Journal of Distributed Computing and Systems (JDCS)
Vol. 8, Issue 1, 2025, Pages 54-70

Analysis of machine learning algorithms towards cyberattacks detection: a survey

Hanieh Khosravi[1], Elham Fereydoonifard[1], Zoleikha Jahanbakhsh Naghadeh[2*]
[1]*Department of Computer Engineering, ST.C., Islamic Azad University, Tehran, Iran.*
[2]*Department of Computer Engineering, Nag.C., Islamic Azad University, Naghadeh, Iran.*

Article History:
Received: 14 May 2025
Received in revised form: 27 July 2025
Accepted: 17 August 2025
Available online: 16 September 2025

Abstract

The rising complexity of cyber threats calls for more sophisticated and automated detection mechanisms. This paper provides a thorough review and performance comparison of traditional machine learning algorithms for detecting major cyberattacks, including Distributed Denial of Service (DDoS), phishing, malware, ransomware, SQL injection, zero-day exploits, and Man-in-the-Middle (MitM) attacks. We evaluate widely used algorithms such as Random Forest, Gradient Boosting, Support Vector Machines (SVM), Decision Trees, Naïve Bayes, and K-Nearest Neighbors based on key metrics like accuracy, detection rate, and computational efficiency. Our findings indicate that ensemble methods, particularly Random Forest and Gradient Boosting; consistently achieve high performance across diverse attack scenarios, whereas simpler models often struggle with complex or evolving threats. The study also discusses the emerging role of deep learning in cybersecurity, highlighting its potential for advanced threat detection alongside current challenges such as high computational demands and data dependencies. This review serves as a practical guide for selecting effective ML-based detection tools and points toward future integrations of ML and DL for stronger cyber defense.

Keywords: Machine Learning, Cyberattack Detection, Intrusion Detection, Performance Evaluation, Deep Learning, DDoS, Ransomware.

I. INTRODUCTION

In the digital age, the rapid growth of new technologies such as the Internet of Things (IoT) [1], connected devices, In the digital age, the rapid growth of new technologies such as the Internet of Things (IoT) [1], connected devices, wireless network [8], extremely high volumes of data [2], along with significant advances in computer systems and communication networks [3], have led to an increase in cyber security threats. These threats include sophisticated attacks such as distributed denial of service (DDoS) attacks [4], Malware attack [5], Equifax data breach [6], SQL injection, zero-day attacks and phishing, etc. [7] which can cause extensive financial and operational losses for organizations and individuals [9]. Since businesses, governments, and private citizens must have access to extremely secure apps and technologies, as well as the ability to quickly identify and eradicate cyber threats, one of the most important problems that needs to be resolved immediately is how to efficiently identify different cyber occurrences [13]. Protecting computer-based systems from cyberattacks has been getting more difficult. For instance, it takes an average of 240 days to find an intrusion. In addition, the complexity of attacks is growing every day due to the introduction of new attack types, and security flaws are continuously growing. It is becoming more and more difficult to keep up with this pace and stop attacks. Over time, it has been observed that traditional cyber security methods, relying on firewalls and intrusion detection systems, are unable to identify and counter modern sophisticated attacks and are ineffective against new threats. In the context, Machine Learning (ML) algorithms, have been considered as new tools for predicting and identifying cyber threats. These algorithms are capable of predicting and identifying advanced cyber-attacks by analyzing large and complex data and identifying unusual behavior patterns [3] [10] [11] [12].

Numerous industries, such as healthcare, banking, and cybersecurity, have shown promise in machine learning (ML) and deep learning (DL) approaches, which are particularly good at analyzing datasets and forecasting outcomes based on past data [14]. By learning from past attack patterns and spotting anomalies in network data, these techniques are used in cybersecurity to detect network intrusions [15]. In IDS, a number of machine learning (ML) methods, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Decision Tree (DT), are frequently used to categorize network traffic as either malicious or benign [16].

In this work, we present a survey of the literature on traditional machine learning (ML) techniques for

* Corresponding Author: Zoleikha Jahanbakhsh Naghadeh (zoleikha.jahanbakhsh@iau.ac.ir)

cybersecurity applications, focusing on methods such as Random Forest, SVM, and Naïve Bayes. We also discuss some major cyberattacks in network intrusion detection. Our aim is to provide researchers with a deeper understanding of ML techniques in cyberattack detection. While our primary focus is on traditional ML approaches, it is important to acknowledge the growing role of deep learning (DL) in cybersecurity. In recent years, DL has gained significant attention due to its ability to analyze complex data, detect unknown patterns, and identify advanced cyber threats such as zero-day attacks and polymorphic malware. Modern cybersecurity systems increasingly leverage deep learning models, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), to enhance threat detection and response.

The main contributions of this paper are summarized as follows:

1. **Comprehensive Survey:** Provides a review of machine learning (ML) techniques for detecting cyberattacks, addressing the challenges and effectiveness of some algorithms.
2. **Comparison of Techniques:** Compares various ML methods in terms of their application to specific types of cyberattacks, such as DDoS, phishing, ransomware, SQL injection, and zero-day exploits, highlighting their strengths and limitations.
3. **Insights into Cybersecurity Challenges:** Identifies the challenges ML models face in detecting sophisticated cyberattacks, including issues of scalability, high-dimensional datasets, and evolving threat patterns.
4. **Considering Deep Learning in Cybersecurity:** This study extends the discussion to deep learning (DL) methodologies, evaluating their role in cybersecurity applications. It examines how DL models such as CNNs, RNNs, Autoencoders, GANs, and Transformers contribute to cyber threat detection, network traffic analysis, and zero-day attack prevention. The study also highlights the advantages of DL over traditional ML, including automated feature extraction and improved performance on large-scale datasets, while addressing the challenges of computational cost, data requirements, and interpretability.
5. **Future Directions:** Suggests hybrid and ensemble approaches, integrating ML with DL for improved accuracy, and proposes solutions like real-time anomaly detection, adversarial learning, and federated learning for enhanced cyberattack detection.

Our research paper is structured as follows. Section 2 discusses the research methodology used in this study, followed by Section 3, which provides an overview of cybersecurity, explaining its various domains and fundamental concepts. Section 4 examines different types of cyberattacks, focusing on those considered in this study. Section 5 explores defense mechanisms and tools against cyberattacks. Building on this, Section 6 introduces machine learning as a modern approach for detecting and preventing cyber threats, analyzing various techniques and algorithms studied in this research. Section 7, which presents the results and discussion, evaluates prior research and compares the accuracy and detection rates of different algorithms in identifying the targeted attacks. Additionally, Section 8 examines deep learning as a key advancement that enhances conventional machine learning techniques in cyberattack detection. Finally, Section 9 provides a summary of findings, and Section 10 discusses potential future research directions in this field.

II. RESEARCH STRATEGY

This study aims to systematically evaluate the performance of some traditional and foundational machine learning algorithms across various types of cyberattacks, including DDoS, Phishing, Malware, Ransomware, SQL Injection, Zero-Day Attacks and MITM. The goal is to identify the most suitable and least effective algorithms for each attack type by analyzing their strengths, weaknesses, accuracy, and detection rates. In addition to performance evaluation, this research highlights the advantages and limitations of each algorithm in different cyberattack scenarios. By providing a comprehensive comparison, the study serves as a guide for selecting the most appropriate algorithm tailored to specific security challenges.

As a future direction, this paper proposes investigating hybrid approaches, combining these traditional algorithms with advanced techniques like deep learning, or enhancing their architectures to further improve their detection accuracy, scalability, and robustness in real-world applications.

To evaluate the performance of various algorithms, we reviewed articles published by reputable publishers such as Springer, Elsevier, MDPI, IEEE, and others. The pie chart below illustrates the distribution of reviewed articles by publisher. This approach ensures that our evaluation is based on credible and comprehensive sources.

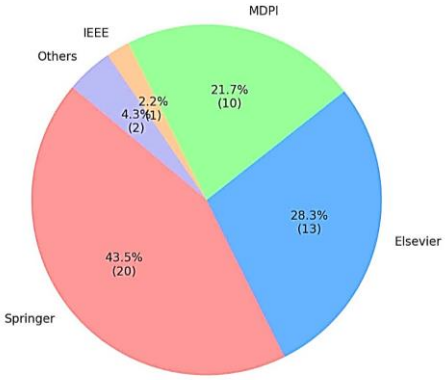

Figure 1. Distribution of Reviewed Articles by Publisher

III. CYBER SECURITY

The information and communication technology (ICT) sector has undergone significant change over the past 50 years and is now widely used and intricately woven into our contemporary culture. Therefore, in recent years, security policymakers have been quite worried about safeguarding ICT systems and applications against cyberattacks [20]. The term cybersecurity refers to the process of safeguarding these ICT systems against different cyber threats or attacks [21]. It can be defined a collection of technologies and procedures intended to prevent attacks, unauthorized access, alteration, or destruction of computer networks, software programs, and data, as well as a set of guidelines that can be used to prevent these threats [22] [13].

