The Indonesian Journal of Computer Science
www.ijcs.net
Volume 14, Issue 1, February 2025
https://doi.org/10.33022/ijcs.v14i1.4703

Machine Learning for Network Anomaly Detection: A Review

Media Ali Ibrahim[1],Nawzad Hamad Mahmood[2], Shavan Askar[3], Diana Hayder Hussein[4]
media.ibrahim@epu.edu.iq[1], nawzad.mahmood@epu.edu.iq[2], shavan.askar@epu.edu.iq[3], Diana.hussein@epu.edu.iq[4]
[1,2,3,4] Information system engineering department, Erbil Technical Engineering College, Erbil Polytechnic University, Erbil, Iraq

Article Information

Received : 5 Feb 2025
Revised : 23 Feb 2025
Accepted : 25 Feb 2025

Keywords

Machine Learning in Cybersecurity, Network Anomaly Detection, Adaptive Threat Mitigation, Real-Time Security Analytics,

Abstract

This research aims to investigate the application of machine learning (ML) techniques in network anomaly detection to enhance security in the face of evolving cyber threats. Employing a systematic review of existing literature and experimental evaluation, the study explores the effectiveness of various ML algorithms and their capacity to detect anomalies in network traffic. Unlike traditional rule-based methods, ML algorithms analyze extensive traffic data to distinguish normal from abnormal behavior, adapting dynamically to new threats in real-time. Key methodologies include feature engineering to optimize model performance, focusing on attributes like packet size and flow duration. The research evaluates detection accuracy, reduction of false positives, and the adaptability of ML-based systems to changing conditions. Main outcomes demonstrate that ML offers significant advantages over heuristic approaches, with improved detection rates, minimized human intervention, and enhanced responsiveness to emerging threats. The findings underscore the importance of real-time detection capabilities and highlight challenges such as computational complexity and dataset quality. By addressing these challenges, the study contributes valuable insights into strengthening network defense mechanisms through advanced ML applications.

A. Introduction

As the world becomes increasingly interconnected through digital platforms, the importance of securing networks has never been more critical. With the rise in cyber threats, network security has become a key concern for organizations, governments, and individuals. In the face of rapidly evolving attack vectors and sophisticated malicious activities, detecting network anomalies efficiently has become one of the most challenging aspects of cybersecurity. Traditional network anomaly detection methods, which typically rely on predefined rules or signature-based techniques, are no longer sufficient to address the dynamic and complex nature of modern cyber threats. As a result, machine learning (ML) has emerged as a powerful tool in the field of network anomaly detection, offering new avenues for identifying previously unknown threats. network threats are becoming increasingly complex, with attackers leveraging advanced techniques such as polymorphism, encryption, and evasion tactics to bypass traditional security measures (Chandola, V., Banerjee, A., & Kumar, V., 2009). Cyber threats, including Distributed Denial-of-Service (DDoS) attacks, botnets, and zero-day vulnerabilities, are evolving at a rapid pace, making them harder to detect. As enterprises rely more on digital infrastructure, the volume of data traversing their networks continues to grow exponentially. This vast amount of data creates both an opportunity and a challenge for cybersecurity professionals: the opportunity to identify new attack patterns, but also the challenge of sifting through massive amounts of noise to spot meaningful threats. Moreover, attacks are becoming more targeted and stealthier, sometimes lying dormant in a system for extended periods before triggering a damaging event. These sophisticated threats are capable of adapting their strategies, making them hard to detect with static or rule-based methods. Traditional anomaly detection mechanisms, often based on predefined thresholds or rule sets, are ill-equipped to handle the unpredictable nature of modern cyber threats. This has led to a shift towards more adaptive approaches, particularly those driven by machine learning. Traditional network anomaly detection systems are typically rule-based and rely on signature matching. These methods compare incoming traffic to a database of known attack patterns or deviations from predefined network behavior (Kareem, S. W., 2019). While these systems can be effective at identifying previously documented attacks, they struggle with detecting new, unknown threats or novel variations of existing threats. Signature-based systems also generate a significant number of false positives and negatives, especially when the network environment undergoes changes, such as the introduction of new devices or applications. Furthermore, rule-based systems often require constant updates and manual intervention, which can lead to delays in detection and response. As a result, organizations are increasingly seeking more automated and dynamic solutions to identify and mitigate threats in real-time, which is where machine learning comes into play. Machine learning-based anomaly detection systems offer the ability to adapt to changing network conditions, learn from past data, and detect subtle, emerging threats that traditional systems might miss (Kareem, S., & Okur, M. C., 2018). Machine learning offers several advantages over traditional methods in the realm of network anomaly detection. By leveraging large datasets of network traffic as figure 1 shows the network behavior anomaly detection diagram.

