Enhancing Cybersecurity with Machine Learning: Development and Evaluation of Intrusion Detection Systems

Ogundipe Ademola Oluwatobi [1], Waheed Azeez Ajani[2], Adedokun Okikiade Salim[3], Akinmuda Oluseye Ayobami[4], Odeajo Israel[5]

[1,2,3]Department of Computer Science, LeadCity University, Ibadan, Oyo State, Nigeria
[4]AI Engineer, Join Momentum, USA

doi: https://doi.org/10.37745/ejcsit.2013/vol13n51138173 Published September 10, 2025

Citation: Oluwatobi O.A., Ajani W.A., Salim A.O. Ayobami A.O., Israel O. (2025) Enhancing Cybersecurity with Machine Learning: Development and Evaluation of Intrusion Detection Systems, *European Journal of Computer Science and Information Technology*, 13(51),138-173

Abstract: *The widespread adoption of digital networks and information systems has transformed modern society, but it has also led to a surge in sophisticated cyber threats such as malware, phishing, denial-of-service (DoS) attacks, ransomware, and advanced persistent threats (APTs). Traditional rule-based security systems are increasingly ineffective against these evolving threats, often failing to detect novel attack patterns, leading to false positives, missed detections, and delayed responses. This study aimed to address these challenges by applying machine learning algorithms to improve the accuracy and efficiency of cyber-attack detection. Using the UNSW-NB15 dataset, which contains 175,341 training and 82,332 testing records representing both benign and malicious network traffic with 49 relevant features, the research applied synthetic minority over-sampling technique (SMOTE) to balance the dataset and principal component analysis (PCA) to reduce feature dimensionality by retaining up to 95% of data variance. Five machine learning models Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), Decision Tree, and Random Forest were trained and evaluated using metrics such as accuracy, precision, recall, and F1 score.The results demonstrated that KNN achieved the highest accuracy of 94.69%, with balanced precision (95.31%), recall (93.96%), and F1 score (94.63%), showing robust classification of both attack and non-attack instances. Random Forest and ANN also showed strong performances with accuracies of 92.81% and 95%, respectively, highlighting their effectiveness in handling complex cybersecurity data. SVM and Decision Tree had slightly lower accuracies of 90.88% and 92.22%. These findings confirm the value of machine learning, especially KNN and ensemble methods, for real-world intrusion detection. Regular model retraining is essential to address emerging attack patterns and maintain effective cybersecurity defenses.*

Keywords: cyber security, cyber threats, feature selection, intrusion detection, machine learning

INTRODUCTION

The adoption of digital networks and information systems has transformed every facet of modern society, providing numerous benefits to businesses, governments, and individuals. However, with this widespread digitalization comes the rise of various cyber threats. These threats have evolved over the years, becoming increasingly sophisticated and complex. The growing frequency and severity of cyber-attacks such as malware, phishing, denial-of-service (DoS) attacks, ransomware, and advanced persistent threats (APTs) have highlighted the vulnerabilities in network infrastructures and information systems (Aslan et al., 2023; Mallick & Nath, 2024). These attacks not only lead to significant financial losses but also cause data breaches, loss of customer trust, and damage to the reputation of organizations (Kumar, 2023). The consequences of cyber-attacks are severe, affecting not only the targeted organizations but also individuals whose personal information may be compromised. As cyber attackers continue to develop innovative methods to bypass traditional security measures, the need for advanced and more dynamic cybersecurity solutions has become critical. In response to this growing concern, cybersecurity measures have undergone significant transformations in an attempt to address the evolving nature of cyber threats. Traditional rule-based detection systems have become increasingly ineffective as attackers continue to devise new techniques to circumvent static security measures. This inadequacy has driven the search for more proactive and intelligent solutions to improve the detection and prevention of cyber-attacks (Kumar, 2023). The rapid advancements in artificial intelligence (AI) and machine learning (ML) technologies have provided new avenues for combating these persistent cyber threats. Machine learning has gained substantial attention in the cybersecurity domain due to its ability to learn from data, make predictions, and adapt to evolving attack patterns (Yu et al., 2024) Unlike traditional systems that rely on pre-defined rules and signatures, machine learning algorithms are capable of analyzing large volumes of network traffic and identifying anomalies that may indicate a cyber-attack (Manoharan & Sarker, 2023). This ability to recognize previously unknown threats makes ML an attractive solution for enhancing the security of digital infrastructures.

Machine learning has emerged as a powerful tool in safeguarding digital environments. By leveraging the power of data, ML algorithms can learn from historical patterns, predict potential threats, and dynamically respond to new attack vectors (Manoharan & Sarker, 2023). This proactive approach significantly improves the accuracy and efficiency of intrusion detection systems (IDS), which are integral components of modern cybersecurity infrastructures (Yu et al., 2024). An intrusion detection system monitors network traffic for signs of malicious activity and generates alerts when suspicious patterns are detected. However, the increasing complexity of attacks necessitates the adoption of more sophisticated systems capable of evolving with the changing threat landscape. Machine learning provides such a solution by enabling IDS to continuously improve and adapt to new types of attacks without requiring manual intervention or constant rule updates (Alam et al., 2024). Research has shown that the application of machine learning in intrusion detection systems has significantly improved their performance compared to traditional methods. Studies have demonstrated that ML-based IDS are capable of achieving higher detection rates and lower false positive rates, which are key indicators of the effectiveness of an intrusion detection system. By using algorithms such as decision trees, support vector machines (SVM), and deep learning, ML-based systems can classify network traffic and identify malicious activity with a high degree of accuracy (Sowmya & Anita, 2023; Al Farsi et al., 2024). These techniques allow for more precise detection of cyber threats, reducing the number of false alarms that can overwhelm security teams and detract from the effectiveness of the system.

Statement of Problem

As cyber-attacks become more frequent, sophisticated, and diverse, these conventional detection systems struggle to keep pace with emerging threats. The current rule-based systems often fail to identify new or evolving cyber threats effectively, leading to inefficiencies in defense mechanisms and wasted resources. These systems typically rely on pre-defined rules or signatures, which makes them ill-equipped to detect novel attack patterns, especially those that are not yet known or classified. Also, previous studies such as Manjramkar & Jondhale et al. (2023), Marengo et al. (2024) presented the attack detection using machine learning algorithms with limited accuracy. This study tends to and improve on the accuracy of related studies. By applying machine learning algorithms. By leveraging machine learning, this research aims to provide more accurate and efficient detection of cyber-attacks, helping to overcome the issues of false positives, missed detections, and delayed responses that hinder the effectiveness of conventional systems.

Aim and Objectives

The aim of this project is to improve cybersecurity by using machine learning algorithms to detect cyber-attacks. The specific objectives are to:

i. collect and preprocess network traffic data from various sources, ensuring that the data is representative of real-world scenarios and contains examples of both normal behavior and cyber-attacks.

ii. identify relevant features in the network traffic data that are indicative of potential cyber-attacks, including factors such as packet size, duration, and communication patterns.

iii. develop and implement multiple machines learning models, including decision trees, support vector machines (SVM), XGBoost, random forests, and neural networks, to identify and classify potential cyber-attacks.

iv. compare the performance of these algorithms in terms of accuracy, precision, recall, and computational efficiency to determine the most effective approach for cyber-attack detection.

LITERATURE REVIEW

Cyber Security

The rapid proliferation of digital technologies over recent decades has fundamentally reshaped modern society, transforming how individuals communicate, businesses operate, and governments deliver services. According to report more than 5.3 billion people over half of the global population are active internet users, reflecting the unprecedented scale and ubiquity of digital connectivity worldwide (Lawelai et al., 2025). This widespread adoption of digital platforms and services has unlocked immense opportunities for economic growth, social inclusion, innovation, and global collaboration. However, this digital expansion also introduces a complex and rapidly evolving landscape of cyber security challenges that threaten the stability and safety of digital ecosystems (Lawelai et al., 2025). The integration of digital technologies into virtually every aspect of daily life and organizational operations has created an expanded attack surface for malicious actors. Cyber-attacks have evolved from isolated incidents targeting individual computers to sophisticated, large-scale campaigns capable of disrupting critical infrastructure, compromising sensitive data, and undermining public trust (Ţălu , 2025).

Evolving Cyber Security Threat

Ransomware and Malware

Ransomware remains one of the most pervasive and destructive cyber threats in recent times, representing a critical challenge for organizations across all sectors. This form of malicious software operates by infiltrating victim systems, encrypting valuable data, and then demanding a ransom payment usually in cryptocurrency in exchange for the decryption keys necessary to restore access (Styles, 2025). The financial and operational impact of ransomware attacks can be devastating, often leading to prolonged downtime, loss of sensitive information, and significant reputational damage (Dobrovolska & Rozhkova, 2024). Among the most notorious ransomware groups active are LockBit and Clop (Lee et al., 2024; Tan et al., 2025). These groups have demonstrated the ability to exploit vulnerabilities in widely used enterprise applications to gain initial access and deploy ransomware payloads.. These attacks underscore the critical importance of timely patching and vulnerability management, as attackers increasingly leverage zero-day exploits and supply chain weaknesses to maximize impact. Beyond ransomware, malware in its broader sense continues to evolve, encompassing a variety of malicious software types including trojans, worms, spyware, and rootkits (Triantafyllou, 2024). Modern malware employs advanced obfuscation techniques such as polymorphism and encryption to evade detection by traditional antivirus and intrusion detection systems (Imamverdiyeva & Baghirovb, 2024). Polymorphic malware, for example, changes its code signature with each infection, making signature-based detection ineffective and forcing defenders to rely on behavior-based and heuristic analysis (Mazhar & Rohatg, 2025). The proliferation of mobile devices and Internet of Things (IoT) endpoints has expanded the attack surface for malware infections.

Mobile malware targets smartphones and tablets, often aiming to steal credentials, intercept communications, or deliver ransomware payloads. IoT devices, which range from smart home appliances to industrial control systems, frequently have limited built-in security, outdated firmware, and weak authentication mechanisms, making them susceptible to malware infections and botnet recruitment (Al Hwaitat et al., 2024).

However, the dynamic nature of ransomware and malware continues to challenge defenders, requiring continuous innovation in detection, prevention, and incident response capabilities (Mohammed et al., 2024).

Types of Malware and Ransomeware

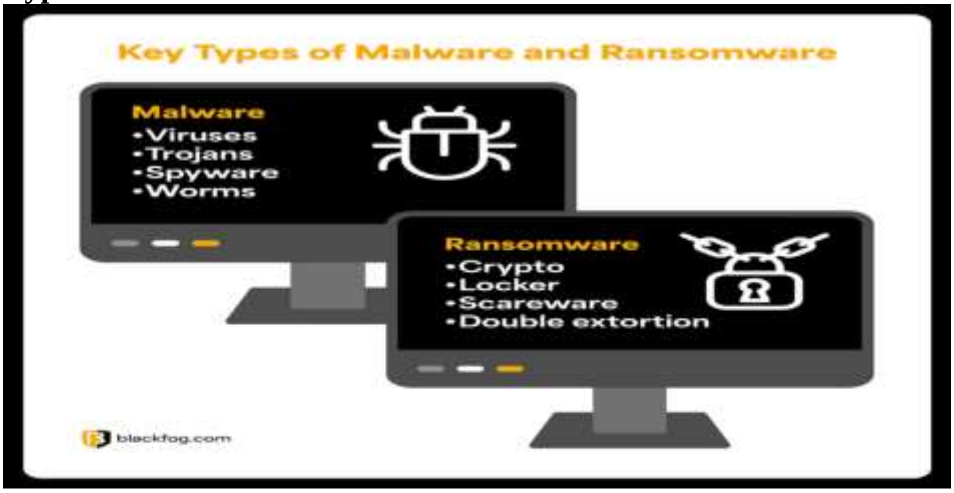


Figure 1: Types of Malware and Ransomware (blackfrog.com)

The main types of malware include:

i. **Viruses**: Viruses are malicious programs that attach themselves to legitimate files or programs. When the infected file is executed, the virus activates, replicates, and spreads to other files or systems (Kovalchuk, 2024). Viruses often corrupt data, slow down system performance, or cause system crashes.

ii. **Trojans**: Trojans (or Trojan horses) disguise themselves as legitimate software or files to trick users into installing them. Once activated, they can create backdoors for attackers, steal sensitive information, or download additional malicious payloads. Unlike viruses, trojans do not self-replicate (Ravichandran et al., 2024).

iii. **Spyware:** Spyware is designed to secretly monitor user activity and collect information, such as keystrokes, browsing habits, or login credentials, without the user's consent (Subramanian, 2025). This information is then sent to a third party, often for malicious purposes like identity theft or financial fraud.

iv. **Worms:** Worms are standalone malware that replicate themselves to spread across networks and devices without needing to attach to other files (Ansarullah et al., 2024). They often exploit vulnerabilities in network protocols, causing widespread damage and network congestion.