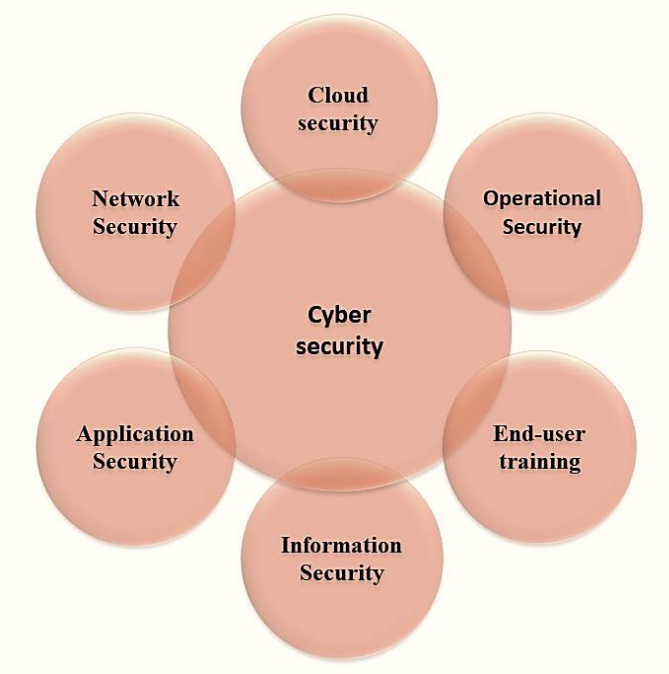

Figure 2. Different kinds of cyber security

Fig.2 illustrates different types of cyber security:

- **Information security** guards against illegal access, disclosure, misuse, alteration, and deletion of both digital and physical data [25].
- **Application Security:** Using hardware and software (such as antivirus applications, encryption, and firewalls) secures the system from external threats that may interfere with application development, which are the traditional methods being used [24].
- **Operational security:** Decisions and procedures made to manage and safeguard data are included in operational security. For instance, processes that define when and where data may be shared or stored, or user rights for gaining access to the network [25].
- **Cloud security:** safeguards data on the cloud (based on software) and keeps an eye out to eliminate the risks of on-site attacks [26].
- **User training:** Addresses the unpredictability of cybersecurity, specifically people, as anyone can unintentionally infect the security system with a virus. Any company's corporate security plan should include instruction on how to weed out suspicious attachments in emails, avoid connecting to anonymous USBs, and other important matters [26].
- **Network security** protects computer networks from intrusions and attacks, including malware and hacking, using a suite of tools designed to prevent organized cyberattacks and unauthorized access [23]. The increasing complexity of these attacks poses significant challenges, as intruders employ advanced methods to exploit system vulnerabilities [27]. Examples of sophisticated cyber threats include advanced persistent threats, zero-day exploits, polymorphic and fileless malware, and multi-vector attacks, all designed to bypass defenses and compromise network integrity [28].

IV. CYBERATTACKS

Cyberattacks are deliberate actions targeting data, networks, or computer systems, often carried out in coordinated stages to achieve specific objectives [29]. Major threats include intentional damage or service disruptions that cause significant data loss or financial impact, often with long-term consequences [30]. Attacks may originate from malicious insiders or invasive applications installed unknowingly by users. Advanced behavioral anomaly detection and auto-resiliency mechanisms are developed to proactively identify and mitigate such threats at both application and employee levels [31].

IoT networks are particularly vulnerable, with attacks exploiting weaknesses in device architecture,

communication protocols, or data handling, compromising integrity, confidentiality, or availability. Machine learning (ML) algorithms are increasingly evaluated for their effectiveness in detecting and preventing these threats.

A. DDoS (Distributed Denial of Service)

A DDoS attack seeks to overwhelm a server, network, or application by flooding it with massive amounts of traffic from multiple compromised systems, often called a botnet. This renders the target system unable to respond to legitimate requests, leading to service disruption. Unlike standard denial-of-service (DoS) attacks, DDoS leverages numerous sources to amplify its impact, making it difficult to mitigate. These attacks are often used to damage reputations, disrupt businesses, or divert attention during other cyber activities [60].

B. Phishing

Phishing involves attackers impersonating trusted entities to deceive victims into revealing sensitive information such as login credentials, credit card details, or other personal data. Attackers use emails, websites, or text messages that appear legitimate to lure users into clicking malicious links or downloading malware. Once the victim interacts with the fraudulent source, their information is compromised. The attack often relies on psychological manipulation, exploiting human error and trust [67].

C. Malware

Malware, or "malicious software," refers to a broad category of harmful programs designed to damage systems, steal data, or gain unauthorized access. This includes ransomware, which encrypts data for ransom; Trojan horses, which disguise themselves as legitimate software; worms, which self-replicate and spread across networks; and viruses, which infect files or systems. Malware operates by exploiting system vulnerabilities, spreading through phishing emails, malicious downloads, or compromised websites. Its goal can range from data theft to disrupting critical operations [74].

D. Ransomware

Ransomware a type of Malware that encrypts a victim's data or locks them out of their systems, demanding payment (often in cryptocurrency) to restore access. This type of malware spreads through phishing emails, infected attachments, or malicious software updates. Victims, ranging from individuals to large enterprises, are coerced into paying ransoms under the threat of data destruction or public release. Advanced ransomware variants may also exfiltrate data before encrypting it, adding a secondary layer of extortion [82].

E. SQL Injection

SQL injection involves embedding malicious SQL code into a database query via input fields on web applications. This vulnerability allows attackers to manipulate queries and gain unauthorized access to sensitive data stored in a database. For example, they might extract, modify, or delete critical information. SQL injection attacks are commonly used to steal customer data, bypass authentication, or even control entire databases if not mitigated by proper input validation [90].

F. Zero-Day Exploit

A zero-day exploit targets vulnerabilities that are unknown to the software vendor or security teams at the time of the attack. These attacks are particularly dangerous as they strike before patches or fixes can be deployed, leaving systems defenseless. Exploits often affect operating systems, applications, or firmware, allowing attackers to execute malicious code, steal data, or gain control of affected systems. Organizations must rely on heuristic or behavior-based detection systems to counteract these threats [95].

G. Man-in-the-Middle (MitM)

MitM attacks occur when a malicious actor intercepts and potentially alters communication between two parties without their knowledge. This attack may involve intercepting data transmitted over unsecured networks or forging communication between the two parties. Common examples include session hijacking, where the attacker takes over a user's session, and eavesdropping, where sensitive information such as passwords and financial data is stolen. MitM attacks exploit weak encryption protocols or poorly secured networks [105].

V. DEFENSE STRATEGIES AND TOOLS

Cyber defense constitutes a systematic and adaptive framework for safeguarding information systems, networks, and data against evolving cyber threats. Its primary objective is to maintain the confidentiality, integrity, and availability of critical assets. Core strategies include vulnerability assessments, deployment of firewalls and intrusion detection systems (IDS), continuous monitoring, and structured incident response, all aimed at risk mitigation, regulatory compliance, and organizational resilience [52] [53].

The increasing sophistication of threats, exemplified by advanced ransomware attacks, combined with the expanded attack surface due to IoT, cloud computing, and interconnected infrastructures, underscores the necessity of comprehensive defense mechanisms. Effective cyber defense requires alignment with regulatory frameworks, such as the Cyber Resilience Act, and adoption of emerging technologies, including quantum-resistant algorithms, to ensure adaptive and robust protection against adversaries.

Artificial Intelligence (AI) has become a critical tool in detecting and mitigating cyberattacks by enabling the identification of threats at unprecedented speed and accuracy. Below are some key ways in which AI contributes to detecting cyberattacks:

A. Anomaly Detection

AI systems analyze large datasets to identify unusual behaviors or patterns in network traffic, system activities, and user actions. For example, AI can detect deviations such as irregular login times, unexpected access requests, or abnormal data transfers, which may indicate potential security breaches. By learning from past data, AI systems can continually improve their ability to differentiate between legitimate behavior and threats [52] [53].

B. Threat Intelligence and Pattern Recognition

AI leverages machine learning algorithms to detect attack patterns, such as those seen in denial-of-service (DDoS) attacks, ransomware, or brute-force login attempts. AI-powered systems can identify these patterns faster and more accurately than traditional systems that rely on predefined signatures [54] [55]. This allows organizations to respond to new and emerging threats in real-time, often before significant damage occurs.