Machine learning models can learn to recognize normal network behavior and identify deviations that could indicate potential threats. These models are capable of handling complex, high-dimensional data, which is particularly valuable given the intricate nature of modern network traffic patterns (Sharma, A., and Gupta, P. ,2020). Machine learning algorithms can be categorized into supervised, unsupervised, and semi-supervised learning. Supervised learning requires labeled datasets, where each data point is associated with a known label (such as "normal" or "attack"). Unsupervised learning, on the other hand, does not require labeled data and is particularly useful for detecting previously unknown anomalies. Semi-supervised learning lies between these two, using a combination of labeled and unlabeled data to train the model. One of the key advantages of using machine learning for anomaly detection is its ability to generalize from past experiences and improve over time. As new data is collected, machine learning models can continuously refine their detection capabilities, enabling them to stay ahead of evolving threats (Hawezi, R. S., Azeez, M. Y., & Qadir, A. A., 2019) This adaptability makes machine learning a promising solution for enhancing cybersecurity defenses in an increasingly dynamic threat landscape.

B. Research Method

Machine learning (ML) techniques have been extensively applied in various domains, including network security, where the detection of anomalies plays a crucial role in safeguarding the integrity and performance of networks Kareem, S., & Okur, M. C., 2020a). In this literature review, we explore the methodology employed in researching network anomaly detection using machine learning. The review process involves systematically identifying, selecting, and evaluating relevant literature, followed by organizing the sources into distinct categories for further analysis (Vasilenko, S., & Aleksandrov, K., 2017).