Table 1: **Malware and Ransomeware**

Category	**Type**	**Description**
Malware	Viruses	Attach to files/programs, replicate and spread
	Trojans	Disguise as legitimate software, create backdoors or steal data
	Spyware	Secretly monitor and collect user information
	Worms	Self-replicate and spread across networks

Ransomware	Crypto	Encrypts files, demands ransom for decryption
	Locker	Locks users out of their device
	Scareware	Uses fake warnings to trick users into paying
	Double Extortion	Encrypts and steals data, threatens to leak if ransom isn't paid

Phishing and Social Engineering

Phishing and social engineering attacks continue to be among the most prevalent and effective cyber threats (Akeiber, 2025). These attacks exploit human psychology rather than technical vulnerabilities, making them particularly difficult to defend against. As technology advances, attackers have refined their methods, leveraging sophisticated techniques such as AI-generated content and deepfakes to increase the credibility of their lures (Panda, 2025). This evolution has made phishing and social engineering increasingly challenging to detect and prevent, emphasizing the critical importance of understanding their mechanisms and varieties. Phishing is a cyber attack technique where attackers impersonate legitimate entities to deceive victims into divulging sensitive information such as usernames, passwords, credit card numbers, or installing malicious software (Ayeni et al., 2024). It is often the initial step in broader cyber attacks, including ransomware campaigns and data breaches. Phishing attacks typically use communication channels like email, instant messaging, social media, or SMS to deliver deceptive messages. The messages often contain urgent or enticing content designed to provoke an emotional response, such as fear, curiosity, or greed, compelling victims to act without due caution.

Types of Phishing Attacks

Phishing has diversified into several types, each with unique characteristics and delivery methods:

i. **Email Phishing:** The most common form of phishing, email phishing involves sending fraudulent emails that appear to come from trusted sources such as banks, government agencies, or well-known companies (Pinjarkar et al., 2024). These emails often include malicious links or attachments designed to steal credentials or install malware. Attackers use spoofed sender addresses, logos, and language mimicking legitimate communications to increase believability.

ii. **Spear Phishing:** Unlike broad email phishing campaigns, spear phishing targets specific individuals or organizations. Attackers gather detailed information about their targets such as job roles, contacts, and recent activities to craft personalized messages that are harder to detect as fraudulent. Spear phishing is often used in targeted attacks against executives or employees with access to critical systems (Bethany et al., 2024).

iii. **Whaling:** Whaling is a specialized form of spear phishing aimed at high-profile targets like CEOs, CFOs, or other senior executives. These attacks often involve highly customized messages that exploit the target's authority and access, aiming to manipulate them into authorizing fraudulent transactions or revealing sensitive corporate data (Birthriya et al., 2025).

iv. **Vishing (Voice Phishing):** Vishing involves the use of phone calls to impersonate trusted entities and manipulate victims into revealing confidential information or performing actions such as transferring funds. Attackers may use caller ID spoofing to appear legitimate and employ social engineering tactics during the conversation to build trust (Simé et al., 2024).

v. **Smishing (SMS Phishing):** Smishing uses text messages to lure victims into clicking malicious links or sharing sensitive data. These messages often mimic alerts from banks, delivery services, or government agencies, urging immediate action to avoid penalties or receive benefits (Al Saidat et al., 2024).

vi. **Clone Phishing:** In clone phishing, attackers create a near-identical replica of a legitimate email previously sent to the victim but replace links or attachments with malicious ones. Because the victim recognizes the email content, they are more likely to trust and engage with it (Al Qwaid, 2025).

vii. **Business Email Compromise (BEC):** BEC is a highly targeted phishing attack where attackers compromise or spoof a business email account to impersonate executives or trusted partners (Al Qwaid, 2025). The goal is often to trick employees into making unauthorized wire transfers or disclosing confidential information. BEC attacks have caused billions in losses globally.

Social Engineering

Social engineering encompasses a broader range of manipulative tactics that exploit human behavior to gain unauthorized access or information (Nifakos et al., 2024). It relies on psychological manipulation rather than technical exploits, making it a persistent threat regardless of technological defenses.

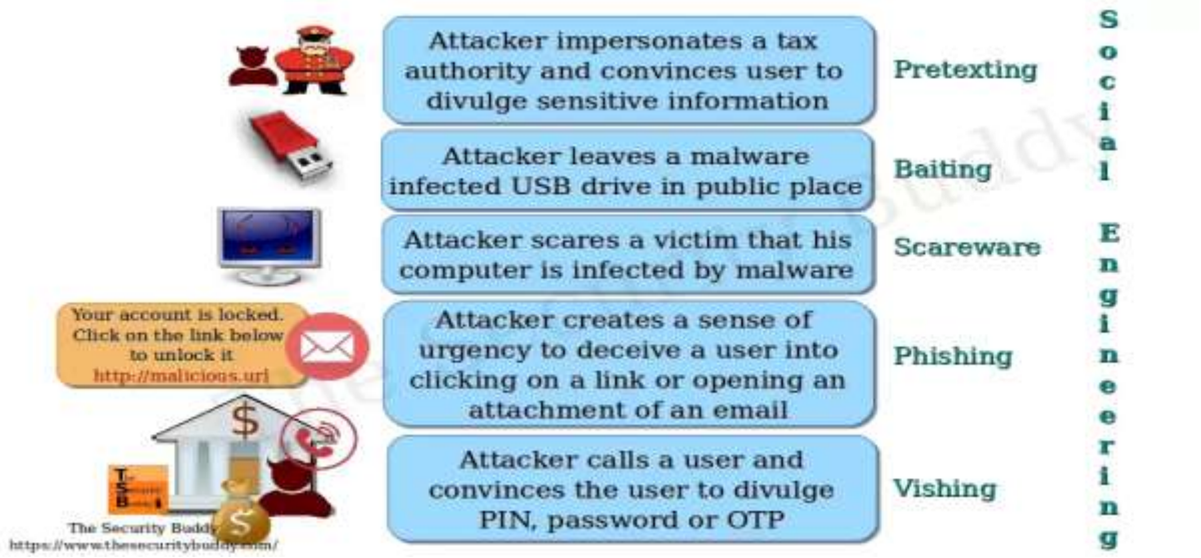

Attacker impersonates a tax authority and convinces user to divulge sensitive information
Pretexting
Attacker leaves a malware infected USB drive in public place
Baiting
Attacker scares a victim that his computer is infected by malware
Scareware
Your account is locked. Click on the link below to unlock it http://malicious.url
Attacker creates a sense of urgency to deceive a user into clicking on a link or opening an attachment of an email
Phishing
Attacker calls a user and convinces the user to divulge PIN, password or OTP
Vishing
Social Engineering
The Security Buddy
https://www.thesecuritybuddy.com/

Figure : Social Engineering (Securitybuddy.com)

Types of Social Engineering Attacks

i. **Pretexting:** Pretexting involves creating a fabricated scenario to persuade a victim to divulge information or perform actions (Gururaj et al., 2024). For example, an attacker might pose as an IT technician needing to verify user credentials or as a bank representative conducting a security check. The attacker builds trust by providing plausible details and exploiting the victim’s willingness to help.

ii. **Baiting:** Baiting uses false promises or incentives to entice victims into compromising security. This could involve leaving infected USB drives labeled “Confidential” in public areas, hoping someone will plug them into their computer, thereby installing malware. Online baiting can also include offers of free downloads or prizes that require users to provide personal information(Hawamdah, 2024).

iii. **Tailgating (or Piggybacking):** Tailgating involves physically following authorized personnel into restricted areas without proper credentials. Attackers exploit social norms such as politeness or trust to gain physical access to secure environments, which can then be used to launch cyber attacks or steal sensitive data.

iv. **Quizzes and Surveys:** Attackers may use seemingly harmless quizzes, surveys, or polls on social media or websites to collect personal information that can be used for identity theft or to craft more convincing phishing attacks. These tactics prey on curiosity and the desire for engagement.

v. **Impersonation**: Impersonation involves pretending to be someone trustworthy, such as a coworker, vendor, or authority figure, to manipulate victims into revealing information or granting access. This can be done via phone, email, or in person, often combining elements of pretexting and phishing (Barker, 2024).

Critical Infrastructure Attacks

Critical infrastructure, which includes essential systems and assets such as energy grids, healthcare networks, transportation systems, water supply, and telecommunications, forms the backbone of modern society. The uninterrupted operation of these infrastructures is vital for national security, economic stability, public health, and safety. However, attacks targeting critical infrastructure have intensified in both frequency and sophistication, leading to significant impacts and exposing systemic vulnerabilities. This escalation underscores the urgent need for modernization and robust, layered security defenses. Critical infrastructure sectors are inherently attractive targets for cyber adversaries due to their strategic importance and the potentially devastating consequences of disruption. A successful attack on these systems can result in widespread power outages, compromised patient care, transportation paralysis, or contamination of water supplies, with cascading effects throughout society and the economy. The interconnected nature of modern infrastructure means that a breach in one sector can ripple across others, amplifying the damage and complicating recovery efforts. Several high-profile cyber incidents have highlighted the vulnerabilities of critical infrastructure and the real-world consequences of cyber-attacks. In the energy sector, a coordinated cyber-attack disrupted operations at a major regional power grid in North America, causing rolling blackouts that affected millions of people (Atıcı & Tuna, 2025). The attackers exploited legacy control systems that lacked modern security features, using spear phishing to gain initial access and deploying ransomware that encrypted operational technology systems. This incident underscored the risks posed by outdated infrastructure and insufficient network segmentation between IT and OT environments. Healthcare organizations have faced a surge in ransomware and data exfiltration attacks, compromising patient data and disrupting critical services (George et al., 2024).

Machine Learning

Machine learning (ML) is a transformative field within artificial intelligence (AI) that empowers computers to learn from data and make decisions or predictions without being explicitly programmed for specific tasks. Machine learning (ML) is a subfield of artificial intelligence (AI) that focuses on developing algorithms that allow computers to learn from and make predictions or decisions based on data, without being explicitly programmed for every task (Hussain et al., 2024). Machine learning is defined as the process by which machines improve their performance on a task through experience, typically by analyzing data.

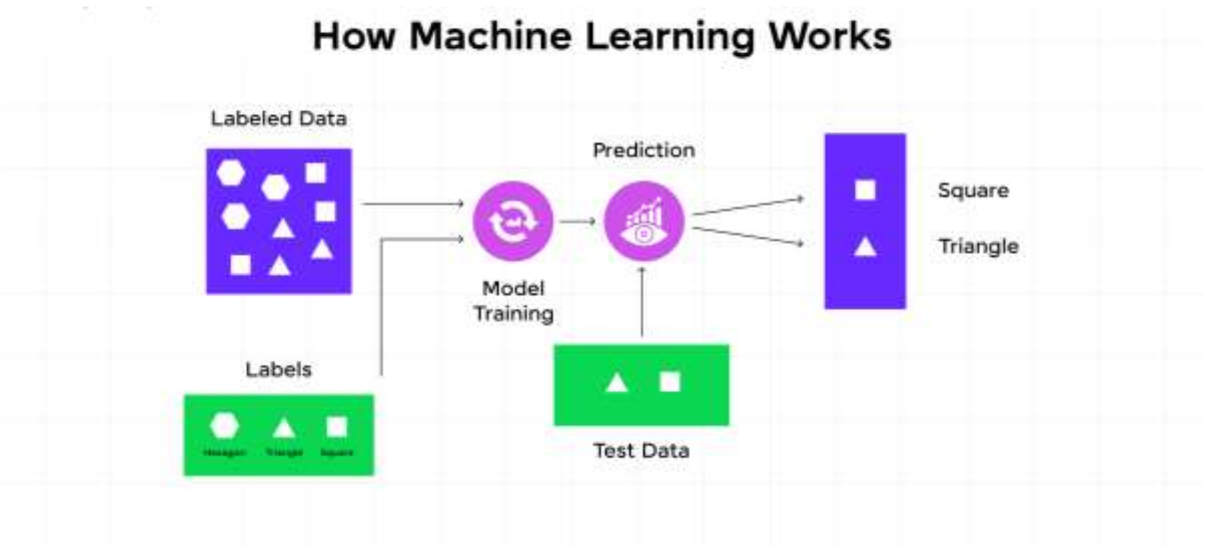


Figure 3: How Machine Learning Works.

Machine learning involves approximating an unknown function f that maps inputs X to outputs y:

y=f(X)y = f(X)y=f(X)

1

The goal is to learn f from a dataset $\{(x_i,y_i)\}^{n}_{i}=1$

such that the model generalizes well to unseen data. This is typically achieved through a systematic workflow involving data collection, preprocessing, model training, evaluation, and deployment (Monaco et al., 2024).

Types of Machine Learning

Typically, machine learning has three categories: supervised, unsupervised and reinforcement learning (Yadav, 2024).