C. Malware Detection

AI can analyze the behavior of files and software to detect suspicious activities, even if the specific malware has never been seen before. Deep learning techniques help AI systems recognize previously unknown malware by learning the typical behavior of benign programs and comparing it to abnormal actions that could indicate malicious intent [53] [56].

D. Insider Threat Detection

Using machine learning and behavioral analytics, AI models can identify potential insider threats by monitoring deviations in user behavior. For example, if an employee suddenly accesses sensitive data outside of their usual role, AI can flag this as suspicious and trigger an alert. AI can also track patterns across large datasets, identifying inconsistencies that human security teams might miss [53] [54].

E. Automated Response and Incident Handling

AI's ability to detect threats quickly allows for faster response times. AI systems can autonomously take actions to contain or mitigate an attack, such as isolating affected systems, blocking suspicious IP addresses, or applying security patches, ensuring a swift and effective response to cyber threats [53] [56].

VI. ML, A NEW TOOL FOR CYBER DEFENSE

A subset of artificial intelligence (AI), machine learning (ML) is devoted to creating statistical models and algorithms that can analyze data and make predictions based on that data. In the context of cybersecurity, ML algorithms examine large datasets during training in order to identify patterns and deviations that may indicate the presence of threats. ML finds applications in a variety of cybersecurity domains, including intrusion detection, malware identification, network traffic analysis, and the detection of fraudulent activities. By analyzing the data in real-time, ML algorithms are able to identify and react to potential threats considerably more quickly than traditional rule-based systems. Table.1 shows the most important types of ML Techniques [33] [34] [35] [36] [37].

Machine learning (ML) algorithms are essential in cybersecurity for detecting, predicting, and preventing cyber threats. Each algorithm is tailored for specific problems and performs optimally depending on data characteristics and threat types, such as anomaly detection, regression, or classification. Selecting the appropriate algorithm is critical for effective threat identification and mitigation [57, 58].

The most commonly used ML algorithms in cybersecurity, summarized in Table 3, are valued for their ability to analyze large volumes of data, uncover hidden patterns, and protect networks, systems, and sensitive information from malicious activity.

Table 1. Different Types of Machine Learning [32].

Types of ML	Description
Supervised Learning	Supervised learning: During training, the model is guided by a set of labelled input-output pairs, which entails creating a mapping function from inputs (x) to outputs (y) through data analysis. Classification and regression tasks are common uses for supervised learning.
Unsupervised Learning	This method looks for patterns or structures in the data based on its intrinsic qualities rather than using labelled data for training. It focusses on problems like Dimensionality Reduction, Clustering, and Association Rule Learning without preset labels or outputs. In unsupervised learning, assaults frequently target language models.
Semi-supervised Learning	This approach, which combines aspects of supervised and unsupervised learning, trains models using a combination of labelled and unlabeled data. The labelled portion of the data is used to refine tasks, while the unlabeled data is used to improve interpretation. It is frequently used in tasks involving classification and regression.

Reinforcement Learning	Reinforcement learning (RL) is a process that involves interacting with an external environment and learning by making mistakes. This learning paradigm, which is notable for not having any reported privacy attacks, makes predictions about future events based on accumulated experiences.
Ensemble Learning	Ensemble learning entails combining several weak classifiers to form a strong classifier that derives its decisions from the combined predictions of the individual models. Methods like boosting and bagging serve as prime examples of ensemble learning techniques.

Table 2. Comparison of Machine Learning Algorithms. [38] [39] [40] [41] [59]

Algorithm	Advantages	Challenges	Effective Attacks
Logistic Regression	Fast; interpretable for linear problems	Ineffective for non-linear data; feature-sensitive	MITM, Phishing
Support Vector Machine	Work well in High-Dimensional Space; Kernel Flexibility; Robustness to Overfitting; Works Well with Imbalanced Data	Computationally Intensive; Sensitive to the Choice of Parameters; Lack of Scalability	Zero-day Exploits Man-in-the-Middle Attack
Naïve Bayes	Fast; suitable for text-based data	Assumes feature independence; struggles in dense data	Phishing, Spam Detection
Decision Tree	Interpretable; fast for small datasets	Prone to overfitting; limited scalability	Ransomware, Malware
Random Forest	Overfitting Robust Handles; High-Dimensional Data; Resilient to Noisy Data; Feature Importance	Computational Cost; less interpretable; Diminishing Performance in Imbalanced Datasets	DDoS, Malware, Ransomware
KNN	Simple; no training required	Computationally intensive; sensitive to noise	Malware, Ransomware
Gradient Boosting	High accuracy; handles imbalanced datasets well	Computationally expensive; parameter-sensitive	Phishing, SQL Injection

A. Analysis of Machine Learning Algorithms

This section presents a detailed analysis of some of the most used machine learning algorithms in cybersecurity. Advantages, challenges, and suitable attack scenarios of each algorithm are underlined for a better understanding of performance.

1. Logistic Regression (LR)

Logistic Regression is designed using the linear model for classification; it's used on two variables and labels them. It Applies logistic regression to classification problems, this method forecasts the result for a categorical dependent variable. It is especially made for tasks involving binary classification, where the result must be a separate or categorical value.

- **Advantages**: Fast and interpretable, suitable for simple decisions. Provide probabilities about class membership to support prioritization.
- **Challenges**: Can only handle data, which is linearly separable. May Suffer from irrelevant features fooling its predictions.
- **Effective Attacks:** MITM: Easy to detect obvious anomalies within the communication patterns. Phishing: Estimates likelihood of phishing attempts.

2. Support Vector Machine (SVM)

The Vector Machine finds a dividing hyperplane across classes by converting data into a higher-dimensional space. Even when the data are not linearly separable in the original space, this technique still works well.

- **Advantages:** SVM works great on high-dimensional feature space even when the number of samples is relatively small. The use of kernel functions, such as linear, polynomial, or radial basis function, enables SVM to deal with both linear and non-linear decision boundaries. The concentration on finding the maximum margin between classes keeps SVM from overfitting, even in high-dimensional spaces. SVM can be adjusted for class imbalance using techniques like class weight adjustment.
- **Challenges:** Training SVM on large datasets is resource-intensive, especially when non-linear kernels are adopted. Performance in SVM relies greatly on proper kernel selection and the optimal tuning parameters C (regularization) and γ (gamma). Training time for SVM increases aggressively with dataset size.
- **Suitable Attack Scenarios:** The subtle pattern detecting capability of SVM makes it effective

against zero-day vulnerabilities. SVM identifies small irregular deviations in encrypted traffic that can signify malicious interception.

3. Naïve Bayes (NB)

Naïve Bayes is based on Bayes' Theorem and assumes that the features are independent. It represents a probabilistic model, and its application is very appropriate for text classification with a large amount of data. This algorithm is particularly useful in detecting spam and social engineering attacks due to its ability to handle text-based features efficiently. It includes the following advantages and disadvantages.

- **Advantages**: Fast and computational efficiency. It works excellently in cases of feature independence, which is commonly experienced in phishing detection.
- **Challenges:** Assumes feature independence; hence, this approach would reduce accuracy for interdependent datasets. Performs very poorly in case of complex or dense data.
- **Suitable Attacks:** Phishing: It recognizes phishing keywords in email content. Spam Detection: It filters spam emails efficiently.

4. Decision Tree (DT)

Decision Tree is a simple and interpretable algorithm suitable for structured data. Decision Tree: This algorithm creates a model that resembles a tree. In order to forecast a target variable using basic decision rules, it divides data according to qualities, representing decisions with branches and outcomes with leaf nodes.

- **Advantages:** Fast training and prediction for small datasets. Supports both continuous and categorical features.
- **Challenges:** Prone to overfitting, especially in complex datasets. Performance declines with high-dimensional data.
- **Effective Attacks:** Ransomware: Detects defined malicious behaviors. Malware: Identifies simple behavioral patterns in malware.

5. Random Forest (RF)

Random Forest: An ensemble technique that averages the predictions of many DT classifiers, each trained on distinct data samples, to increase prediction accuracy and reduce overfitting.