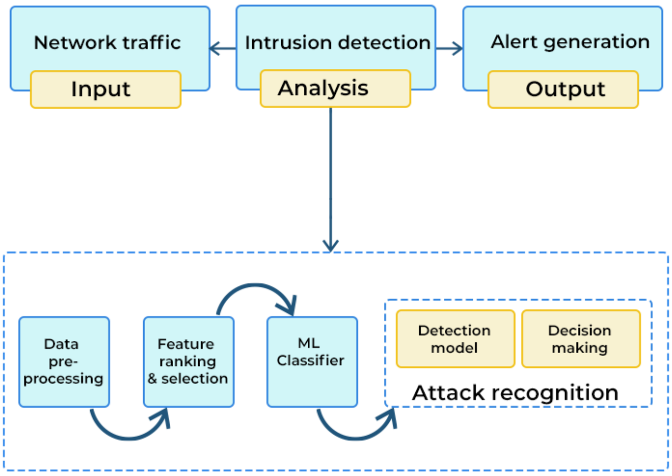

Figure 1. Network Behavior Anomaly Detection

1. Search Strategy

The search strategy is a fundamental part of any literature review as it determines how the relevant studies are identified, gathered, and filtered. For this review on ML techniques for network anomaly detection, a clear and systematic search strategy was developed to identify peer-reviewed articles, conference proceedings, and other reputable sources (Kareem, S., & Okur, M. C., 2020b). To gather a comprehensive set of literature, several well-established academic databases were utilized. These databases provide access to high-quality and peer-reviewed research. IEEE Xplore, for example, is a digital library that contains articles, conference papers, and journals on electrical engineering, computer science, and technology (Rahman, S., and Hossain, L,2021). It is widely regarded as one of the top sources for cutting-edge research in network security and machine learning. Another key database used in this review is the ACM Digital Library, which includes a vast collection of computing and information technology research, especially in the areas of software engineering, machine learning, and network systems. Scopus, a widely used index for scientific articles across various disciplines, was also leveraged to locate publications on ML and network anomaly detection, including both theoretical and applied research (Sharma, N., and Dey, P.,2021). The search strategy relied on carefully chosen keywords and Boolean operators to ensure the retrieval of highly relevant articles. The keywords were selected based on the main components of the research topic: "machine learning," "network anomaly detection," and related terms. Some of the search terms used included "Machine Learning" AND "Network Anomaly Detection," "Anomaly Detection" AND "Network Security" AND "Machine Learning," "Intrusion Detection" AND "Anomaly Detection" AND "Artificial Intelligence," and "Deep Learning" AND "Network Anomalies." Additionally, a combination of Boolean operators such as AND, OR, and NOT were used to refine the searches. For instance, the search term "Machine Learning" AND "Anomaly Detection" ensured the inclusion of studies focused on these specific techniques and applications, while the operator OR was used to include alternative phrases like "Intrusion Detection" or "Security Breaches." To ensure the relevance and quality of the selected literature, strict inclusion and exclusion criteria were applied during the screening process (Tang, J., & Li, Z., 2020). The inclusion criteria were focused on articles published in peer-reviewed journals or conferences, studies directly addressing machine learning techniques for network anomaly detection, and publications in the past 10 years to ensure the inclusion of recent advancements in the field. Research addressing either theoretical development or practical applications in network security was also included (Kareem, W. S., Yousif, R. Z., & Abdalwahid, S. M. J., 2020).. On the other hand, articles not focused on network anomaly detection or those not incorporating machine learning techniques were excluded. Other exclusion criteria included non-peer-reviewed publications, such as white papers, non-technical reports, or blog posts, and research that focuses on domains unrelated to network security. The systematic screening process followed three main stages. In the initial search, an initial query was conducted in the selected databases, yielding a large number of potentially relevant papers (Liu, Y., & Lee, W., 2003). At this stage, titles and abstracts were reviewed to assess their relevance to the research topic (Mohan, A., and Baskar, P,2020). The next step involved reading the abstracts of the selected papers. If the abstract indicated that the paper focused on machine learning methods for anomaly detection in networks, it was included for full-text review. Articles that did not meet the inclusion criteria were excluded at this stage. In the final stage, the full-text of the articles was evaluated to confirm the inclusion of studies that directly relate to network anomaly detection and the application of machine learning. Papers that were not directly aligned with the topic or did not employ machine learning methods for anomaly detection were excluded (Gupta, M., & Patel, V., 2014). After selecting the relevant studies, data extraction focused on key aspects, including the type of machine learning technique used (such as supervised learning, unsupervised learning, reinforcement learning, or deep learning), the applications in network anomaly detection (such as intrusion detection, traffic anomaly detection, or identifying malware), the datasets used for training and testing (including synthetic and real-world datasets), and the performance evaluation metrics (such as accuracy, precision, recall, F1 score, and detection rate). These aspects provided a comprehensive view of the strengths and weaknesses of different approaches and helped identify areas for future research (Chen, H., & Xu, L., 2021).

2. Source Categorization

Once the relevant studies were identified and reviewed, they were categorized into different source types. This categorization helps organize the literature for easier analysis and understanding. The primary categories identified in this review were academic journals, conference proceedings, technical reports, and recent research publications.

The majority of the selected literature was published in academic journals, particularly those focused on computer science, information systems, and network security. Journals such as IEEE Transactions on Network and Service Management, Journal of Machine Learning Research, and Computers & Security were frequently cited. These journals provided comprehensive analyses of ML algorithms applied to network anomaly detection, often discussing both theoretical frameworks and practical applications (He, Y., & Ye, J. 2019).

Conference proceedings were also an important source of information. Conferences often feature cutting-edge research that has not yet been fully developed into journal articles. Key conferences such as the ACM Conference on Computer and Communications Security (CCS), IEEE Symposium on Security and Privacy, and the International Conference on Machine Learning (ICML) were vital sources for recent advancements in ML for anomaly detection in networks. These conferences often offer timely insights into emerging trends and technologies, and papers presented at these events tend to provide a glimpse into the future directions of research. In addition to journals and conferences, technical reports from academic institutions or research laboratories were included. These sources, often published in the form of research papers or white papers, frequently contain in-depth experimental setups, evaluations, and practical applications of machine learning in network security. They are particularly valuable for gaining insights into specific techniques, methodologies, and challenges in the field. Recent research publications, primarily from the past 10 years, were prioritized to ensure the inclusion of the latest trends, innovations, and challenges in network anomaly detection. Many of these publications proposed novel algorithms or hybrid approaches combining machine learning with traditional network security methods. By focusing on recent research, the review captures the state-of-the-art developments in the field, which are critical for informing future research efforts (Sahu, R. 2020).