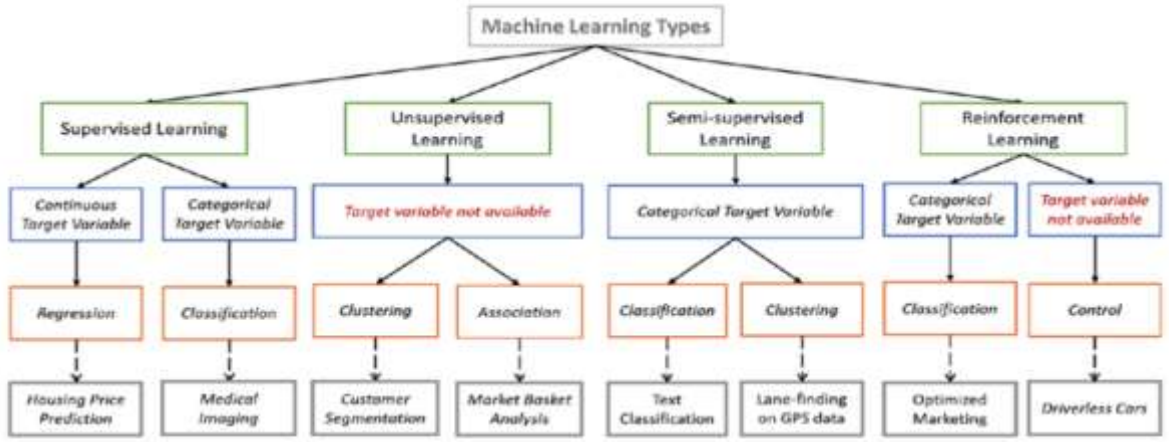


Figure 4: Types of Machine Learning (Hyunjulie, 2019)**.**

Supervised Learning

Supervised learning can be defined as a machine learning approach in which both input and output labels are provided to the model to train. The supervised model uses the input and output labeled data for training, and it extracts the patterns from the input data. These extracted patterns are used to support future judgments.

Supervised learning can be formally represented as follows:

$Y = f(x)$

2

where x represents the input variables, Y denotes an output variable and f(x) is a mapping function.

The goal is to approximate mapping function such that when an unseen input is given to the mapping function, it can predict the output variable (Y) correctly (Wang, 2024). Furthermore, supervised learning has two sub-categories: classification and regression (Vanhove et al., 2025). In a classification problem, the output variable is a category, (e.g., fraud or genuine, rainy or sunny, etc.). In a regression problem, the output variable is a real value, (e.g., the price of a house, temperature, etc.).

Unsupervised Machine Learning

Unsupervised machine learning involves training a machine using an unlabeled dataset, whereby the machine is capable of predicting output without any form of supervision (Mishra et al., 2024). The models are trained using unclassified and unlabeled data, and subsequently operate on this data in an unsupervised manner. The primary objective of the unsupervised learning algorithm is to cluster or classify the unstructured dataset based on similarities, patterns, and dissimilarities.

Clustering: The clustering methodology is employed to identify the intrinsic clusters within the dataset. Cluster analysis is a method of categorising objects into groups based on their similarities, with the aim of ensuring that objects within a group share the most similarities while having fewer or no similarities with objects in other groups (Khan et al., 2024).

Association: Association rule learning is an unsupervised machine learning methodology that aims to discover significant associations between variables in a vast dataset (Hasudungan et al., 2024).

AI and Cybersecurity

Artificial Intelligence (AI) is increasingly recognized as a transformative force in the field of cybersecurity, fundamentally reshaping how organizations detect, prevent, and respond to cyber threats (Manoharan & Sarker, 2023). As cyber-attacks grow in complexity and scale, traditional security measures reliant on human oversight and static rules have become inadequate. AI technologies, particularly machine learning (ML) and natural language processing (NLP), offer dynamic, scalable, and intelligent solutions that can analyze vast amounts of data in real time, identify subtle patterns, and adapt to emerging threats (Rajendran & Tulasi, 2025). AI in cybersecurity involves the use of computer systems capable of performing tasks that typically require human intelligence, such as pattern recognition, anomaly detection, decision-making, and language understanding. One of the most significant contributions of AI to cybersecurity is in threat detection and prevention. AI-powered security platforms continuously monitor network traffic, endpoints, cloud environments, and user behaviors to identify anomalies indicative of cyber attacks. AI systems use behavioral analytics and statistical models to detect deviations from normal patterns (Khatoon et al., 2024). AI system may flag an unusual login time or location for a user account, signaling a potential compromise. This proactive detection capability allows organizations to respond to threats before they escalate into breaches. AI also excels in malware detection, a critical area where traditional antivirus solutions often fall short (Gundoor & Mulimani, 2025). AI-driven malware detection employs static and dynamic analysis techniques, examining file attributes and runtime behavior to identify suspicious activities. By training on vast datasets of malware and benign files, machine learning models can generalize patterns and detect previously unseen threats with high accuracy. This approach significantly reduces false positives and enhances the speed and reliability of malware identification (Almomani et al., 2025).

Methodological Review

This section gives a theoretical background of the main classification algorithms used in this study.

Random Forest (RF) Algorithms

Random forest Random forest is an ensemble learning algorithm, which can be used for both regression and classification task (Patsakis et al., 2024). Random forest is a supervised ensemble method that uses a collection of numerous decision trees to make predictions. Random Forest is a classifier consisting of a set of tree-structured classifiers with identically distributed independent random vectors and each tree casting a unit vote at input x for the most popular class. Random forest utilizes bootstrapping such that each decision tree will be trained with different subsamples of data· Moreover, the random forest uses random subsets of features. For example, if there are 50 features in the data, random forest will only choose a certain number of them, let's say 10, to train on each tree. Once there is a collection of decision trees, the results of each tree will be aggregated to get the final result (vote). The model trained in such a way will ensure generalization since not one, but multiple decision trees are used for making the decision, and moreover, each tree is trained with different subsections of data.

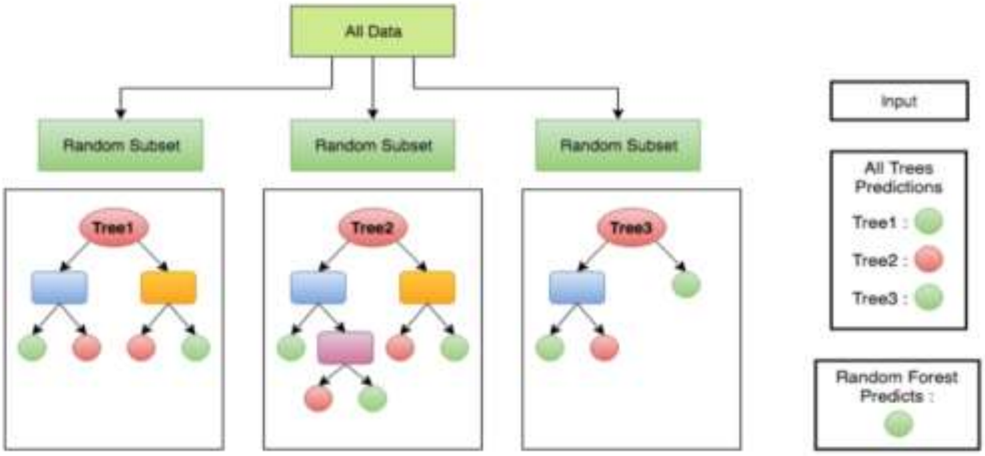


Figure 5: Random Forest Flow Chart (Oluwatoyin & Akinola, 2024).

The RF algorithm is very efficient, as it handles datasets that contain continuous variables, as well as categorical variables robustly. An RF classifier contains subsets of various tree classifiers $\{h(x,\Theta_k), k=1,2,...\}$ where the Θ_k are independently and identically distributed random vectors, with each tree being able to specify the modal class at input x. The performance index, which solely approximates the confidence interval (CI) of the RF model is given as

$$mg(x,y)=av_k I(h_k(x,\Theta_k)=y)-\max_{j\neq y} av_k I(h_k(x,\Theta_k)=j) \quad 3$$

where $I(.)$ denotes an indicator function, and $av(.)$, the average value. It is observed that as the margin increases, the confidence level also increases. The generalisation error becomes

$$PE^*=P_{x,y}(mg(x,y)<0), \quad 4$$

where $P(.)$ denotes probability. With an increase in trees for all sequences Θ_k, PE^* converges to

$$P_{x,y}(P_\Theta(h(x,\Theta)=y)-\max_{j\neq y} P_\Theta(h(x,\Theta)=j<0) \quad 5$$

Convergence of this generalisation error proves that the RF model does not overfit as more trees are introduced. The upper bound for the generalisation error is given as

$$PE^*\leq\frac{\bar{\rho}(1-s^2)}{s^2} , \quad 6$$

where $\bar{\rho}$ is the average correlation value, s is the strength of each tree in the model. An increased strength of individual trees and a low correlation between them produces more accurate prediction results.

Decision Tree

Decision Tree is a type of supervised machine learning, used for either classification or regression, also used where the data is continuously split according to a certain parameter, and to provide a graphical representation of all the possible solutions. All decisions were dependent on a number of conditions. It starts from the root node and branches off to the number of solutions, just like a tree. The tree starts from the root, then it grows branches and grows bigger and bigger. The main idea is to build a tree T from our set of observations S. if all S belongs to a class C, then the node is a leaf node and receives a label. If not, the algorithm goes to the next most informative attribute and builds sub-trees until goal is met **(**Oluwatoyin & Akinola, 2024).

$$\text{Entropy } S=-\sum P(x)\log_2 P(x) \quad 7$$

Also,the information gain that measures the relative change in entropy with respect to the independent attribute is given as:

$$\text{Gain}(S,A)=\text{Entropy}(S)-\sum_{v\in A}\frac{|S_v|}{S} \text{ x Entropy}(S_v) \quad 8$$

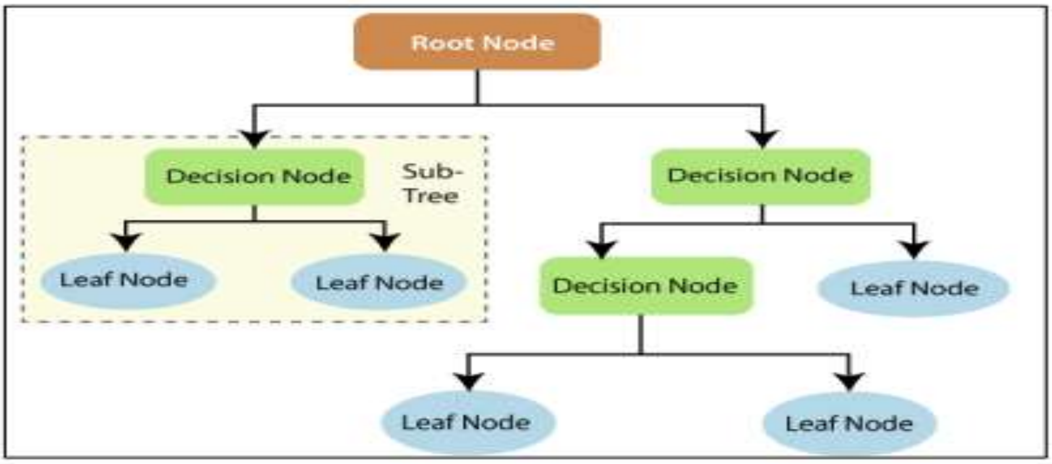


Figure 5: Decision Tree Flow Chart **(**Oluwatoyin & Akinola, 2024).

Support Vector Machine (SVM)

The Support Vector Machine (SVM) is commonly utilised for tasks involving regression or two-group classification, with a primary focus on classification applications (Al-Mejibli et al. 2020). This approach involves the representation of individual data points in a multi-dimensional space, where each dimension corresponds to the value of a specific attribute. In this context, the process of categorization involves identifying the demarcation line that effectively separates the two distinct classes (Al-Mejibli et al. 2020). An ideal hyperplane can be defined as a linear decision function that exhibits the most distinct boundary between vectors belonging to different groups. In the event that there is a requirement for an additional feature, the Support Vector Machine (SVM) employs the kernel algorithm to transform a low-dimensional input space into a higher-dimensional space (Bansal et al., 2022). In essence, it converts seemingly irreconcilable problems into ones that can be integrated. If there are no errors in this separation, then the expected value of the error can be expressed as:

$$E[\Pr(error)] \leq \frac{E[number\ of\ support\ vectors]}{[number\ of\ training\ vectors]} \quad 9$$

The decision function will be given as

$$D(x) = w\emptyset(x) + b \quad 10$$

which is the best line that integrates the training data, w and b are parameters of the SVM, and $\emptyset(x)$ is the function which transforms the data into the new M dimension

Review of Related Studies

Over the past few years, significant research has been conducted to enhance the performance and effectiveness of Intrusion Detection Systems (IDS) using machine learning and artificial intelligence techniques. Several studies have explored the use of hybrid models, combining both signature-based and anomaly-based methods, to achieve higher detection rates and lower false positives.

Sreelakshmi et al. (2025) explores the potential of machine learning techniques to improve the accuracy, speed, and adaptability of IDS. This research study reviews various machine learning approaches, including classification, regression, clustering, dimensionality reduction, semi-supervised, and reinforcement learning, and their application to intrusion detection. Effective data preprocessing and feature selection techniques are crucial for optimizing the performance of machine learning models. The evaluation of IDS performance is discussed, focusing on metrics such as accuracy, precision, recall, and ROC-AUC. The challenges associated with low-quality data, computational complexity, and adversarial attacks are highlighted, along with potential future research directions. By integrating machine learning, IDS can be transformed into more intelligent and adaptive systems, capable of detecting and responding to advanced cyber threats.