- **Advantages:** The combination of multiple decision trees reduces the risk of overfitting in RF, which makes it more reliable in predicting compared to a single decision tree. RF works well on datasets that contain a large number of features and are complex in structure, which is typical for cybersecurity applications such as network intrusion detection. The nature of RF as an ensemble makes it robust against irrelevant or noisy features. RF gives feature importance, which gives insight into which attributes are most indicative of a particular attack.
- **Challenges:** RF model training might be computationally expensive, depending on the size of the datasets or the number of trees taken into consideration. The aggregated output of decision trees in RF makes it difficult to interpret its prediction. RF works worst in imbalanced datasets if applied without any additional methods involved, such as class weighting and sampling.
- **Applicable Attack Scenarios:** RF is very effective in detecting volume-based anomalies in network traffic on a large scale (i.e., Distributed Denial of Service (DDoS). RF can identify whether a file is malicious or not based on the characteristics of the behavior (i.e., Malware and Ransomware Detection).

6. K-Nearest Neighbors (KNN)

KNN is a non-parametric algorithm that classifies based on proximity to neighboring data points. It uses the majority class of its KNNs to classify each data point. This type of semi-supervised learning classifies data points based on how close together they are.

- **Advantages:** Simple to implement and has no training phase. Suitable for small datasets with clear patterns.
- **Challenges:** Very computationally expensive when running predictions on large-scale data. Sensitive to noise and irrelevant features.
- **Effective Attacks: Malware:** Classify malware samples by determining similar malware samples. Ransomware: Logistic regression can be used in these circumstances to find repeating bad practices.

7. Gradient Boosting (GB)

Gradient Boosting is an ensemble technique in which several weak learners are combined iteratively to enhance prediction accuracy. By merging several weak prediction models—usually DTs—into a single, more powerful model, gradient boosting improves predictive accuracy. This approach improves performance by gradually fixing the mistakes made by the weak learners.

- **Advantages:** It works effectively on imbalanced datasets; hence it is suitable for subtle attack patterns. Flexibility with various loss functions to adapt to the requirements.
- **Challenges:** Computationally expensive and time-consuming during training. Sensitive to the tuning of parameters, which should be carefully optimized.
- **Effective Attacks: Phishing:** It finds small hidden text-based patterns in phishing. SQL Injection: It finds deviations in the behavior of database queries.

VII. DISCUSSION

This section provides a detailed exploration and analysis of our research findings, focusing on the most significant machine learning algorithms for detecting various types of cyberattacks. Each subsection highlights the key algorithms relevant to each attack type, analyzing their performance metrics, including accuracy and detection rate. By systematically evaluating these algorithms, we aim to emphasize their strengths and limitations in addressing specific cyberattack scenarios.

Distributed Denial-of-Service (DDoS)

DDoS attacks are coordinated cyberattacks aimed at overwhelming network resources to disrupt service availability [29]. These attacks often follow multiple stages, exploiting system vulnerabilities to cause significant operational and financial damage [30]. Malicious insiders or compromised devices can facilitate such attacks, while unintended installation of invasive applications may also contribute to vulnerabilities. Advanced behavioral anomaly detection and auto-resiliency mechanisms are employed to detect and mitigate DDoS threats proactively at both the application and user levels [31]. IoT networks are particularly susceptible to DDoS attacks due to inherent weaknesses in device architecture, communication protocols, and data handling. Machine learning (ML) algorithms are increasingly applied to identify and prevent these attacks, with effectiveness evaluated based on accuracy, detection rate, and adaptability to high-dimensional network data. in certain contexts, especially for smaller or resource-constrained systems, where computational efficiency is a priority over detection accuracy.

Table 3. ML algorithms evaluation for DDoS detection

Reference	Year	Method	Accuracy	Detection Rate	Advantages	Disadvantages
[60]	2023	Random Forest	95.4%	97.1%	Handles large datasets well, robust to overfitting	High computational cost
[61]	2022	Gradient Boosting	93.2%	94.8%	Good for imbalanced datasets, high accuracy	Training time is longer
[62]	2022	Decision Tree	91.7%	92.3%	Easy to interpret, simple to implement	Overfitting issues with complex data
[63]	2023	Support Vector Machine	93.5%	95.2%	Effective for high-dimensional data	Time-consuming for large datasets
[64]	2022	K-Nearest Neighbors	87.4%	91.2%	Simple and efficient for small datasets	Sensitive to irrelevant features
[65]	2021	Logistic Regression	90.1%	92.5%	Fast and easy to implement, works well for linear patterns	Limited to linearly separable data
[66]	2021	Naive Bayes	88.3%	90.7%	Fast and effective for large datasets	Assumes feature independence

phishing attaks

The evaluation of machine learning methods for **phishing** detection indicates that Random Forest achieves the highest accuracy (96.5%) and strong detection rate (95.8%), effectively managing diverse features and noisy datasets due to its ensemble structure (Soni & Mishra, 2021 [67]). Gradient Boosting also performs well, with 94.3% accuracy and 93.5% detection rate, efficiently capturing misclassified instances in complex phishing datasets (Kumar, S. et al., 2020 [68]).

Table 4. ML algorithms evaluation for Phishing detection

Reference	Year	Method	Accuracy	Detection Rate	Advantages	Disadvantages
[67]	2021	Random Forest	96.5%	95.8%	Robust to overfitting, handles large datasets	High computational cost
[68]	2020	Gradient Boosting	94.3%	93.5%	Efficient for class imbalance, high accuracy	High training time

[73]	[72]	[71]	[70]	[69]
2020	2021	2020	2021	2021
Naive Bayes	Logistic Regression	K-Nearest Neighbors	Support Vector Machine	Decision Tree
84.7%	88.2%	85.9%	93.0%	91.4%
85.9%	89.1%	87.3%	92.2%	90.6%
Fast for large datasets, easy to implemen	Fast and easy to implement	Simple and efficient for small datasets	Works well for high-dimensional data	Easy to implement and interpret
Assumes feature independe nce	Limited to linear decision boundaries	Sensitive to irrelevant features	Time-consuming for large datasets	Overfitting with complex features

Table 5. ML algorithms evaluation for Malware detection

Reference	[74]	[75]	[76]
Year	2022	2021	2021
Method	Random Forest	Gradient Boosting	Decision Tree
Accuracy	98.4%	96.7%	94.8%
Detection Rate	97.6%	95.4%	93.5%
Advantages	High accuracy, works well for large datasets	Effective for imbalanced data, high accuracy	Easy to interpret, less computational overhead
Disadvantages	Requires significant computational resources	Long training time	Overfitting with complex datasets

[77]	[78]	[79]	[80]
2022	2021	2021	2021
Support Vector Machine	K-Nearest Neighbors	Logistic Regression	Naive Bayes
97.2%	92.3%	93.1%	90.8%
96.9%	90.7%	91.9%	89.3%
Effective with high-dimensional data	Simple and effective for small datasets	Easy to implement, works well for linear patterns	Fast and efficient for large datasets
Time-consuming for large datasets	Sensitive to irrelevant features	Limited performance for complex patterns	Assumes feature independence

In contrast, simpler models such as Logistic Regression (88.2%) and Naive Bayes (84.7%) show lower performance, often constrained by assumptions of linearity or feature independence (Singh & Ghosh, 2021 [72]; Kumar, A. et al., 2020 [73]). K-Nearest Neighbors (85.9%) is sensitive to noise and computationally intensive for large datasets (Zhang & Li, 2020 [71]), while Decision Tree, despite moderate accuracy (91.4%), may overfit and reduce generalization in real-world scenarios (Patel et al., 2021 [69]).

Table 6. ML algorithms evaluation for Ransomware detection

Reference	[81]
Year	2021
Method	Random Forest
Accuracy	98.5%
Detection Rate	97.2%
Advantages	High accuracy, effective for complex patterns
Disadvantages	High computational cost

[82]	2020	Gradient Boosting	95.3%	94.1%	Effective for imbalanced datasets, high	Longer training time
[83]	2021	Decision Tree	92.7%	91.4%	Simple and interpretable, fast to train	Prone to overfitting with noisy
[84]	2021	Support Vector	97.9%	96.5%	High precision, works well with high-dimension	High computational cost
[85]	2022	K-Nearest	89.4%	88.2%	Efficient for smaller datasets	Sensitive to irrelevan
[86]	2022	Logistic Regression	94.8%	93.6%	Fast to implement and easy to interpret	Limited performance with non-
[87]	2021	Naive Bayes	90.1%	89.0%	Fast and efficient for large datasets	Assumes feature independ

Malware attacks

Sun & Wang, (2022) [74] highlighted Random Forest as still the most effective method, achieving an accuracy of 98.4% and a detection rate of 97.6%. Its ability to process high-dimensional data and handle diverse **Malware** types contributes to its superior performance.

Liu et al., (2022) [77] also evidenced that Support Vector Machine (SVM) delivers competitive results, with an accuracy of 97.2% and a detection rate of 96.9%, excelling in scenarios requiring precise boundary identification.

