Autoencoders, RNN)	85-95	< 5
U2R	Traditional (Naive Bayes)	60-75	15-25
	Deep learning (RNN, GANs)	75-90	< 10

The use of deep learning (DL) in intrusion detection systems (IDS) is one of the most promising areas for combating cyberattacks. Deep learning models, such as convolutional neural networks (CNN), recurrent neural networks (RNN) and their variants (LSTM, GRU), as well as deep neural networks (DNN), demonstrate high efficiency in detecting various types of attacks due to their ability to automatically learn complex features under uncertainty.

The performance of IDS based on deep learning is often evaluated by metrics derived from the Confusion Matrix, in particular, the True Positive Rate (TPR), known as the Recall or Detection Rate, and the False Positive Rate (FPR), known as the False Alarm Rate.

3. Conclusions

The hybrid method uses each component to compensate for the shortcomings of the others. Thus, PCA is used to optimize the input data. It discards redundant information, leaving only those features (principal components) that best distinguish between normal and anomalous behavior. This reduces the complexity for SVM and clustering and increases their sensitivity to subtle changes (R2L/U2R). SVM is the main classifier for high-frequency and known attacks (DoS, DDoS). It provides an extremely low FPR (<5%), filtering out the vast majority of legitimate traffic. Clustering/anomalous detection for "unknown" threats (high TPR): unmanaged components (clustering or PCA) work in parallel, scanning traffic that the SVM did not classify for anomalies. This allows the detection of Zero-Day attacks and rare R2L/U2R attacks that do not have known signatures.

Thus, the hybrid method uses the principle of multi-layered protection. It combines the high accuracy of SVM for known threats with the high sensitivity of clustering/PCA for unknown threats, providing an optimal balance between correct and false positives in a real-world environment.

In the direction of further research, it is advisable to consider combinatorics of data mining and deep learning methods, methods for optimizing features from known datasets.

Declaration on Generative AI

While preparing this work, the authors used the AI programs Grammarly Pro to correct text grammar and Strike Plagiarism to search for possible plagiarism. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] Y. H. Chen, Y. Y. Yao. A multiview approach for intelligent data analysis based on data operators. Inf. Sci. 2008, 178, 1–20.

[2] J. Yang, et al. Brief introduction of medical database and data mining technology in big data era. J. Evid.-Based Med. 2020, 13, 57–69.

[3] T. Young, D. Hazarika, S. Poria, E. Cambria. Recent trends in deep learning based natural language processing. IEEE Comput. Intell. Mag.,13, 2017, 55–75.

[4] S. Abkenar, M. Kashani, E. Mahdipour, S. Jameii. Big data analytics meets social media: A systematic review of techniques, open issues, and future directions. Telemat. Inform. 2020, 57, 101517.

[5] D. Lande, I. Subach, Y. Boiarynova. Osnovy teorii i praktyky intelektualnoho analizu danykh u sferi kiberbezpeky: navchalnyi posibnyk, 2018. [Ukrainian].

[6] S. Yevseiev, et al. Metodolohiia syntezu modelei intelektualnykh system upravlinnia ta bezpeky obiektiv krytychnoi infrastruktury. Monohrafiia. Novyi Svit, 2024. [Ukrainian].

[7] V. Buriachok. Osnovy formuvannia derzhavnoi systemy kibernetychnoi bezpeky: Monohrafiia. NAU, 2013. [Ukrainian].

[8] V. Buriachok, V. Tolubko, V. Khoroshko, S. Toliupa. Informatsiina ta kiberbezpeka: sotsiotekhnichnyi aspekt: pidruchnyk. DUT, 2015. [Ukrainian].

[9] S. Toliupa, N. Lukova–Chuiko, V. Nakonechnyi, M. Brailovskyi. Metody intelektualnoho rozpodilu danykh v systemakh vyiavlennia merezhevykh vtorhnen ta funktsionalna stiikist informatsiinykh system do kiberatak. Monohrafiia. Format, 2021.

[10] E. Kabir. A novel statistical technique for intrusion detection systems. Future Generation. Comput. Syst., 79 (2018) 303–318.

[11] H. Tianfield. Data mining based cyber-attack detection Syst. simul. Technol., 13, 2017.