Villegas-Ch et al. (2024) examines the implementation and effectiveness of an adaptive intrusion detection system using deep learning algorithms to strengthen cybersecurity. The research focused on evaluating the system's ability to identify and neutralize cyber threats more efficiently and accurately than traditional methods. Quantitative analysis showed that AIDS significantly improved in several key metrics: precision increased by 12.5%, reaching 90%, while recall enhanced by 13.3%, reaching 85%. Furthermore, the F1-score experienced an increase of 12.9%, settling at 87.5%. Qualitative evaluations complemented these results through case studies and testimonials from IT staff, which corroborated the improvement in the detection and response to security incidents. The results reveal that the adaptive intrusion detection system, with its machine learning approach, not only improves threat detection and management but also optimizes operational efficiency, reducing false positives and accelerating response times.

Omarov et al. (2023) introduces a novel framework for identifying network intrusions, leveraging the power of advanced machine learning techniques. The proposed methodology steps away from the rigidity of conventional systems, bringing a flexible, adaptive, and intuitive approach to the forefront of network security. This study employs a diverse blend of machine learning models including but not limited to, Convolutional Neural Networks (CNNs), Support Vector Machines (SVMs), and Random Forests. This research explores an innovative feature extraction and selection technique that enables the model to focus on high-priority potential threats, minimizing noise and improving detection accuracy. The framework's performance has been rigorously evaluated through a series of experiments on benchmark datasets. The results consistently surpass traditional methods, demonstrating a remarkable increase in detection rates and a significant reduction in false positives. Further, the machine learning-based model demonstrated its ability to adapt to new threat landscapes, indicating its suitability in real-world scenarios.

Kataria (2024) developed an innovative machine learning based intrusion detection system to enhance cyber security. The system can rapidly and accurately identify both known and novel risk by leveraging cutting edge machine learning techniques. We trained and validated our model using an extensive dataset encompassing various network scenarios. In comparison to conventional IDS, the ML IDS demonstrated superior detection, accuracy and reduced incidence of false positives. Additionally, the MLIDS provides to be a reliable solution for diverse network. topology is due to its adaptive learning capabilities, making it resilient against evolving cyber threats. this encompasses the design, design, implementation, and evaluation of the MLIDS, highlighting its potential as a valuable tool in next generation, cyber security solutions.

Kumar et al. (2025) presented a novel, scalable Hybrid Autoencoder–Extreme Learning Machine (AE–ELM) framework for Intrusion Detection Systems (IDS), specifically designed to operate effectively in dynamic, cloud-supported IoT environments. The scientific novelty lies in the integration of an Autoencoder for deep feature compression with an Extreme Learning Machine for rapid and accurate classification, enhanced through adaptive thresholding techniques. Evaluated on the CSE-CIC-IDS2018 dataset, the proposed method demonstrates a high detection accuracy of 98.52%, outperforming conventional models in terms of precision, recall, and scalability. Additionally, the framework exhibits strong adaptability to emerging threats and reduced computational overhead, making it a practical solution for real-time, scalable IDS in next-generation network infrastructures.

Golande et al. (2024) explores the development of an efficient network intrusion detection and classification system utilizing machine learning techniques to address these challenges. By leveraging datasets such as NSL-KDD and UNSW-NB15, the author employed a combination of supervised learning algorithms, including Support Vector Machines (SVM), Random Forests, and Neural Networks, alongside comprehensive data preprocessing and feature engineering strategies. The evaluation of their models through metrics like accuracy, precision, recall, and ROC-AUC demonstrates a marked improvement in detection capabilities and computational efficiency. Our findings suggest that machine learning-based IDS can significantly enhance network security by reducing false positives and adapting to emerging threats more effectively than traditional systems. The paper presents a novel approach utilizing machine learning techniques to enhance the efficiency and accuracy of intrusion detection systems (IDS). By employing a combination of supervised and unsupervised learning algorithms, our system can identify and classify both known and unknown threats in real-time. The authors leverage advanced feature selection methods to optimize the performance of our models, ensuring high detection rates with minimal false positives.

METHODOLOGY

Data Description

The dataset to be used for this study included variety of features related to network traffic, both benign and malicious. The key features include source and destination IP addresses, port numbers, transaction protocols, packet sizes, and state information. The dataset captured numerous attributes that are indicative of network behaviors, making it suitable for cyber attack detection. The raw network packets of the UNSW-NB 15 dataset was created by the IXIA PerfectStorm tool in the Cyber Range Lab of UNSW Canberra for generating a hybrid of real modern normal activities and synthetic contemporary attack behaviour. The number of records in the training set is 175,341 records and the testing set is 82,332 records from the different types, attack and normal

Data Preprocessing

Data preprocessing is a crucial phase in the development of machine learning models, as it ensures the dataset is clean, well-structured, and suitable for analysis. This section outlines the various preprocessing techniques applied to the dataset in this research, including handling missing values, encoding categorical variables, standardizing numerical features, and selecting relevant features based on insights derived from exploratory data analysis (EDA). In this study, we began by loading the dataset, separating the feature matrix (X) from the target variable (y), which indicates whether each instance represents a cybersecurity attack or normal activity. Standardization was applied using the StandardScaler from the sklearn library, which transforms the data so that each feature has a mean of zero and a standard deviation of one. This transformation, helps mitigate issues related to features with different scales, which can significantly impact models that rely on distance metrics (e.g., Support Vector Machine and K-Nearest Neighbors).

Handling Class Imbalance

In this project, class imbalance in the target variable (label) to ensure the model provides accurate and unbiased predictions across all classes will be addressed. The Synthetic Minority Over-sampling Technique (SMOTE) was specifically chosen to tackle this imbalance. In the cybersecurity dataset, SMOTE was applied to balance the target variable by increasing the representation of underrepresented attack categories relative to normal traffic or other more common attack types. This approach will ensure that the model was exposed to a sufficient number of instances from all categories during training, enabling it to learn distinguishing features for both majority and minority classes effectively.

Preprocessing for Modeling

In this project, Principal Component Analysis (PCA) was employed as a key dimensionality reduction technique to streamline the dataset while retaining the most informative components of the data. PCA was applied to retain 95% of the dataset's variance, capturing nearly all of the original information while significantly reducing the number of features. Later, PCA was also experimented with a 70% variance retention to further simplify the dataset, thus focusing on only the most essential aspects of the data. By retaining 70% of the variance,

Model Selection

In selecting models for intrusion detection, a comprehensive approach was taken to encompass a variety of machine learning techniques, each with distinct algorithmic foundations and strengths. The chosen classifiers included Support Vector Machine (SVM), Neural Networks, Decision Trees, and Random Forests.

Model Evaluation Metrics

Evaluating the performance of classification models is crucial to understanding their effectiveness and reliability. In this study, accuracy, precision, recall, and F1 score are employed as the primary metrics, each providing insights into different aspects of model behavior

Accuracy

Accuracy is one of the most straightforward metrics for evaluating a classification model (Hossin & Sulaiman, 2015). It measures the proportion of correctly predicted instances (both true positives and true negatives) out of the total number of instances. The formula for accuracy is:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where:

- TP = True Positives (correctly predicted positive instances)
- TN = True Negatives (correctly predicted negative instances)
- FP = False Positives (incorrectly predicted positive instances)
- FN = False Negatives (incorrectly predicted negative instances)

Precision

Precision, also known as the positive predictive value, measures the proportion of true positive predictions out of all positive predictions (Cabot & Ross, 2023). It provides insight into the accuracy of the positive predictions made by the model. The formula for precision is:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall (Sensitivity)

Recall, also known as sensitivity or true positive rate, measures the proportion of true positive predictions out of all actual positive instances (Miao & Zhu, 2022). It indicates how well the model is able to identify all positive instances in the dataset. The formula for recall is:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score

The F1-score is the harmonic mean of precision and recall. It provides a single metric that balances both precision and recall, making it particularly useful when dealing with imbalanced datasets (Miao & Zhu, 2022). The F1-score is calculated as:

$$F1\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

RESULT

Data Description

The dataset used for this cyber security project includes a variety of features related to network traffic, both benign and malicious. The key features include source and destination IP addresses, port numbers, transaction protocols, packet sizes, and state information. The dataset captures numerous attributes that are indicative of network behaviors, making it suitable for cyber attack detection. The dataset consists of 49 columns, including both categorical and numerical variables, with over 175,000 records representing different types of network traffic and attacks. A snapshot of the dataset columns is as shown in the table below:

No.	Feature Name	Type	Description
1	srcip	Nominal	Source IP address
2	sport	Integer	Source port number
3	dstip	Nominal	Destination IP address
4	dsport	Integer	Destination port number
5	proto	Nominal	Transaction protocol
6	state	Nominal	Indicates the state and its dependent protocol
7	dur	Float	Record total duration
8	sbytes	Integer	Source to destination transaction bytes
9	dbytes	Integer	Destination to source transaction bytes
10	sttl	Integer	Source to destination time to live value
11	dttl	Integer	Destination to source time to live value
12	sloss	Integer	Source packets retransmitted or dropped
13	dloss	Integer	Destination packets retransmitted or dropped
14	service	Nominal	Service type (e.g., HTTP, FTP, SMTP, SSH, DNS)
15	Sload	Float	Source bits per second
16	Dload	Float	Destination bits per second
17	Spkts	Integer	Source to destination packet count
18	Dpkts	Integer	Destination to source packet count
19	swin	Integer	Source TCP window advertisement value
20	dwin	Integer	Destination TCP window advertisement value
21	stcpb	Integer	Source TCP base sequence number
22	dtcpb	Integer	Destination TCP base sequence number
23	smeansz	Integer	Mean packet size transmitted by the source
24	dmeansz	Integer	Mean packet size transmitted by the destination
25	trans_depth	Integer	Represents the pipelined depth into the connection

26	res_bdy_len	Integer	Actual uncompressed content size of the data transferred (response body length)
27	Sjit	Float	Source jitter (mSec)
28	Djit	Float	Destination jitter (mSec)
29	Stime	Timestamp	Record start time
30	Ltime	Timestamp	Record last time
31	Sintpkt	Float	Source interpacket arrival time (mSec)
32	Dintpkt	Float	Destination interpacket arrival time (mSec)
33	tcprtt	Float	TCP connection setup round-trip time (sum of synack and ackdat)
34	synack	Float	TCP connection setup time between SYN and SYN-ACK
35	ackdat	Float	TCP connection setup time between ACK and data
36	is_sm_ips_ports	Binary	If source (1) and destination (3) IP addresses and ports are the same (1 if true, 0 if false)
37	ct_state_ttl	Integer	Count for each state according to a specific record's TTL value
38	ct_flw_http_mthd	Integer	Number of flows that use HTTP methods like GET and POST
39	is_ftp_login	Binary	Whether an FTP session is accessed by user login (1 if true, 0 if false)
40	ct_ftp_cmd	Integer	Number of flows with an FTP command in the session
41	ct_srv_src	Integer	Number of connections with the same service and source address
42	ct_srv_dst	Integer	Number of connections with the same service and destination address
43	ct_dst_ltm	Integer	Number of connections to the same destination address
44	ct_src_ltm	Integer	Number of connections from the same source address
45	ct_src_dport_ltm	Integer	Number of connections from the same source address and source port
46	ct_dst_sport_ltm	Integer	Number of connections to the same destination address and destination port
47	ct_dst_src_ltm	Integer	Number of connections between the same source and destination
48	attack_cat	Nominal	The name of the attack category
49	Label	Binary	0 for normal and 1 for attack records

Table 2: Dataset Columns

Exploratory Data Analysis (EDA)

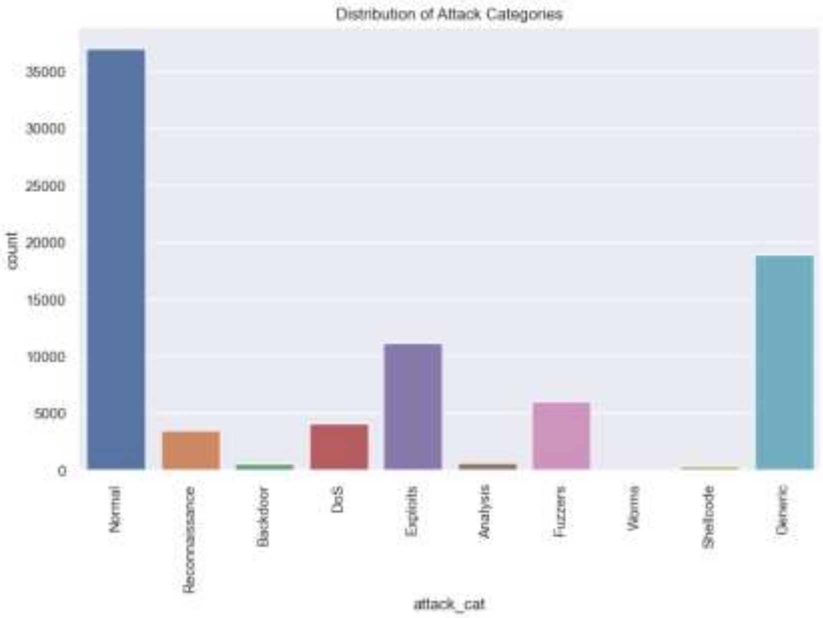


Figure 6: Distribution of Attack Categories (Researcher, 2025)

The dataset’s bar chart reveals that the majority of network traffic is classified as Normal, with over 35,000 instances, indicating prevalent benign activities. Following this, Generic attacks constitute the largest category of malicious activity, highlighting their significant presence in the dataset. Exploit-based attacks are the next most frequent, reflecting attempts to leverage system vulnerabilities. Other attack types such as Fuzzers, Reconnaissance, and Denial of Service (DoS) occur less frequently but represent important threat vectors. Less common attacks include Backdoor, Analysis, Shellcode, and Worms.