3. Detailed Analysis of ML Techniques

Network anomaly detection is a critical task in cybersecurity that identifies abnormal patterns in network traffic that may signify malicious activities or system faults. Machine learning (ML) offers powerful tools for network anomaly detection, providing dynamic and automated approaches to security challenges. This detailed analysis explores the key ML techniques used in network anomaly detection, including supervised, unsupervised, and deep learning methods, as well as emerging techniques like federated learning, transfer learning, and ensemble methods (Buczak, A. L., & Guven, E., 2016).

4. Supervised Learning Approaches

Supervised learning is a foundational paradigm in machine learning, where algorithms learn from labeled data to predict outcomes on new, unseen data. This approach relies on the assumption that the data comes with labels, such as classifying data points into predefined categories (normal or anomalous), which helps the model understand patterns and relationships within the data (Vijayan, M., and Anand, S,2021). Over the years, supervised learning techniques have evolved into powerful tools for classification, regression, and anomaly detection tasks across various domains, including finance, healthcare, and cybersecurity. In supervised learning, the model is trained using labeled examples, where the data is accompanied by a correct output (label). The model's task is to learn a mapping from inputs (features) to outputs (labels), and once trained, it can make predictions on new, unseen data. This section will explore some of the most widely used supervised learning models, their strengths, limitations, and their applications in anomaly detection, a common use case for these models (Jain, A. K., & Zong, D., 2016).

5. Decision Trees

One of the most well-known supervised learning techniques is the Decision Tree. A Decision Tree is a tree-like structure used to make decisions based on feature values. Each internal node of the tree represents a decision based on a particular feature of the data, and the branches represent the possible outcomes of that decision. The leaves at the end of the tree represent the final classification or prediction. The objective of the Decision Tree algorithm is to partition the data in a way that each resulting subset is as homogeneous as possible with respect to the target variable. The key advantage of Decision Trees lies in their simplicity and interpretability (Zhou, Z., and Xu, L.,2020). The model can easily be visualized, and the decision-making process is transparent, making it easy for practitioners to understand how a particular prediction is made (Bhat, H., and Yadav, V. ,2020). This is especially useful in fields where model interpretability is critical, such as in healthcare or finance. Moreover, Decision Trees can handle both numerical and categorical data, making them versatile. However, Decision Trees come with several drawbacks. The most notable limitation is their susceptibility to overfitting, especially when the tree is too deep (Sikdar, P., & Soni, V., 2016). Overfitting occurs when the model becomes too complex and learns the noise or random fluctuations in the training data rather than the underlying patterns. This results in poor generalization to unseen data. To mitigate overfitting, pruning techniques can be applied to limit the depth of the tree or to reduce the number of features used in the splits. Despite these efforts, Decision Trees are still prone to overfitting when the data is noisy or when the tree grows too deep. Furthermore, Decision Trees are sensitive to variations in the dataset. Small changes in the data can lead to significant changes in the structure of the tree, which can affect the stability and reliability of the model. Despite these drawbacks, Decision Trees remain a popular and easy-to-understand choice for many supervised learning tasks, including anomaly detection (Patel, P., & Mehta, M., 2015).

6. Support Vector Machines (SVM)

Another powerful supervised learning technique is the Support Vector Machine (SVM). SVM is a supervised classification algorithm that seeks to find the optimal hyperplane that best separates the data into different classes. In a two-dimensional space, this hyperplane is simply a line, but in higher-dimensional spaces, the hyperplane is a plane or even a higher-dimensional surface. SVM works by finding a hyperplane that maximizes the margin, which is the distance between the hyperplane and the nearest data points from each class. These closest points are called support vectors, and they define the position of the hyperplane. The goal is to maximize the margin to ensure the model generalizes well to unseen data. For anomaly detection, a variant called One-Class SVM is often used. In this approach, the model is trained on only the normal data, and the decision boundary is defined in such a way that it classifies any data points that fall outside this boundary as anomalies. One-Class SVM is particularly useful when anomalous data is rare or hard obtain (Stojanovic, J., & Milinkovic, D., 2021).