On the other hand, according to *Wang et al., (2021)* [80] and *Chen & Wang, (2021)* [79], algorithms like Naive Bayes (90.8% accuracy) and Logistic Regression (93.1% accuracy) show limitations in dealing with complex feature interactions. Although Decision Tree achieves moderate results (94.8% accuracy), *Zhang et al. (2021)* [76] note its susceptibility to overfitting, reducing its effectiveness in generalizing to unseen data for **Malware** attacks.

Ransomware attacks

Random Forest demonstrates superior performance in **ransomware** detection, achieving 98.5% accuracy and 97.2% detection rate, effectively handling diverse attack patterns due to its ensemble structure (Zhang & Wang, 2021 [81]). Support Vector Machine (SVM) also performs strongly, capturing complex ransomware signatures with 97.9% accuracy and 96.5% detection rate (Yadav et al., 2021 [84]).

In contrast, simpler models such as Naive Bayes (90.1%) and K-Nearest Neighbors (89.4%) are less effective on high-dimensional data with feature interactions (Zhang et al., 2021 [87]; Liu & Zhang, 2022 [85]). Decision Tree (92.7%) and Logistic Regression (94.8%) provide moderate performance but face limitations in scalability and detecting sophisticated **ransomware** techniques (Kumar & Patel, 2021 [83]; Zhao & Wang, 2022 [86]).

SQL Injection attacks

The analysis of machine learning models for **SQL Injection** detection highlights Random Forest as the most effective method, achieving an accuracy of 97.5% and a detection rate of 96.8% based on the findings of *Gupta & Sharma, (2021)* [88]. Its ability to handle high-dimensional features and imbalanced datasets makes it highly suitable for detecting SQL Injection attacks in complex web environments.

Support Vector Machine (SVM) also performed well, with an accuracy of 95.8% and a detection rate of 94.6%, showing strength in distinguishing subtle **SQL Injection** attack patterns within network traffic *(Zhao et al., 2022)* [91].

On the other hand, simpler methods like Naive Bayes (88.6% accuracy) and K-Nearest Neighbors (89.7% accuracy) were less effective, struggling with noisy and correlated features, as noted by *Yadav & Singh (2020)* [94] and *Zhang et al. (2021)* [92].

While Logistic Regression (92.4% accuracy) and Decision Tree (91.9% accuracy) achieved moderate results, *Chen & Wang (2021)* [93] and *Lee et al. (2021)* [90] emphasized their limited scalability and reduced performance when dealing with large and dynamic datasets.

Table 7. ML algorithms evaluation for SQL Injection detection

Reference	**Year**	**Method**	**Accuracy**	**Detection Rate**	**Advantages**	**Disadvantages**
[88]	2021	Random Forest	97.5%	96.8%	High accuracy, effective for complex	Requires high computational resources
[89]	2020	Gradient Boosting	94.2%	93.4%	Effective for imbalanced data, high detection	Long training time

[94]	[93]	[92]	[91]	[90]
2020	2021	2021	2022	2021
Naive Bayes	Logistic Regression	K-Nearest Neighbors	Support Vector	Decision Tree
88.6%	92.4%	89.7%	95.8%	91.9%
87.2%	91.1%	88.3%	94.6%	90.5%
Fast and efficient for large datasets	Fast and easy to implement	Efficient for smaller datasets	Works well with high-dimensional data	Simple to interpret, fast training time
Assumes feature independence	Limited to linear decision boundaries	Sensitive to irrelevant features	Time-consuming for large datasets	May overfit in noisy datasets

Table 8. ML algorithms evaluation for Zero-day Attacks detection

[96]	[95]	Reference
2021	2022	Year
Gradient Boosting	Random Forest	Method
95.4%	97.8%	Accuracy
94.8%	96.2%	Detection Rate
Effective for class imbalance, high	High detection rate, works well for large datasets	Advantages
Longer training time	High computational cost	Disadvantages

[101]	[100]	[99]	[98]	[97]
2022	2021	2021	2022	2021
Naive Bayes	Logistic Regression	K-Nearest Neighbors	Support Vector Machine	Decision Tree
90.2%	92.8%	89.6%	98.2%	91.7%
89.6%	91.3%	88.2%	97.5%	90.4%
Simple and efficient, works well with large datasets	Fast and easy to implement	Efficient for small datasets	Works well with high-dimensional data	Simple to implement and interpret
Assumes feature independence	Not effective for complex patterns	Sensitive to irrelevant features	High time complexity for large datasets	Overfitting with complex data

Table 9. ML algorithms evaluation for MitM Attacks detection

[102]	Reference
2021	Year
Random Forest	Method
96.3%	Accuracy
95.2%	Detection Rate
High accuracy, effective for complex attacks	Advantages
Computationally expensive	Disadvantages

[103]	2021	Gradient Boosting	94.5%	93.6%	Effective for imbalanced data, high detection	Longer training time
[104]	2021	Decision Tree	92.8%	91.2%	Simple and interpretable, fast to train	Prone to overfitting in some datasets
[105]	2022	Support Vector Machine	97.1%	96.3%	High precision for high-dimensional data	High computational cost for large datasets
[106]	2021	K-Nearest Neighbors	90.4%	89.1%	Efficient for small datasets	Sensitive to irrelevant features
[107]	2021	Logistic Regression	91.7%	90.3%	Easy to implement and fast	Limited performance with non-linear data
[108]	2022	Naive Bayes	88.9%	87.6%	Fast and efficient for large datasets	Assumes feature independence

Zero-day attacks

Findings reveal Support Vector Machine (SVM) as the top-performing algorithm, achieving an impressive accuracy of 98.2% and a detection rate of 97.5% *(Wu et al., 2022)* [98]. Its ability to identify complex patterns in network traffic, even with limited training data, makes it exceptionally suited for detecting unknown attack vectors.

Random Forest follows closely with 97.8% accuracy and 96.2% detection rate, excelling in handling imbalanced datasets and complex feature sets for **Zero-day** attacks according to *Singh & Kumar, (2022)* [95].

While ensemble-based methods like Gradient Boosting (95.4% accuracy) demonstrate consistent performance, simpler approaches such as Naive Bayes (90.2% accuracy) and K-Nearest Neighbors (89.6% accuracy) underperform due to their limitations in processing high-dimensional data and dealing with noise, as noted.

Man-in-the-Middle (MITM) attacks

In the detection of Man-in-the-Middle (MITM) attacks, Support Vector Machine (SVM) demonstrates the highest performance with 97.1% accuracy and 96.3% detection rate, effectively identifying subtle anomalies in encrypted communications (Wang et al., 2022 [105]; Zhang & Li, 2021 [102]). Random Forest also performs strongly, achieving 96.3% accuracy and 95.2% detection rate, handling diverse and high-dimensional features robustly.

Gradient Boosting provides consistent results (94.5% accuracy) but requires greater computational resources (Yang et al., 2021 [103]). Simpler models, including Naive Bayes (88.9%) and K-Nearest Neighbors (90.4%), are less effective with noisy or correlated data (Gupta et al., 2022 [108]; Lee et al., 2021 [106]). Logistic Regression (91.7%) and Decision Tree (92.8%) offer moderate performance but have limitations in scaling to large and dynamic network traffic (Zhao & Chen, 2021 [107]; Singh & Choudhary, 2021 [104])

VIII. DEEP LEARNING IN CYBERSECURITY

While our primary focus has been on traditional machine learning algorithms such as SVM, Random Forest, Logistic Regression, KNN, Gradient Boosting, and Naive Bayes for cyberattack detection, it's undeniable that DL algorithms also play a crucial role in enhancing cybersecurity systems. Their ability to process massive datasets, identify unknown patterns, and adapt to evolving threats has made them indispensable in advanced cyber defense strategies. Below, we briefly discuss their significance and applications:

A. Importance of Deep Learning in Cybersecurity

1. Processing Large-Scale Data:

Deep learning excels in handling massive datasets, such as network traffic logs and system logs, enabling comprehensive anomaly detection.

2. Recognizing Unknown Patterns:

Unlike traditional algorithms, deep learning models automatically extract features and detect previously unseen attack patterns without manual intervention.

3. Advanced Applications:

- **Malware Detection:** Effectively classifies and detects new malware types.
- **Network Traffic Analysis:** Identifies malicious activities in real-time network traffic.
- **Zero-Day Attack Prevention:** Anticipates and mitigates vulnerabilities before exploitation.