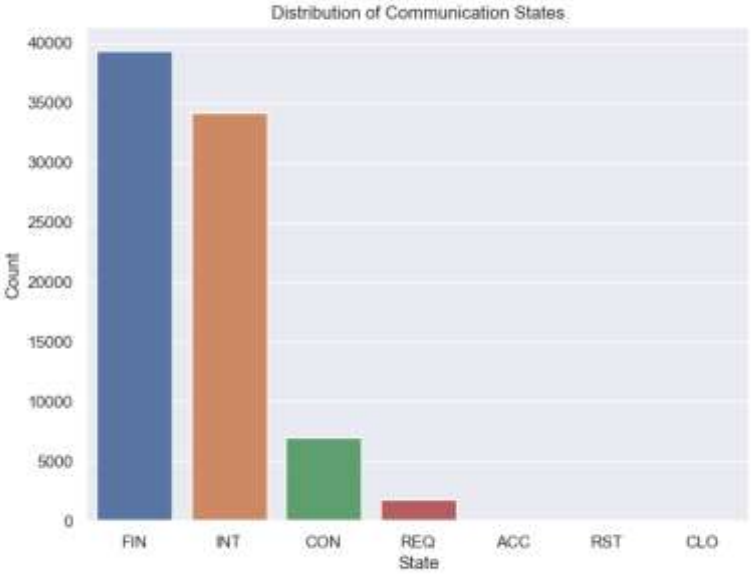


Figure 7: Communication Stats Across Netwirk Traffic

The bar chart of communication states in the dataset shows that the "FIN" state is the most frequent, with nearly 40,000 instances, indicating the majority of network sessions were finished or completed. The "INT" state follows closely with just under 35,000 occurrences, representing ongoing or intermediate communication, suggesting active sessions during data capture. The "CON" state, with fewer than 10,000 occurrences, likely represents connections currently being established or maintained but is less common. Other states such as "REQ," "ACC," "RST," and "CLO" have minimal counts, indicating they occur less frequently or transit quickly.

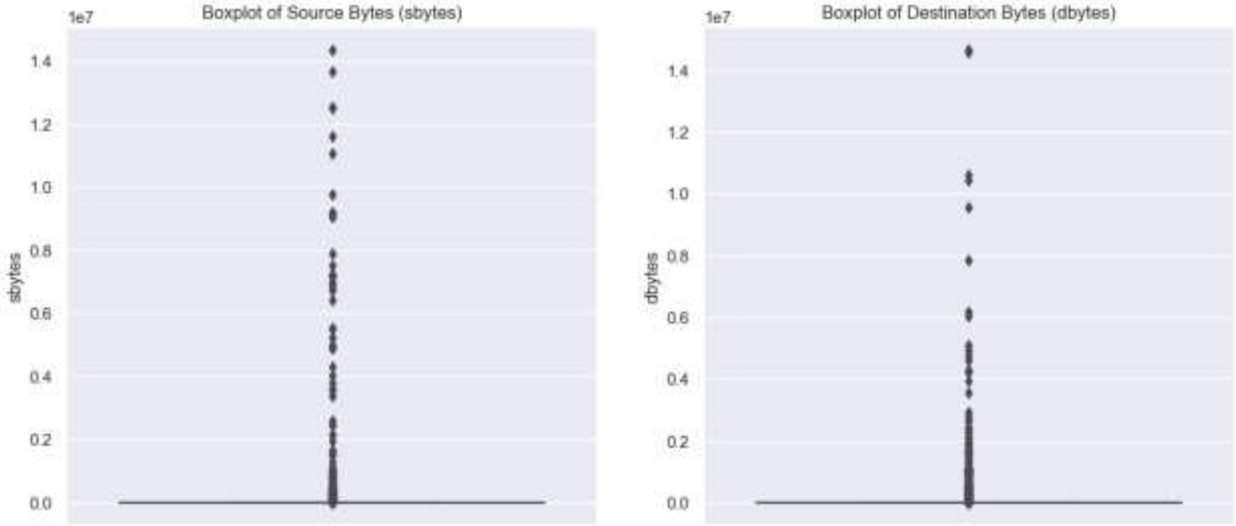


Figure 8: Distribution of Source and Destination in Byte in Network Traffic

The boxplots show the distribution of source bytes (sbytes) and destination bytes (dbytes) in network traffic, revealing that most packets carry relatively small data sizes. Both plots exhibit a dense cluster of data points near the lower byte range, indicating the majority of network communications involve low-volume data transfers. However, numerous extreme outliers exist, with byte counts reaching up to about 14 million. These outliers suggest occasional large data exchanges, such as file downloads or potential malicious activity like data exfiltration.

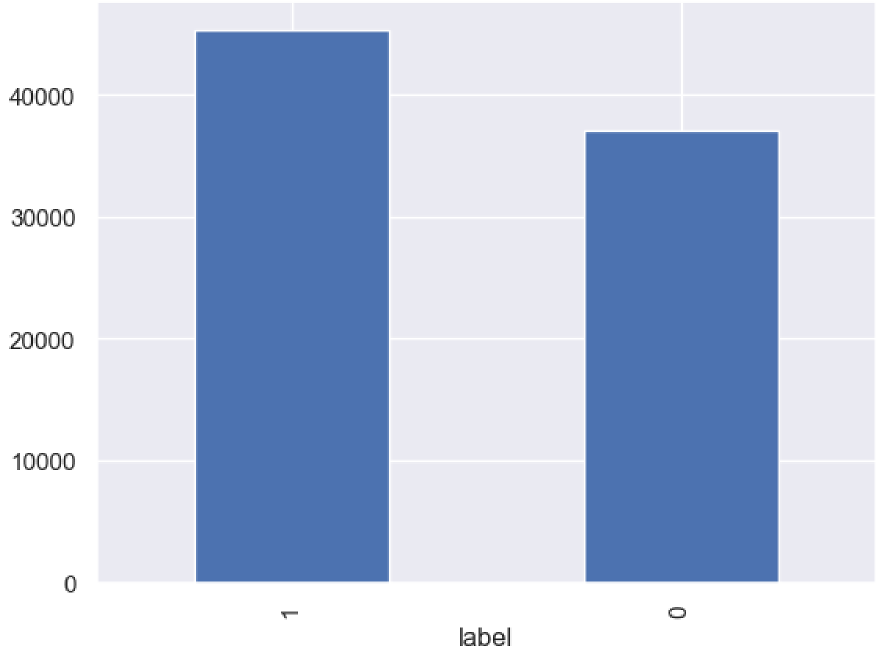


Figure 9: Distribution of Instances

The bar chart shows a nearly balanced dataset with attack instances ("1") slightly exceeding 40,000 and non-attack instances ("0") just below that. This distribution provides substantial data for training cybersecurity models like Intrusion Detection Systems, helping distinguish between attack and normal behaviors. However, the slight imbalance may necessitate resampling techniques to prevent model bias toward the more frequent attack category, ensuring accurate detection across both classes.

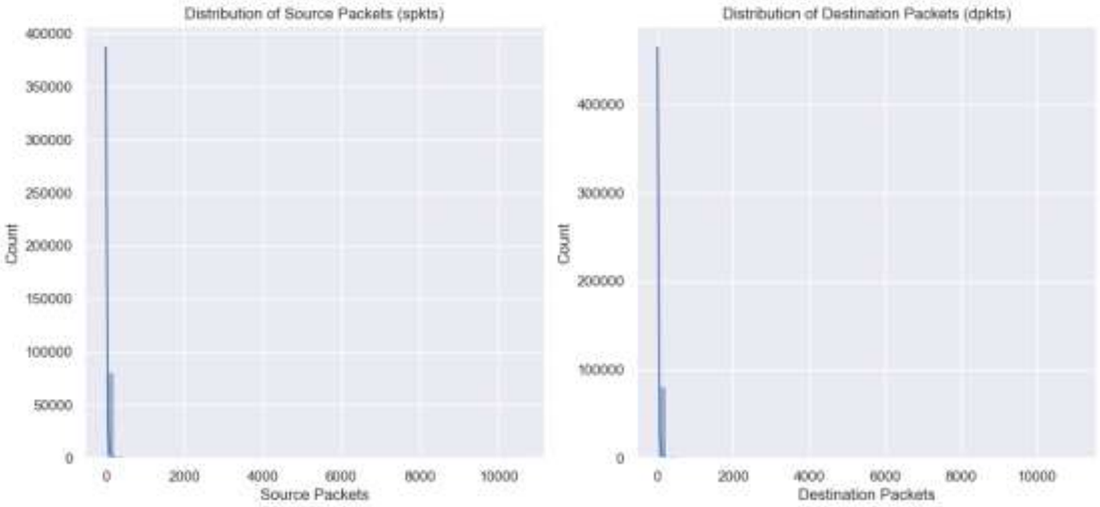


Figure 11: Distribution of Packet Count

The histograms display highly right-skewed distributions for source packets (spkts) and destination packets (dpkts), showing that most network flows consist of relatively few packets, typically under 1,000. A dense concentration of values near zero highlights frequent small or short sessions. However, long tails reveal a minority of instances with substantially higher packet counts, indicating larger data exchanges. These rare, larger flows may represent abnormal or suspicious activities, such as attacks or anomalies.

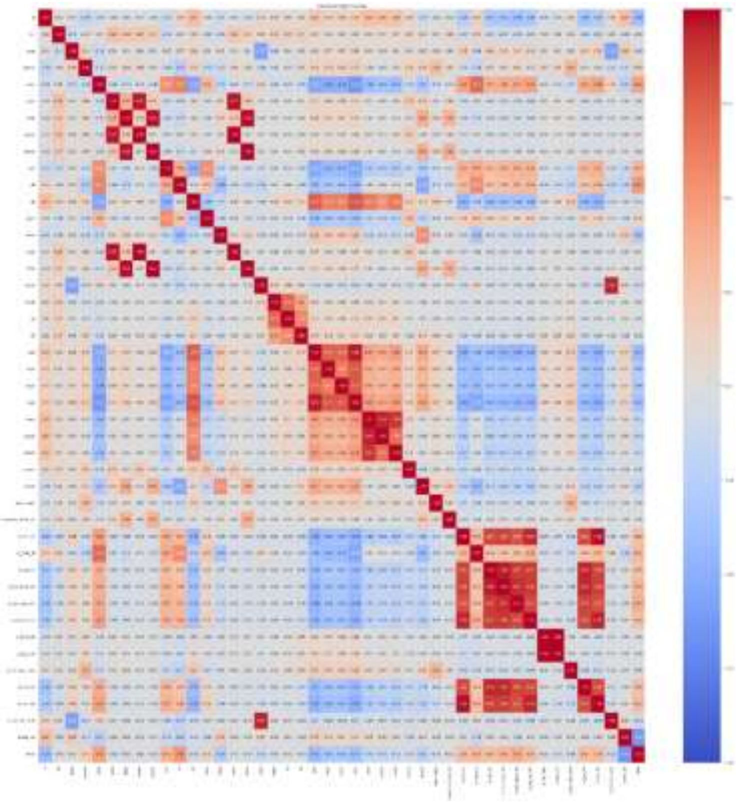


Figure 12: Correlation Matrix

The heatmap presents a correlation matrix where colors range from deep blue (strong negative correlation) to deep red (strong positive correlation). Correlation coefficients near 1 indicate that as one feature increases, the other tends to increase, while coefficients near -1 show opposing trends. The heatmap reveals clusters of highly interrelated features, suggesting shared network or traffic patterns, alongside negatively correlated feature pairs indicating contrasting behaviors. Negative correlations may reveal complementary features that improve predictive power. This matrix informs feature selection and dimensionality reduction techniques like PCA, optimizing intrusion detection system performance.