[12] B. Subba, S. Biswas, S. Karmakar. Intrusion detection systems using linear discriminant analysis and logistic regression. 2015 Annual IEEE India Conf. (INDICON), IEEE, 2015, 1–6.

[13] K. Peng, et al. Intrusion detection system based on decision tree over big data in fog environment. Wireless Commun. Mobile Comput., 2018, 1–10.

[14] Y. Xue, W. Jia, X. Zhao, W. Pang. An Evolutionary Computation Based Feature Selection Method for Intrusion Detection. Security and Communication Networks, 2018, 1–10.

[15] L. Xiao, Y. Chen, C. K. Chang. Bayesian model averaging of Bayesian network classifiers for intrusion detection. 2014 in IEEE 38th Int. Computer Software and Applications Conf. Workshops on 35, 2014, 1302–1310.

[16] H. Om, A. Kundu. A hybrid system for reducing the false alarm rate of anomaly intrusion detection system. 2012 1st Int. Conf. on Recent Advances in Information Technology (RAIT), IEEE, 2012, 131–136.

[17] Z. Muda, et al. Intrusion detection based on k-means clustering and OneR classification. 2011 7th Int. Conf. on Information Assurance and Security (IAS), 2011, 192–197.

[18] U. Chowdhury, et al. A few-shot DL approach for improved intrusion detection. 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), 2017, 456–462.

[19] H. Debar, M. Dacier, A. Wespi. Towards a Taxonomy of Intrusion Detection Systems. Computer Networks, 31, 1999, 805–822.
[20] Debar, H., Dacier, M., and Wespi, A. (2000), "A Revised Taxonomy for Intrusion–Detection Systems," presente dat Annalesdes communications, vol. 55, 2000, pp. 361–78.
[21] Kabiri, P., and Ghorbani, A., A. (2005), "Researchon Intrusion Detectionand Response: A Survey", International Journalof NetworkSecurity, Vol.1, No.2, Sep. 2005,pp.84–102.
[22] S. Buchyk, S. Toliupa, O. Buchyk, A. Shevchenko. Method for Detecting Phishing Sites. In: Digital Ecosystems: Interconnecting Advanced Networks with AI Applications. TCSET 2024. Lecture Notes in Electrical Engineering, 1198, 2024, 301–323, doi:10.1007/978-3-031-61221-3_15
[23] A. A. Ghorbani, WeiLu, M. Tavallaee, Network Intrusion Detectionand Prevention: concepts and techniques. Springer, 2010, 27–49.
[24] B. Sharma, L. Sharma, C. Lal, S. Roy. Anomaly based network intrusion detection for IoT attacks using deep learning technique. Comput. Electr. Eng., 2023, 107, 108626.
[25] D. Sudyana, et al. Improving generalization of ML-Based IDS with Lifecycle-Based dataset, Auto-Learning features, and deep learning. In Proceedings of the IEEE Transactions on Machine Learning in Communications and Networking, Arlington, 2024, 2, 645–662.
[26] P. Jayant, et al. Intrusion detection in network traffic using LSTM and deep learning. In 2024 15th Int. Conf. on Computing Communication and Networking Technologies (ICCCNT), 1–6.
[27] M. Narender, B. N. Yuvaraju. Deep regularization mechanism for combating class imbalance problem in intrusion detection system for defending DDoS attack in SDN. J. Comput. Sci., 2023, 19, 334–344.
[28] R. Jablaoui, N. Liouane. Efficient RNN model for IoT intrusion detection systems. In Proceedings of the 2024 International Conference on Control, Automation, and Diagnosis (ICCAD), 2024, 1–6.
[29] H. Zhao, et al. CAN intrusion detection system based on data augmentation and improved Bi-LSTM. In 2024 IEEE Asia Pacific Conf. on Circuits and Systems (APCCAS), 2024, 198–202.
[30] O. Elghalhoud, K. Naik, M. Zaman, R. S. Manzano. Data balancing and CNN based network intrusion detection system. In Proceedings of the 2023 IEEE Wireless Communications and Networking Conference (WCNC), 2023, 1–6.
[31] W. Wan, et al. An effective integrated intrusion detection model based on deep neural network. In Proceedings of the 2021 Int. Conf. on Computer Engineering and Application (ICCEA), 2021, 146–152.