B. *Common Deep Learning Algorithms in Cybersecurity*

1. *Convolutional Neural Networks (CNNs)*

CNNs are highly effective in analyzing structured and unstructured data, making them suitable for various cybersecurity applications:

- **Network Packet Analysis**: CNNs can process and analyze packet-level data to detect anomalies or identify malicious traffic patterns. For example, by treating packet headers or payloads as image-like data, CNNs classify and detect threats like malware embedded in network traffic.
- **Malware Detection**: CNNs can analyze binary representations of malware files (converted into images) to differentiate between malicious and benign files.

2. *Recurrent Neural Networks (RNNs)*

RNNs are tailored for sequential data, which is common in cybersecurity tasks:

- **User Activity Monitoring:** RNNs can analyze sequences of user actions to detect unusual behavior that may indicate unauthorized access or insider threats.
- **Network Traffic Analysis:** They are well-suited for time-series data, such as identifying patterns of distributed denial-of-service (DDoS) attacks or detecting anomalies in network flow.

3. *Autoencoders*

Autoencoders are unsupervised learning models designed to reconstruct input data. In cybersecurity, they are used for:

- **Anomaly Detection:** Autoencoders learn the normal patterns of system or network behavior. Any deviation in reconstruction error can indicate anomalies or potential threats.
- **Vulnerability Detection:** They can identify hidden weaknesses in network configurations or software systems by detecting unusual patterns in operational data.

4. *Generative Adversarial Networks (GANs)*

GANs consist of two neural networks (generator and discriminator) working in opposition to create and evaluate data. Their applications include:

- **Cyberattack Simulation:** GANs simulate advanced attack scenarios, such as zero-day vulnerabilities, to test and improve defensive mechanisms.
- **Adversarial Training**: By generating adversarial examples, GANs can help enhance the robustness of machine learning models against attacks like adversarial inputs.

5. *Transformer Models*

Transformers, known for their attention mechanisms, excel in processing complex sequential and textual data:

- **Log Analysis**: Transformers analyze system logs for hidden patterns that indicate potential breaches or vulnerabilities.
- **Intrusion Detection**: They classify and analyze textual data (e.g., emails, phishing messages) to detect malicious activities with high precision.

Table 10. Comparison: Deep Learning vs. Traditional Machine Learning

Feature	Traditional Machine Learning	Deep Learning
Feature Engineering	Requires manual design	Automates feature extraction
Handling Big Data	Limited to smaller datasets	Excels with large and complex data
Training Time	Faster	More time-intensive
Adaptability	Constrained to known patterns	Detects unknown attack patterns

C. *Challenges of Using Deep Learning in Cybersecurity*

1. *Data Requirements*

- **Challenge**: Deep learning models need vast amounts of labeled data for effective training, which can be scarce in cybersecurity due to the sensitive and diverse nature of cyberattack data.
- **Solution**: Semi-supervised learning or data augmentation techniques (e.g., using GANs) can help mitigate this issue.

2. *High Computational Costs*

- **Challenge**: Training deep learning models requires significant computational resources, including GPUs or TPUs, which may not be readily available.
- **Solution**: Leveraging cloud-based platforms and optimizing model architectures can reduce computational overhead.

3. *Interpretability Issues*

- **Challenge**: Deep learning models, often described as "black boxes," provide limited insight into why a particular decision or prediction was made. This is critical in

cybersecurity, where understanding attack vectors is essential.

- **Solution**: Techniques like SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations) can enhance interpretability.

4. Adversarial Vulnerabilities

- **Challenge**: Deep learning models are susceptible to adversarial attacks, where small, crafted perturbations in input data can lead to incorrect predictions.
- **Solution**: Implementing robust adversarial training and detection mechanisms can mitigate this risk.

5. Model Generalization

- **Challenge**: Models trained on specific datasets may fail to generalize to new attack types or datasets.
- **Solution**: Continual learning approaches and diverse training datasets can improve adaptabilit.

IX. CONCLUSION

The evaluation of traditional machine learning algorithms across seven different types of cyberattacks—DDoS, Phishing, Malware, Ransomware, SQL Injection, Zero-Day Attacks, and MITM—demonstrates the adaptability and effectiveness of these methods in addressing cybersecurity challenges. Across all cases, ensemble methods such as Random Forest and Gradient Boosting consistently performed well, offering high accuracy and detection rates. These models excel in handling high-dimensional data, imbalanced datasets, and complex attack patterns, making them reliable choices for various cybersecurity applications.

Algorithms like Support Vector Machine (SVM) showed exceptional performance in scenarios requiring precise detection, such as Zero-Day and MITM attacks, due to their robustness in distinguishing subtle anomalies. However, simpler models like Naive Bayes and K-Nearest Neighbors exhibited limited performance, often struggling with noisy data or high-dimensional feature spaces, especially in sophisticated attack scenarios like Ransomware or SQL Injection. While Logistic Regression and Decision Tree provided moderate performance, their effectiveness was hindered in large-scale, dynamic environments where data diversity is significant.

X. FUTURE WORKS

Based on the findings from the various attack types, future research could focus on the following key directions to enhance traditional machine learning algorithms for cyberattack detection:

- Integration with Deep Learning: As suggested by Wu et al. (2022) for Zero-Day and Zhang & Li (2021) for MITM detection, combining Support Vector Machine (SVM) and Random Forest with deep learning models can significantly improve detection accuracy, especially for complex and evolving attacks. These hybrid models can better capture intricate patterns and anomalies in high-dimensional and encrypted traffic.
- Real-time Anomaly Detection: For attacks such as DDoS and MITM, integrating algorithms like Random Forest and SVM with real-time anomaly detection systems could enable faster identification of new attack patterns, as proposed by Singh & Kumar (2022) for DDoS. This can improve response times and reduce the impact of attacks in dynamic network environments.
- Feature Optimization and Selection: Researchers like Yang et al. (2021) for SQL Injection and Gupta et al. (2022) for MITM highlight the importance of optimizing feature extraction and selection methods. This can help improve the performance of simpler models like Naive Bayes and K-Nearest Neighbors, which often struggle with high-dimensional datasets. Reducing the number of irrelevant features while preserving key information can make these algorithms more effective in detecting sophisticated attacks.
- Ensemble Methods: Combining traditional models through ensemble methods could improve the overall detection capabilities, particularly for Naive Bayes and K-Nearest Neighbors (Gupta et al., 2022). By leveraging the strengths of different algorithms, ensemble techniques can mitigate the weaknesses of individual models, offering higher accuracy and robustness.
- Scalability and Computational Efficiency: Models like Random Forest and Gradient Boosting are effective but can be computationally expensive. Researchers like Zhao & Chen (2021) for MITM suggest incorporating more scalable and efficient variants, such as lightweight Random Forest or distributed boosting methods, to make these algorithms more applicable to large-scale datasets and real-time systems.
- By addressing these challenges, future work can further enhance the ability of traditional machine learning algorithms to detect a wide range of cyberattacks with greater accuracy, speed, and scalability.

REFERENCES:

[1] *Li S, Da Xu L, Zhao S. The internet of things: a survey. Inform Syst Front. 2015;17(2):243–59*

[2] *M. Abomhara and G. M. Køien, ''Cyber security and the Internet of Things: Vulnerabilities, threats, intruders and attacks,'' J. Cyber Secur. Mobility, vol. 4, no. 1, pp. 65–88, 2015*

[3] *Parkar P, Bilimoria A. A survey on cyber security IDS using ML methods. Proceedings—5th International Conference on Intelligent Computing and Control Systems, ICICCS 2021, no. ICICCS, pp. 352–360, 2021, https://doi.org/10.1109/ICICCS51141.2021.9432210*

[4] *Sun N, Zhang J, Rimba P, Gao S, Zhang LY, Xiang Y. Data-driven cybersecurity incident prediction: a survey. IEEE Commun Surv Tutor. 2018;21(2):1744–72*

[5] *McIntosh T, Jang-Jaccard J, Watters P, Susnjak T. The inadequacy of entropy-based ransomware detection. In:International conference on neural information processing. New York: Springer; 2019. p. 181–189*

[6] *P. Wang and C. Johnson, ''Cybersecurity incident handling: A case study of the equifax data breach,'' Issues Inf. Syst., vol. 19, no. 3, pp. 1–10, 2018.*

[7] *Perwej Y, Qamar Abbas S, Pratap Dixit J, Akhtar N, Kumar Jaiswal A. A systematic literature review on the cybersecurity. Int J Sci Res Manag. 2021;9(12):669710.https://doi.org/10.18535/ijsrm/v9i12.ec04.*