Evaluation of Machine Learning Models

Evaluation of Support Vector Machine

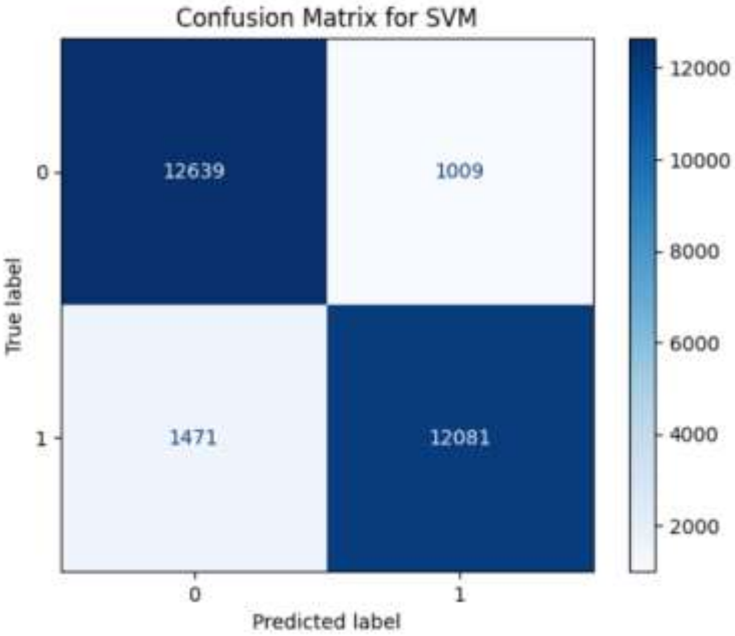
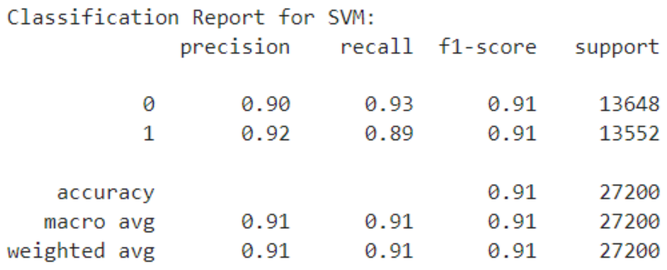


Figure 13: Confusion matrix for the Support Vector Machine (SVM)

The confusion matrix for the SVM model shows it correctly identified 12,639 true negatives (no attack) and 12,081 true positives (attacks), demonstrating strong detection capabilities for both classes. However, there were 1,009 false positives, where normal instances were misclassified as attacks, potentially causing unnecessary alerts. More critically, 1,471 false negatives occurred, meaning actual attacks were missed, posing security risks. While the SVM performs well overall, reducing false negatives is essential, as missed detections are more costly in intrusion detection. This highlights the need for further model tuning or exploring alternative models to enhance accuracy and robustness.



```
Classification Report for SVM:
              precision    recall  f1-score   support

           0       0.90      0.93      0.91     13648
           1       0.92      0.89      0.91     13552

    accuracy                           0.91     27200
   macro avg       0.91      0.91      0.91     27200
weighted avg       0.91      0.91      0.91     27200
```

Figure 14: Classification Report for SVM

The classification report for the SVM model shows strong performance across key metrics. For class 0 (no attack), precision is 0.90, recall 0.93, and F1-score 0.91, indicating effective identification and low false positives. For class 1 (attack), precision is 0.92, recall 0.89, and

F1-score 0.91, reflecting good detection with slightly fewer true positives. Both classes have balanced support. Overall accuracy is 91%, demonstrating reliable classification. Macro and weighted averages for precision, recall, and F1-score are all 0.91, underscoring the model's balanced and consistent ability to predict attacks and non-attacks accurately.

Evaluation of K-Nearest Neighbor

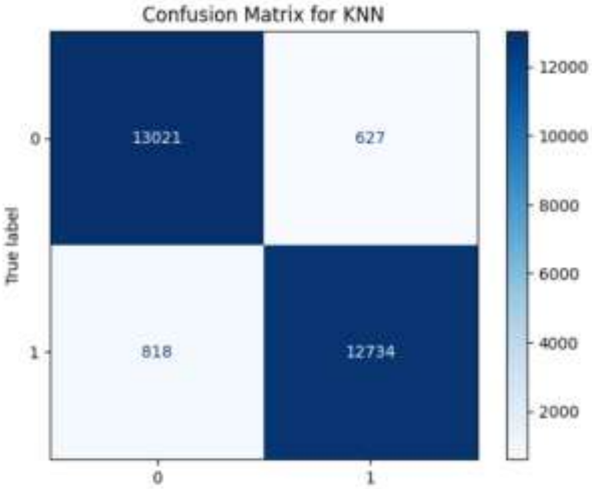
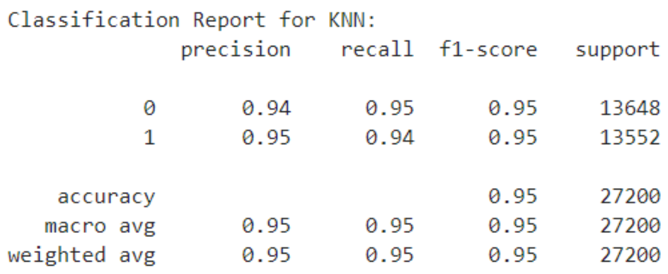


Figure 15: Confusion matrix for the K-Nearest Neighbors (KNN)

The confusion matrix for the K-Nearest Neighbors (KNN) model shows strong classification performance, with 13,021 true negatives (correctly identified non-attacks) and 12,734 true positives (correctly detected attacks). The model has 627 false positives, indicating some normal instances were misclassified as attacks, and 818 false negatives, meaning some attacks were missed. While the high true positive and true negative counts demonstrate good overall detection ability, the false negatives highlight a risk of missed attacks, which is critical in cybersecurity. The relatively low false positive rate helps reduce unnecessary false alarms, making KNN effective but requiring careful consideration of missed detections.



```
Classification Report for KNN:
              precision    recall  f1-score   support

           0       0.94      0.95      0.95     13648
           1       0.95      0.94      0.95     13552

    accuracy                           0.95     27200
   macro avg       0.95      0.95      0.95     27200
weighted avg       0.95      0.95      0.95     27200
```

Figure 16: Classification Report for K-Nearest Neighbors (KNN)

The classification report for the K-Nearest Neighbors (KNN) model shows strong and balanced performance. For class 0 (non-attack), precision is 0.94, recall 0.95, and F1-score 0.95, indicating excellent detection of non-attack instances with minimal false positives. For class 1 (attack), precision is 0.95, recall 0.94, and F1-score 0.95, reflecting effective attack identification with slightly fewer true positives. The model achieves an overall accuracy of 95%, with macro and weighted averages for precision, recall, and F1-score also at 0.95. These results demonstrate KNN's reliability and suitability for intrusion detection tasks, balancing false positives and false negatives effectively.

Neural Network

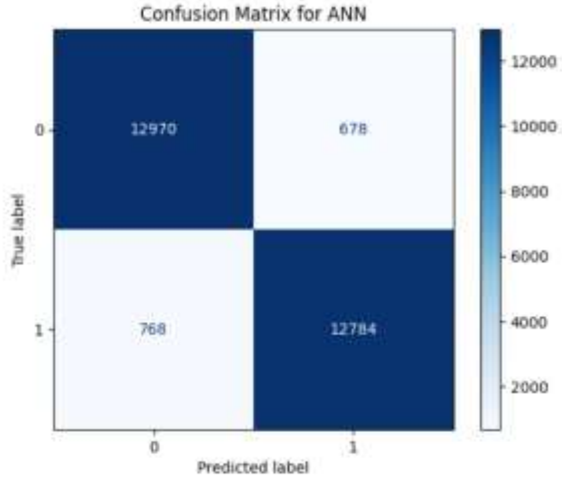
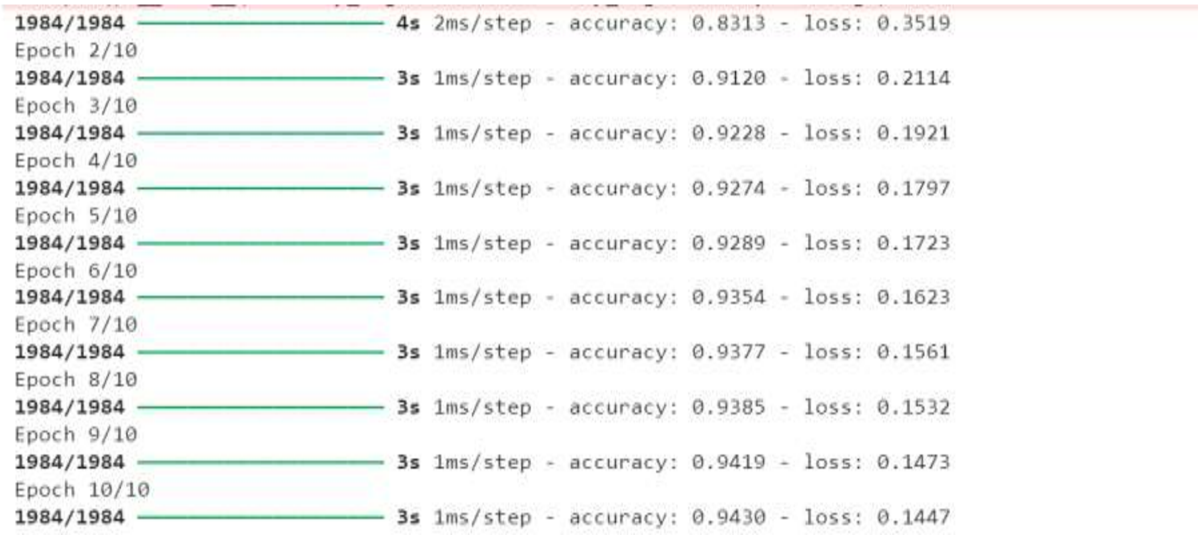


Figure 17: Confusion Matrix for Artificial Neural Network (ANN) (Researcher, 2025)

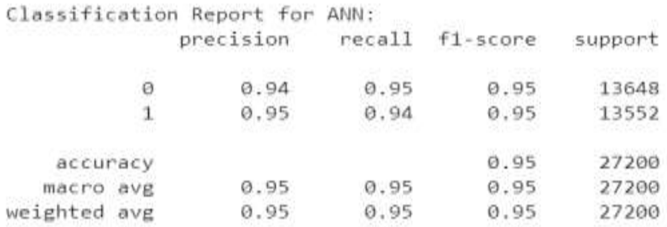
The confusion matrix of the Artificial Neural Network (ANN) model shows strong classification performance with 12,784 true positives (correctly detected attacks) and 12,970 true negatives (correctly identified non-attacks). The model has 678 false positives, where normal activities were misclassified as attacks, and 768 false negatives, representing missed attacks. Despite the overall high accuracy and balanced detection, false negatives are critical in cybersecurity as they indicate undetected threats, posing risks. The relatively low false positive rate minimizes false alarms.



```
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 4s 2ms/step - accuracy: 0.8313 - loss: 0.3519
Epoch 2/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9120 - loss: 0.2114
Epoch 3/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9228 - loss: 0.1921
Epoch 4/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9274 - loss: 0.1797
Epoch 5/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9289 - loss: 0.1723
Epoch 6/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9354 - loss: 0.1623
Epoch 7/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9377 - loss: 0.1561
Epoch 8/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9385 - loss: 0.1532
Epoch 9/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9419 - loss: 0.1473
Epoch 10/10
1984/1984 ━━━━━━━━━━━━━━━━━━━━ 3s 1ms/step - accuracy: 0.9430 - loss: 0.1447
```

Figure 17: Training Progress of an ANN

The training output of the Artificial Neural Network (ANN) over 10 epochs shows steady improvement. Starting with an accuracy of 83.13% and a loss of 0.3519 in epoch 1, the model refines its weights through each pass over the dataset. By epoch 10, accuracy rises to 94.30%, while loss reduces to 0.1447, indicating improved prediction precision and minimized errors. Training each epoch takes about 3 seconds, reflecting efficient computation. The increasing accuracy coupled with decreasing loss signifies effective model convergence, suggesting the ANN is learning well and approaching optimal performance, though further validation on test data is necessary to confirm generalization.



```
Classification Report for ANN:
              precision    recall  f1-score   support

           0       0.94      0.95      0.95     13648
           1       0.95      0.94      0.95     13552

    accuracy                           0.95     27200
   macro avg       0.95      0.95      0.95     27200
weighted avg       0.95      0.95      0.95     27200
```

Figure 18: Classification Report for ANN

The Artificial Neural Network (ANN) training over 10 epochs demonstrates clear learning progress. Initially, the model achieves 83.13% accuracy and a loss of 0.3519. With each epoch, accuracy steadily increases, reaching 94.30% by the 10th epoch, while loss decreases to 0.1447. This trend reflects effective optimization and improved prediction accuracy. Each epoch completes quickly, indicating efficient processing. The converging accuracy and loss values suggest the model is approaching its optimal performance, although further testing is needed to confirm generalizability.