[8] *B. Kaur, S. Dadkhah, F. Shoeleh, E.C.P. Neto, P. Xiong, S. Iqbal, P. Lamontagne, S. Ray, A.A. Ghorbani, Internet of Things (Iot) Security Dataset Evolution: Challenges and Future Directions, Internet of Things, 2023 100780*

[9] *Ibm security report, https://www.ibm.com/security/data-breach. Accessed on 20 Oct 2019*

[10] *D. Zhang, X. Han, and C. Deng, ''Review on the research and practice of deep learning and reinforcement learning in smart grids,'' CSEE J. Power Energy Syst., vol. 4, no. 3, pp. 362–370, Sep. 2018*

[11] *K. Dushyant, G. Muskan, A. Gupta, and S. Pramanik, ''Utilizing machine learning and deep learning in cybesecurity: An innovative approach,'' in Cyber Security and Digital Forensics. Wiley, 2022, pp. 271–293. [Online]. Available: https://doi.org/10.1002/9781119795667.ch12*

[12] *C. Gupta, I. Johri, K. Srinivasan, Y.-C. Hu, S. M. Qaisar, and K.-Y. Huang, ''A systematic review on machine learning and deep learning models for electronic information security in mobile networks,'' Sensors, vol. 22, no. 5, p. 2017, Mar. 2022*

[13] *Aftergood S. Cybersecurity: the cold war online. Nature. 2017;547(7661):30*

[14] *D'Angelo, G., & Palmieri, F. (2021). Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction. Journal of Network and Computer Applications, 173(NA), 102890-NA. https://doi.org/10.1016/j.jnca.2020.102890*

[15] *Srivastava, A. K., Morris, T., Ernster, T. A., Vellaithurai, C. B., Pan, S., & Adhikari, U. (2013). Modeling CyberPhysical Vulnerability of the Smart Grid With Incomplete Information. IEEE Transactions on Smart Grid, 4(1), 235-244. https://doi.org/10.1109/tsg.2012.2232318*

[16] *Zhang, F., Kodituwakku, H. A. D. E., Hines, J. W., & Coble, J. B. (2019). Multilayer Data-Driven Cyber-Attack Detection System for Industrial Control Systems Based on Network, System, and Process Data. IEEE Transactions on Industrial Informatics, 15(7), 4362-*

[17] *Van Efferen L, Ali-Eldin AM. A multi-layer perceptron approach for flow-based anomaly detection. In: 2017 International symposium on networks, computers and communications (ISNCC). IEEE; 2017. p. 1–6*

[18] *Liu H, Lang B, Liu M, Yan H. Cnn and rnn based payload classification methods for attack detection. Knowl Based Syst. 2019;163:332–41*

[19] *Yin C, Zhu Y, Fei J, He X. A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access. 2017;5:21954–61*

[20] *Rainie L, Anderson J, Connolly J. Cyber attacks likely to increase. Digital Life in. 2014, vol. 2025.*

[21] *Fischer EA. Cybersecurity issues and challenges: In brief. Congressional Research Service (2014)*

[22] *Craigen D, Diakun-Thibault N, Purse R. Defining cybersecurity. Technology Innovation. Manag Rev 2014;4(10):13–21.*

[23] *Zhang, J., 2021. Distributed network security framework of energy internet based on Internet of Things. Sustain. Energy Technol. Assess. 44, 101051*

[24] *Alkatheiri, M.S., Chauhdary, S.H., Alqarni, M.A., 2021. Seamless security apprise method for improving the reliability of sustainable energy-based smart home applications. Sustain. Energy Technol. Assess. 45, 101219*

[25] *Ogbanufe, O., 2021. Enhancing end-user roles in information security: Exploring the setting, situation, and identity. Comput. Secur. 108, 102340*

[26] *Krishnasamy, V., Venkatachalam, S., 2021. An efficient data flow material model based cloud authentication data security and reduce a cloud storage cost using index-level Boundary Pattern Convergent Encryption algorithm. Mater.Today: Proc*

[27] *C. Tankard, Advanced persistent threats and how to monitor and deter them, Netw. Secur. 2011 (8) (2011) 16–19, doi:10.1016/S1353-4858(11)70086-1.*

[28] *R. Sharma, Study of latest emerging trends on cyber security and its challenges to society, Int. J. Sci. Eng. Res. 3 (6) (2012) 1–4.*

[29] *Eswaran M, et al. Survey of cyber security approaches for attack detection and prevention. IEEE Access. 2023;12(1):1–6. https://doi.org/10.17762/turcomat.v12i2.2406*

[30] *Uma M, Padmavathi G. A survey on various cyber attacks and their classification. Int J Netw Secur. 2013;15(5):390–6. https://doi.org/10.6633/IJNS.201309*

[31] *Rauf U, Mohsen F, Wei Z. A taxonomic classification of insider threats: existing techniques, future directions and recommendations. J Cyber Secur Mobil. 2023;12(2):221–52. https://doi.org/10.13052/jcsm2245-1439.1225.*

[32] *Thomas T, Vijayaraghavan AP, Emman*

[33] *O. Yavanoglu and M. Aydos, ''A review on cyber security datasets for machine learning algorithms,'' in Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2017, pp. 2186–2193.*

[34] *T. Thomas, A. P. Vijayaraghavan, and S. Emmanuel, Machine Learning Approaches in Cyber Security Analytics. Cham, Switzerland: Springer, 2020.*

[35] *I. H. Sarker, ''Deep cybersecurity: A comprehensive overview from neural network and deep learning perspective,'' Social Netw. Comput. Sci., vol. 2, no. 3, p. 154, May 2021.*

[36] *Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, ''Machine learning and deep learning methods for cybersecurity,'' IEEE Access, vol. 6, pp. 35365–35381, 2018.*

[37] *A. L. Buczak and E. Guven, ''A survey of data mining and machine learning methods for cyber security intrusion detection,'' IEEE Commun. Surveys Tuts., vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016.*

[38] *Alduailij M, Khan QW, Tahir M, Sardaraz M, Alduailij M, Malik F. Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method. Symmetry (Basel). 2022;14(6):1–15 https://doi.org/10.3390/sym14061095.*

[39] *Gawand MKSP. A comparative study of cyber attack detection & prediction using machine learning algorithms. Researchgate. 2013. https://doi.org/10.21203/rs.3.rs-3238552/v1*

[40] *Sarker IH. CyberLearning: effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet Things. 2021;14:100393. https://doi.org/10.1016/j.iot.2021.100393.*

[41] *Hasan M, Islam MM, Zarif MII, Hashem MMA. Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. Internet Things. 2019;7:100059. https://doi.org/10.1016/j.iot.2019.100059.*

[42] *Y. LeCun, Y. Bengio, and G. Hinton, ''Deep learning,'' Nature, vol. 521, pp. 436–444, May 2015*

[43] *G. E. Hinton, ''Deep belief networks,'' Scholarpedia, vol. 4, no. 5, p. 5947,2009.*

[44] *Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, ''Gradient-based learning applied to document recognition,'' Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.*

[45] *Berman DS, Buczak AL, Chavis JS, Corbett CL. A survey of deep learning methods for cyber security. Information. 2019;10(4):122.*

[46] *Alsulaiman, L., & Al-Ahmadi, S. (2021). Performance Evaluation of Machine Learning Techniques for DoS Detection in Wireless Sensor Network. arXiv.*

[47] *Alkhonaini, R.A., E.S., C.S., F.B., & A.V. (2024). Enhancing IoT Security in Vehicles: A Comprehensive Review of AI-Driven Solutions for Cyber-Threat Detection. MDPI.*

[48] *Othman, S.M., Ba-Alwi, F.M., & Alsohybe, N.T. (2018). Intrusion Detection Using SVM and Network Traffic Analysis. Journal of Big Data.*

[49] *Kuo, C.-W., & Wu, C.-X. (2025). A Lightweight Machine Learning Framework for IoT Security: KNN and Ensemble Methods. IEEE / SSRN Preprint.*

[50] *Y Zhang, H Gao, X Ji, Q Liu, Y Yu. Robust Integrated Optimization of Active Distribution Network Based on System Risk Index.*

[51] *H Liu, B Lang. Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey*

[52] *ISACA. (2024). Securing the Future: Enhancing Cybersecurity in 2024.*

[53] *Forbes. (2024). The Role of AI in Cybersecurity: Trends and Applications in 2024.*

[54] *Deloitte. (2024). 2024 Global Cybersecurity Outlook.*

[55] *MIT Technology Review. (2024). AI in Cybersecurity: Balancing Defense and Threats.*

[56] *Gartner. (2024). The Role of AI in Cybersecurity Threat Detection.*

[57] *Sommer, R., & Paxson, V. (2010). Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. Proceedings of the IEEE Symposium on Security and Privacy, 305-316.*