Evaluation of Decision Tree

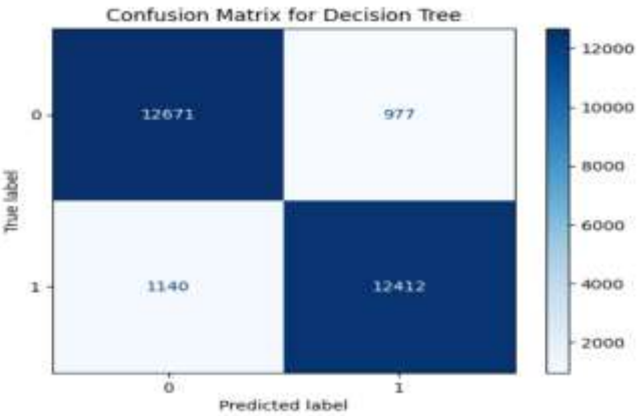
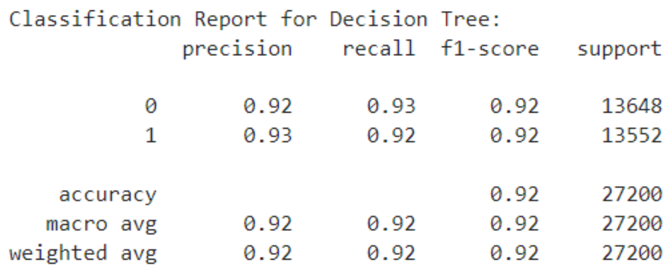


Figure 19: Confusion Matrix for the Decision Tree Model

The confusion matrix for the Decision Tree model shows it correctly identified 12,671 true negatives (non-attacks) and 12,412 true positives (attacks), demonstrating strong classification ability. However, there were 977 false positives, indicating some normal instances were mistakenly flagged as attacks, and 1,140 false negatives, where actual attacks were missed. The false negatives are particularly concerning as they represent security threats that went undetected.

applications.



```
Classification Report for Decision Tree:
              precision    recall  f1-score   support

           0       0.92      0.93      0.92     13648
           1       0.93      0.92      0.92     13552

    accuracy                           0.92     27200
   macro avg       0.92      0.92      0.92     27200
weighted avg       0.92      0.92      0.92     27200
```

Figure 20: Classification Report for Decision Tree

The classification report for the Decision Tree model shows an overall accuracy of 92%, indicating strong overall predictive capability. For class 0 (normal), precision is 0.92, recall 0.93, and F1-score 0.92, reflecting accurate identification and low false positive rates. For class 1 (attack), precision is 0.93, recall 0.92, and F1-score 0.92, showing effective detection of attacks while minimizing false alarms. Macro and weighted averages of 0.92 across precision, recall, and F1-score indicate balanced performance across both classes. This balanced and reliable performance makes the Decision Tree model well-suited for intrusion detection, effectively detecting threats with manageable false alarms.

Evaluation of Random Forest

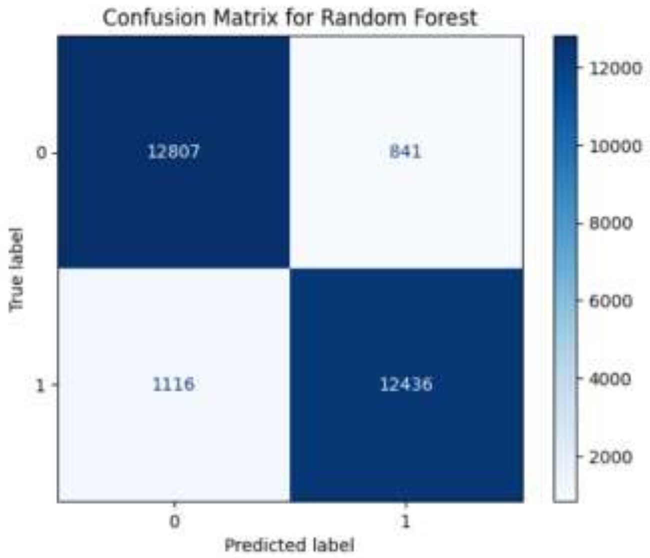
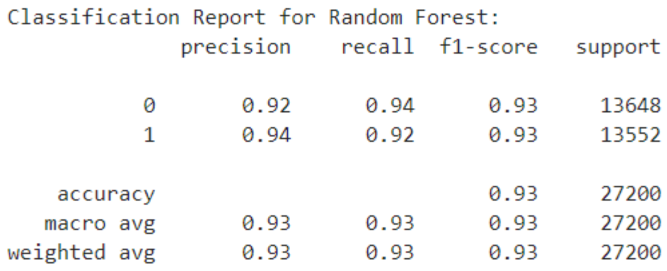


Figure 21: Confusion Matrix for the Random Forest Model

The confusion matrix for the Random Forest model shows strong classification results with 12,807 true negatives and 12,436 true positives, indicating effective identification of both normal and attack traffic. The model misclassified 841 normal instances as attacks (false positives) and missed 1,116 attacks (false negatives). While the false positives are relatively low, reducing false negatives is crucial as missed attacks pose significant security risks. Overall, the Random Forest model demonstrates reliable threat detection with a good balance between detection accuracy and false alarm rates. Further tuning could improve its ability to minimize missed attacks for enhanced security.



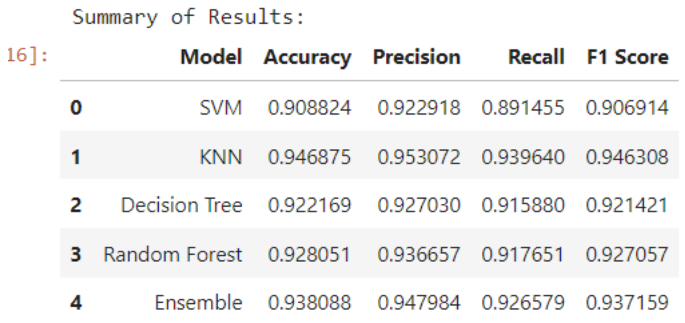
```
Classification Report for Random Forest:
              precision    recall  f1-score   support

           0       0.92      0.94      0.93     13648
           1       0.94      0.92      0.93     13552

    accuracy                           0.93     27200
   macro avg       0.93      0.93      0.93     27200
weighted avg       0.93      0.93      0.93     27200
```

Figure 21: Classification Report for Random Forest

The classification report for the Random Forest model shows strong performance with an overall accuracy of 93%. For class 0 (normal traffic), precision is 0.92, recall 0.94, and F1-score 0.93, indicating accurate identification of legitimate traffic while minimizing false alarms. For class 1 (attack traffic), precision is 0.94, recall 0.92, and F1-score 0.93, reflecting effective detection of malicious instances with balanced precision and recall. Macro and weighted averages of 0.93 confirm consistent performance across both classes.

4.4 Summary of Results

The summary of results showcases the performance metrics for various machine learning models Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree, Random Forest, and an Ensemble model. Each model has been evaluated on four primary metrics: accuracy, precision, recall, and F1 score. Here is a detailed interpretation of each model's performance:



Summary of Results:

16]:

	Model	Accuracy	Precision	Recall	F1 Score
0	SVM	0.908824	0.922918	0.891455	0.906914
1	KNN	0.946875	0.953072	0.939640	0.946308
2	Decision Tree	0.922169	0.927030	0.915880	0.921421
3	Random Forest	0.928051	0.936657	0.917651	0.927057
4	Ensemble	0.938088	0.947984	0.926579	0.937159

Figure 22: Summary of Results

Among the individual machine learning models for intrusion detection, K-Nearest Neighbors (KNN) leads with the highest accuracy of 94.69%, showing strong capability in classifying both attack and non-attack instances. KNN also attains the highest precision (0.9531) and recall (0.9396), minimizing false positives and negatives effectively. The F1 score of 0.9463 further confirms its balanced performance. The Random Forest model closely follows with 92.81% accuracy, high precision (0.9367), and recall (0.9177), providing reliable detection with fewer false alarms. The Decision Tree and SVM models show slightly lower but still strong performance, with accuracies around 90-92%. Overall, KNN is the most accurate and balanced individual model, while ensemble methods like Random Forest also offer robust results for intrusion detection tasks.

DISCUSSION OF FINDINGS

The classification results highlight K-Nearest Neighbors (KNN) as the top performer with 94.69% accuracy, 95.31% precision, 93.96% recall, and a 94.63% F1 score, reflecting strong and balanced classification. Random Forest follows closely, achieving 92.81% accuracy and a 92.71% F1 score due to its ensemble approach handling diverse patterns effectively. The Decision Tree and Support Vector Machine (SVM) models also deliver solid performance with accuracies of 92.22% and 90.88%, respectively. Overall, KNN excels, making it particularly suited for intrusion detection, while Random Forest provides robust alternative performance, demonstrating the effectiveness of machine learning for cybersecurity classification tasks. Comparing with other studies, Sayed et al. (2024) analyzed historical cyberattack data reported Random Forest achieving the highest accuracy at 90%, with precision and recall also outperforming other classifiers including KNN, which lagged behind in that context. Nabi & Zhou (2024) focused on network intrusion systems showed Random Forest outperforming Decision Trees and Logistic Regression, reaching above 90% accuracy on benchmark datasets such as NSL-KDD and UNSW-NB1. Meanwhile, KNN, despite its simplicity, was often noted to underperform compared to ensemble models in large-scale intrusion detection scenarios due to its sensitivity to noisy data and class imbalances. Mohamed (2025) noted that Random Forest and SVM are among the most robust ML models for cybersecurity applications, frequently surpassing 90% accuracy, while KNN's results markedly fluctuate depending on feature selection and noise sensitivity. However, the present research demonstrates an edge in KNN performance, with accuracy surpassing typical cybersecurity-focused benchmarks. This suggests that through targeted data preprocessing and parameter tuning, KNN can be highly effective in certain cybersecurity datasets, especially where data characteristics favor instance-based approaches. The robust performance of Random Forest aligns closely with prevailing literature but still reflects the high standards achieved in this study. Performance metrics such as precision, recall, and F1 score further underscore the practical utility of these models in intrusion detection, where minimizing false alarms and missed detections is critical.

CONCLUSION AND RECOMMENDATION

This study evaluated machine learning models K-Nearest Neighbors (KNN), Random Forest, Artificial Neural Network (ANN), Support Vector Machine (SVM), and Decision Tree for intrusion detection on a comprehensive cybersecurity dataset. KNN achieved the highest accuracy (94.69%) and balanced precision, recall, and F1 scores, highlighting its strong classification capability. Random Forest and ANN also demonstrated robust performance, leveraging ensemble learning and neural networks to manage complex data patterns. Key contributions from balanced class representation (SMOTE) and dimensionality reduction (PCA) improved model effectiveness. Overall, KNN and ensemble models are highly effective for real-world cybersecurity intrusion detection, minimizing false positives and negatives and enhancing network security.

The study recommends prioritizing K-Nearest Neighbors (KNN) algorithms in cybersecurity frameworks for their superior accuracy and ability to reduce false positives and negatives. Also incorporating Synthetic Minority Over-sampling Technique (SMOTE) and Principal Component Analysis (PCA) in preprocessing to handle class imbalance and high dimensionality for enhanced robustness. Lastly, regular retraining and tuning with current cybersecurity data are essential to adapt to new attack patterns and maintain high detection efficiency in dynamic threat environments.

REFERENCES

Akeiber, H. J. (2025). The Evolution of Social Engineering Attacks: A Cybersecurity Engineering Perspective. *Al-Rafidain Journal of Engineering Sciences*, 294-316.

Al Farsi, A., Khan, A., Bait-Suwailam, M. M., & Mughal, M. R. (2024). Comparative Performance Evaluation of Machine Learning Algorithms for Cyber Intrusion Detection.

Al Hwaitat, A. K., Fakhouri, H. N., Alawida, M., Atoum, M. S., Abu-Salih, B., Salah, I. K., ... & Alassaf, N. (2024). Overview of Mobile Attack Detection and Prevention Techniques Using Machine Learning. *International Journal of Interactive Mobile Technologies*, *18*(10).

Al Qwaid, M. (2025). Cybersecurity Threats: Ransomware, Phishing, and Social Engineering. In *Complexities and Challenges for Securing Digital Assets and Infrastructure* (pp. 399-434). IGI Global Scientific Publishing.

Al Saidat, M. R., Yerima, S. Y., & Shaalan, K. (2024). Advancements of SMS spam detection: A comprehensive survey of NLP and ML techniques. *Procedia Computer Science*, *244*, 248-259.

Alam, M., Pandey, B., Ahmad, S., Shahid, M., & Ahmad, F. (2024, December). Machine learning in cybersecurity: Opportunities and challenges. In *2024 IEEE 16th international conference on computational intelligence and communication networks (CICN)* (pp. 663-670). IEEE.

Al-Mejibli, I. S., Alwan, J. K., & Abd, D. H. (2020). The effect of gamma value on support vector machine performance with different kernels. *Int. J. Electr. Comput. Eng*, *10*(5), 5497-5506.

Almomani, A., Aoudi, S., Al-Qerem, A., Aldweesh, A., & Alkasassbeh, M. (2025). Behavioral Analysis of AI-Generated Malware: New Frontiers in Threat Detection. In *Examining Cybersecurity Risks Produced by Generative AI* (pp. 211-234). IGI Global Scientific Publishing.

Ansarullah, S. I., Wali, A. W., Rasheed, I., & Rayees, P. Z. (2024). AI-powered strategies for advanced malware detection and prevention. In *The Art of Cyber Defense* (pp. 3-24). CRC Press.

Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A., & Akin, E. (2023). A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. *Electronics*, *12*(6), 1333.

Atıcı, S., & Tuna, G. (2025). Impact of cybersecurity attacks on electrical system operation. In *Cyber Security Solutions for Protecting and Building the Future Smart Grid* (pp. 117-160). Elsevier.

Ayeni, R. K., Adebiyi, A. A., Okesola, J. O., & Igbekele, E. (2024, April). Phishing attacks and detection techniques: A systematic review. In *2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)* (pp. 1-17). IEEE.

Bansal, M., Goyal, A., & Choudhary, A. (2022). A comparative analysis of K-nearest neighbor, genetic, support vector machine, decision tree, and long short term memory algorithms in machine learning. *Decision analytics journal*, *3*, 100071.

Barker, J. (2024). *Hacked: The Secrets Behind Cyber Attacks*. Kogan Page Publishers.

Bethany, M., Galiopoulos, A., Bethany, E., Karkevandi, M. B., Vishwamitra, N., & Najafirad, P. (2024). Large language model lateral spear phishing: A comparative study in large-scale organizational settings. *arXiv preprint arXiv:2401.09727*.

Birthriya, S. K., Ahlawat, P., & Jain, A. K. (2025). A comprehensive survey of social engineering attacks: taxonomy of attacks, prevention, and mitigation strategies. *Journal of Applied Security Research, 20*(2), 244-292.

Cabot, J. H., & Ross, E. G. (2023). Evaluating prediction model performance. *Surgery*, *174*(3), 723-726.

Chakraborty, S., Pandey, S. K., Maity, S., & Dey, L. (2024). Detection and classification of novel attacks and anomaly in IoT network using rule based deep learning model. *SN Computer Science*, *5*(8), 1056.

Dobrovolska, O., & Rozhkova, M. (2024). The Impact of Digital Transformation on the Anti-Corruption and Cyber-Fraud System. *Business Ethics and Leadership*, *8*(3), 231-252.

George, A. S., Baskar, T., & Srikaanth, P. B. (2024). Cyber threats to critical infrastructure: assessing vulnerabilities across key sectors. *Partners Universal International Innovation Journal*, *2*(1), 51-75.

Golande, S., Vaidya, S., Pardeshi, A., Katkade, V., & Pawar, V. (2024). An Efficient Network Intrusion Detection and Classification System using Machine Learning. *International Journal of Advanced Research in Science, Communication and Technology*.

Gundoor, T. K., & Mulimani, R. (2025). AI-Based Solutions for Malware Detection and Prevention. In *Machine Intelligence Applications in Cyber-Risk Management* (pp. 107-134). IGI Global Scientific Publishing.

Gururaj, H. L., Janhavi, V., & Ambika, V. (Eds.). (2024). *Social Engineering in Cybersecurity: Threats and Defenses*. CRC Press.

Hasudungan, A., Muliono, R., Khairina, N., & Novita, N. (2024). The Impact of k-means on Association Rules Mining Algorithms Performance. *Journal of Computer Science, Information Technology and Telecommunication Engineering*, *5*(2), 640-653.

Hawamdah, L. M. (2024). *From Hooks to Clicks: A Data-Driven Approach to Understanding Language Trends in Phishing Schemes Across Different Attack Vectors* (Doctoral dissertation, The George Washington University).

Hossin, M., & Sulaiman, M. N. (2015). A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process*, *5*(2), 1.

Hussain, M. D., Rahman, M. H., & Ali, N. M. (2024). Artificial intelligence and machine learning enhance robot decision-making adaptability and learning capabilities across various domains. *International Journal of Science and Engineering*, *1*(3), 14-27.

Hyunjulie (2019). Types of Machine. Accessed at Learninghttps://medium.com/hyunjulie/machine-learning-studying-roadmap-8596b6571f8a

Imamverdiyeva, Y., & Baghirovb, E. (2024). Evasion techniques in malware detection: challenges and countermeasures. *Problems of Information Technology*, *15*(2), 9-15.

Kataria, A. (2023, November). An ML-Based Intrusion Detection System Design and Evaluation for Enhanced Cybersecurity. In *2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI)* (pp. 1036-1040). IEEE.

Khan, I. K., Daud, H. B., Zainuddin, N. B., Sokkalingam, R., Farooq, M., Baig, M. E., ... & Zafar, M. (2024). Determining the optimal number of clusters by Enhanced Gap Statistic in K-mean algorithm. *Egyptian Informatics Journal*, *27*, 100504.

Khatoon, A., Ullah, A., & Qureshi, K. N. (2024). Ai models and data analytics. *Next Generation AI Language Models in Research: Promising Perspectives and Valid Concerns*, *45*.

Koroniotis, N., Moustafa, N., Schiliro, F., Gauravaram, P., & Janicke, H. (2020). A holistic review of cybersecurity and reliability perspectives in smart airports. *IEEE Access*, *8*, 209802-209834.

Kovalchuk, D. (2024). Malware development: From early viruses to modern cyber threats. *Вісник Черкаського державного технологічного університету. Технічні науки*, *29*(3), 10-20.

Kumar, A., Radhakrishnan, R., Sumithra, M., Kaliyaperumal, P., Balusamy, B., & Benedetto, F. (2025). A Scalable Hybrid Autoencoder–Extreme Learning Machine Framework for Adaptive Intrusion Detection in High-Dimensional Networks. *Future Internet*, *17*(5), 221.

Kumar, I. (2023). Emerging threats in cybersecurity: a review article. *International Journal of Applied and Natural Sciences*, *1*(1), 01-08.

Lawelai, H., Purnomo, E. P., Nurmandi, A., Jovita, H., & Baulete, E. M. (2025). Cybersecurity policy on smart city infrastructure: a mapping of new threats and protections. *Journal of Science and Technology Policy Management*.

Lee, Y., Lee, J., Ryu, D., Park, H., & Shin, D. (2024). Clop Ransomware in Action: A Comprehensive Analysis of Its Multi-Stage Tactics. *Electronics*, *13*(18), 3689.

Mallick, M. A. I., & Nath, R. (2024). Navigating the cyber security landscape: A comprehensive review of cyber-attacks, emerging trends, and recent developments. *World Scientific News*, *190*(1), 1-69.

Manjramkar, M. A., & Jondhale, K. C. (2023, May). Cyber security using machine learning techniques. In *International Conference on Applications of Machine Intelligence and Data Analytics (ICAMIDA 2022)* (pp. 680-701). Atlantis Press.

Manoharan, A., & Sarker, M. (2023). Revolutionizing cybersecurity: Unleashing the power of artificial intelligence and machine learning for next-generation threat detection. *DOI: https://www. doi. org/10.56726/IRJMETS32644*, *1*.

Mazhar, L., & Rohatgi, S. (2025). Malware Analysis and Detection: New Approaches and Techniques. *Emerging Threats and Countermeasures in Cybersecurity*, 83-109.

Miao, J., & Zhu, W. (2022). Precision–recall curve (PRC) classification trees. *Evolutionary intelligence*, *15*(3), 1545-1569.

Mishra, S., Bandi, S., Komandla, V., & Konidala, S. (2024). Building more efficient AI models through unsupervised representation learning. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *5*(3), 109-120.

Mohamed, N. (2025). Artificial intelligence and machine learning in cybersecurity: a deep dive into state-of-the-art techniques and future paradigms. *Knowledge and Information Systems*, 1-87.

Mohammed, A., Manoharan, A. K., Chelliah, P. R., & Kassim, S. I. (2024). Cultivating a Security-Conscious Smart Manufacturing Workforce: A Comprehensive Approach to Workforce Training and Awareness. In *Artificial Intelligence Solutions for Cyber-Physical Systems* (pp. 385-403). Auerbach Publications.

Monaco, E., Rautela, M., Gopalakrishnan, S., & Ricci, F. (2024). Machine learning algorithms for delaminations detection on composites panels by wave propagation signals analysis: Review, experiences and results. *Progress in Aerospace Sciences*, *146*, 100994.

Nabi, F., & Zhou, X. (2024). Enhancing intrusion detection systems through dimensionality reduction: A comparative study of machine learning techniques for cyber security. *Cyber Security and Applications*, *2*, 100033.

Nifakos, S., Chandramouli, K., & Stathakarou, N. (2024). Social Engineering: The Human Behavior Impact in Cyber Security Within Critical Information Infrastructures. In *Security and Privacy in Smart Environments* (pp. 173-184). Cham: Springer Nature Switzerland.

Oluwatoyin J.A.and Akinola S. (2024) Real Time Credit Card Fraud Detection and Reporting System Using MachineLearning, European Journal of Computer Science and Information Technology, 12 (4),36-56

Omarov, B., Abdinurova, N., & Abdulkhamidov, Z. (2023). A Novel Framework for Detecting Network Intrusions Based on Machine Learning Methods. *International Journal of Advanced Computer Science and Applications*, *14*(7).

Oyadeyi, O. O., Oyadeyi, O. A., & Bello, R. O. (2024). Cybercrime in the Asia-Pacific Region: A Case Study of Commonwealth APAC Countries. *Commonwealth Cybercrime Journal*, *2*, 130-160.

Panda, S. P. (2025). The Evolution and Defense Against Social Engineering and Phishing Attacks. *International Journal of Science and Research (IJSR)*.

Patsakis, C., Arroyo, D., & Casino, F. (2024). The malware as a service ecosystem. In *Malware: Handbook of Prevention and Detection* (pp. 371-394). Cham: Springer Nature Switzerland.

Pinjarkar, L., Hete, P. R., Mattada, M., Nejakar, S., Agrawal, P., & Kaur, G. (2024, July). An Examination of Prevalent Online Scams: Phishing Attacks, Banking Frauds, and E-Commerce Deceptions. In *2024 Second International Conference on Advances in Information Technology (ICAIT)* (Vol. 1, pp. 1-6). IEEE.

Rajendran, R. K., & Tulasi, B. (2025). Natural Language Processing (NLP) for Threat Intelligence. In *Analyzing Privacy and Security Difficulties in Social Media: New Challenges and Solutions* (pp. 247-262). IGI Global Scientific Publishing.

Ravichandran, N., Tewaraja, T., Rajasegaran, V., Kumar, S. S., Gunasekar, S. K. L., & Sindiramutty, S. R. (2024). Comprehensive Review Analysis and Countermeasures for Cybersecurity Threats: DDoS, Ransomware, and Trojan Horse Attacks. *Preprints. org*.

Sannigrahi, M., & Thandeeswaran, R. (2024). Predictive analysis of network based attacks by hybrid machine learning algorithms utilizing Bayesian optimization, logistic regression and random forest algorithm. *IEEE Access*.

Sayed, M. A., Sarker, M. S. U., Al Mamun, A., Nabi, N., Mahmud, F., Alam, M. K., ... & Choudhury, M. Z. M. E. (2024). Comparative analysis of machine learning algorithms for predicting cybersecurity attack success: A performance evaluation. *The American Journal of Engineering and Technology*, *6*(09), 81-91.

Simé, V., Tchakounté, F., Yenké, B. O., Danga, D. E. H., Ngoran, M. D., & Fendji, J. L. K. E. (2024). Emoti-Shing: Detecting Vishing Attacks by Learning Emotion Dynamics through Hidden Markov Models. *Journal of Intelligent Learning Systems and Applications*, *16*(3), 274-315.

Sowmya, T., & Anita, E. M. (2023). A comprehensive review of AI based intrusion detection system. *Measurement: Sensors*, *28*, 100827.

Sreelakshmi, S., Babu, A. A., Lakshmipriya, C., Gracious, L. A., Nalini, M., & Subramanian, R. S. (2024, October). Enhancing intrusion detection systems with machine learning. In *2024 2nd International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)* (pp. 557-564). IEEE.

Styles, C. (2025). *The Rise of Mobile Malware: Challenges in Securing Mobile Banking Applications in Metropolitan Atlanta's Financial Services Sector* (Doctoral dissertation, National University).

Subramanian N. Spyware. InEncyclopedia of Cryptography, Security and Privacy 2025 May 10 (pp. 2508-2511). Cham: Springer Nature Switzerland.

Ţălu, M. (2025). Cyberattacks and Cybersecurity: Concepts, Current Challenges, and Future Research Directions. *Digital Technologies Research and Applications*, *4*(1), 44-60.

Tan, R., Saputri, U., Xiao, J., Liu, J., & Ekeh, D. (2024). A Closer look at the Famous Ransomware Groups. In *Ransomware Evolution* (pp. 18-29). CRC Press.

Triantafyllou, G. P. (2024). *Malware analysis* (Master's thesis, Πανεπιστήμιο Πειραιώς).

Vanhove, A. J., Graham, B. Z., Titareva, T., & Udomvisawakul, A. (2025). Classification Performance of Supervised Machine Learning to Predict Human Resource Management Outcomes: A Meta-Analysis Using Cross-Classified Multilevel Modeling. *Human Resource Management*.

Villegas-Ch, W., Govea, J., Gutierrez, R., Navarro, A. M., & Mera-Navarrete, A. (2024). Effectiveness of an Adaptive Deep Learning-Based Intrusion Detection System. *IEEE Access*.

Wang, S., Wu, R., Jia, S., Diakite, A., Li, C., Liu, Q., ... & Ying, L. (2024). Knowledge-driven deep learning for fast MR imaging: Undersampled MR image reconstruction from supervised to un-supervised learning. *Magnetic Resonance in Medicine*, *92*(2), 496-518.

Yadav, B. R. (2024). Machine learning algorithms: optimizing efficiency in AI applications. *International Journal of Engineering and Management Research*, *14*(5), 49-57.

Yu, J., Shvetsov, A. V., & Alsamhi, S. H. (2024). Leveraging machine learning for cybersecurity resilience in industry 4.0: Challenges and future directions. *IEEE access*.