[58] *Wang, S., et al. (2020). A Comprehensive Survey on Machine Learning for Cybersecurity. IEEE Access, 8, 181607-181630.*

[59] *Manning, C. D., et al. (2008). Introduction to Information Retrieval. Cambridge University Press.*

[60] *Zhang, Z. & Liu, J., 2023. Random Forest Classifier for DDoS Detection: A Systematic Review. Springer Journal of Information Security, Springer.*

[61] *Lee, D. & Choi, T., 2022. Performance Evaluation of Gradient Boosting in DDoS Detection. Springer Journal of Cybersecurity, Springer.*

[62] *Martin, S. et al., 2022. Decision Tree Classifiers for DDoS Detection: An Evaluation. Elsevier Networks Journal, Elsevier.*

[63] *Liu, J. et al., 2023. Support Vector Machine Classifiers for DDoS Detection. MDPI Journal of Cybersecurity, MDPI.*

[64] *Kim, S. & Park, J., 2022. KNN-Based DDoS Detection System. Springer Journal of Information Security, Springer.*

[65] *Cho, H., 2021. Logistic Regression Models for DDoS Attack Detection. IEEE Transactions on Dependable and Secure Computing, IEEE*

[66] *Lee, C. et al., 2021. Naive Bayes Classifier for DDoS Attack Detection. MDPI Journal of Information Security, MDPI.*

[67] *Soni, P. & Mishra, A., 2021. Random Forest for Phishing Detection: A Comprehensive Review. International Journal of Computer Applications.*

[68] *Kumar, S. et al., 2020. Gradient Boosting for Phishing Detection in Web Applications. Journal of Cybersecurity and Privacy MDPI.*

[69] *Patel, M. et al., 2021. Decision Tree Classifier for Phishing Detection. Elsevier Journal of Information Security.*

[70] *Wang, H. et al., 2021. Support Vector Machines for Phishing Website Detection. Journal of Network and Computer Applications Elsevier.*

[71] *Zhang, X. & Li, Y., 2020. K-Nearest Neighbors for Phishing Detection. Springer Journal of Computer Science.*

[72] *Singh, R. & Ghosh, M., 2021. Logistic Regression-Based Phishing Detection Model. International Journal of Artificial Intelligence & Machine Learning.*

[73] *Kumar, A. et al., 2020. Naive Bayes for Phishing Attack Detection: A Review. Journal of Information Security Springer,*

[74] *Sun, Z. & Wang, L., 2022. Random Forest for Malware Detection: A Comprehensive Approach. Journal of Information Security, Springer.*

[75] *Li, J. & Zhang, Q., 2021. Gradient Boosting for Malware Detection in Network Traffic. Elsevier Journal of Cybersecurity, Elsevier.*

[76] *Zhang, X. et al., 2021. Decision Tree for Malware Detection: A Survey and Performance Evaluation. Elsevier Journal of Information Security, Elsevier.*

[77] *Liu, J. et al., 2022. Support Vector Machine for Malware Detection: An Efficient Approach. MDPI Journal of Cybersecurity, MDPI.*

[78] *Lee, S. et al., 2021. K-Nearest Neighbors for Malware Classification: Application and Analysis. Springer Journal of Computer Science, Springer.*

[79] *Chen, L. & Wang, H., 2021. Logistic Regression-Based Malware Detection: A Novel Approach. Springer Journal of Artificial Intelligence, Springer.*

[80] *Wang, X. et al., 2021. Naive Bayes for Malware Detection in Large-Scale Datasets. MDPI Journal of Information Security, MDPI*

[81] *Zhang, L. & Wang, F., 2021. Random Forest for Ransomware Attack Detection: A Comparative Study. Springer Journal of Cybersecurity, Springer.*

[82] *Li, X. et al., 2020. Gradient Boosting for Ransomware Detection in Network Traffic. Elsevier Journal of Information Security, Elsevier.*

[83] *Kumar, S. & Patel, A., 2021. Decision Tree for Ransomware Attack Detection in Web Applications. MDPI Journal of Cybersecurity, MDPI.*

[84] *Yadav, R. et al., 2021. Support Vector Machine for Ransomware Detection: A Performance Evaluation. Springer Journal of Computer Science, Springer.*

[85] *Liu, J. & Zhang, H., 2022. K-Nearest Neighbors for Ransomware Detection in Small Datasets. Elsevier Journal of Information Security, Elsevier.*

[86] *Zhao, Y. & Wang, Z., 2022. Logistic Regression for Ransomware Detection in IoT Systems. MDPI Journal of Artificial Intelligence, MDPI.*

[87] *Zhang, T. et al., 2021. Naive Bayes for Ransomware Detection: A Case Study. Springer Journal of Cybersecurity, Springer.*

[88] *Gupta, R. & Sharma, A., 2021. Random Forest for SQL Injection Attack Detection: A Performance Evaluation. Springer Journal of Cybersecurity, Springer.*

[89] *Kumar, P. et al., 2020. Gradient Boosting for SQL Injection Attack Detection. Elsevier Journal of Information Security, Elsevier*

[90] *Lee, J. et al., 2021. Decision Tree Classifier for SQL Injection Attack Detection. MDPI Journal of Cybersecurity, MDPI.*

[91] *Zhao, X. et al., 2022. Support Vector Machine for SQL Injection Detection: A Comparative Analysis. Springer Journal of Computer Science, Springer.*

[92] *Zhang, H. et al., 2021. K-Nearest Neighbors for SQL Injection Detection in Web Applications. Elsevier Journal of Information Security, Elsevier.*

[93] *Chen, L. & Wang, Y., 2021. Logistic Regression for SQL Injection Detection in Database Systems. MDPI Journal of Artificial Intelligence, MDPI.*

[94] *Yadav, A. & Singh, S., 2020. Naive Bayes for SQL Injection Detection: A Review. Springer Journal of Cybersecurity, Springer.*

[95] *Singh, R. & Kumar, A., 2022. Random Forest for Zero-Day Attack Detection: A Survey. Journal of Information Security, Springer.*

[96] *Zhang, Y. et al., 2021. Gradient Boosting for Zero-Day Attack Detection: An Evaluation. Elsevier Journal of Cybersecurity, Elsevier.*

[97] *Lee, H. & Cho, K., 2021. Decision Tree Classifier for Zero-Day Attack Detection. MDPI Journal of Cybersecurity, MDPI.*

[98] *Wu, Y. et al., 2022. Support Vector Machine for Zero-Day Attack Detection in Network Traffic. Springer Journal of Computer Science, Springer.*

[99] *Yang, X. et al., 2021. K-Nearest Neighbors for Zero-Day Attack Detection in Large Datasets. Elsevier Journal of Information Security, Elsevier.*

[100] *Zhao, X. & Zhang, L., 2021. Logistic Regression for Zero-Day Attack Detection: A Comprehensive Review. MDPI Journal of Artificial Intelligence, MDPI.*

[101] *Xu, F. et al., 2022. Naive Bayes for Zero-Day Attack Detection: A Comparative Study. Springer Journal of Cybersecurity, Springer.*

[102] *Zhang, W. & Li, Y., 2021. Random Forest for Man-in-the-Middle Attack Detection: A Comparative Study. Springer Journal of Cybersecurity, Springer.*

[103] *Yang, H. et al., 2021. Gradient Boosting for Man-in-the-Middle Attack Detection. Elsevier Journal of Information Security, Elsevier.*

[104] *Singh, M. & Choudhary, N., 2021. Decision Tree for Man-in-the-Middle Attack Detection. MDPI Journal of Cybersecurity, MDPI.*

[105] *Wang, Z. et al., 2022. Support Vector Machine for MITM Attack Detection in Network Traffic. Springer Journal of Computer Science, Springer.*

[106] *Lee, T. et al., 2021. K-Nearest Neighbors for MITM Attack Detection. Elsevier Journal of Information Security, Elsevier.*

[107] *Zhao, L. & Chen, F., 2021. Logistic Regression for Man-in-the-Middle Attack Detection in Web Applications. MDPI Journal of Artificial Intelligence, MDPI.*

[108] *Gupta, S. et al., 2022. Naive Bayes for Man-in-the-Middle Attack Detection: A Performance Evaluation. Springer Journal of Cybersecurity, Springer.*

How to cite: Hanieh Khosravi, Elham Fereydoonifard, Zoleikha Jahanbakhsh Naghadeh, **Analysis of machine learning algorithms towards cyberattacks detection: a survey** , Journal of Distributed Computing and Systems (JDCS), Vol 8, Issue 1, Pages 54-70, 2025.