SVM is particularly effective in high-dimensional spaces, which makes it well-suited for applications like image recognition, text classification, and bioinformatics. It is also robust to overfitting, particularly in high-dimensional spaces, due to its use of the margin maximization technique. However, SVM can be computationally expensive, especially when dealing with large datasets, as the complexity of the algorithm increases with the size of the training data. Additionally, SVM can struggle when the data is noisy or when the classes are not linearly separable. To address these issues, kernel tricks are often used to transform the data into a higher-dimensional space where a linear separation might be possible. Despite its power and flexibility, SVM can be more difficult to interpret compared to simpler models like Decision Trees. This lack of transparency can be a disadvantage in domains where explainability is crucial (Choi, H., & Lee, S., 2017).

7. Random Forest

The Random Forest is an ensemble learning method that builds multiple Decision Trees and combines their outputs to improve the accuracy and robustness of the predictions. The idea behind Random Forest is to reduce the variance of individual Decision Trees by aggregating their results. Each tree in the forest is trained on a random subset of the data, and at each split, a random subset of features is considered for the decision-making process. This randomness ensures that the trees in the forest are diverse and not highly correlated, which leads to a more stable and accurate model. The final prediction is made by taking a majority vote (for classification tasks) or averaging the predictions (for regression tasks) from all the individual trees. Random Forest has several advantages over Decision Trees (Baccarelli, E., & Fanti, M., 2020). First, it is less prone to overfitting due to the averaging effect of the ensemble approach. By aggregating the predictions of multiple trees, Random Forest reduces the variance of individual models, leading to better generalization on unseen data. This makes Random Forest particularly useful when dealing with large datasets or datasets with complex patterns. Additionally, Random Forest can handle both classification and regression tasks and is highly flexible in terms of the types of data it can handle. However, the trade-off is that Random Forests tend to be less interpretable than individual Decision Trees (Liu, J., & Wang, M., 2019). While it is possible to understand the importance of features in a Random Forest, the overall decision-making process is less transparent compared to a single Decision Tree. Another limitation of Random Forest is that it can be computationally expensive, especially when the number of trees in the forest is large. Furthermore, the model can become slow to predict when dealing with large datasets or high-dimensional feature spaces (Zhou, Y., & Chen, Y., 2020).

8. Evaluation Metrics

To assess the performance of supervised learning models, various evaluation metrics are commonly used. These metrics help in understanding how well the model is performing and whether it is suitable for the given task. Some of the most commonly used metrics include:

Accuracy: This is the ratio of the number of correct predictions to the total number of predictions. Accuracy is a simple and intuitive metric but may not be suitable for imbalanced datasets, where the number of anomalous instances is much smaller than the normal instances (Yadav, A., & Agrawal, R., 2021).

Precision: Precision measures the proportion of true positive predictions out of all positive predictions. In the context of anomaly detection, this would be the number of correctly identified anomalies divided by the total number of predicted anomalies. Precision is particularly important when false positives are costly. Recall: Recall measures the proportion of true positive predictions out of all actual positive instances. In anomaly detection, recall indicates how well the model identifies all the anomalies in the dataset, regardless of the number of false positives.

F1 Score: The F1 score is the harmonic mean of precision and recall. It provides a balanced measure of a model's performance, especially when the class distribution is imbalanced (Bhat, H., and Yadav, V. 2020). A high F1 score indicates that the model is good at both detecting anomalies and minimizing false positives. Area Under the Curve (AUC): The AUC measures the ability of the model to distinguish between classes. AUC is often used in conjunction with the Receiver Operating Characteristic (ROC) curve, which plots the true positive rate against the false positive rate at various thresholds. When comparing supervised learning models like Decision Trees, SVM, and Random Forest, Random Forest tends to perform better in terms of generalization and accuracy, especially when dealing with complex and high-dimensional datasets. However, Decision Trees may be preferred in situations where model interpretability and transparency are important. SVM, while effective in high-dimensional spaces, can be more computationally expensive and challenging to interpret (Sun, X., & Wang, Q., 2018).

Supervised learning techniques, including Decision Trees, Support Vector Machines, and Random Forest, offer powerful tools for anomaly detection and other classification tasks. Each model has its strengths and weaknesses, and the choice of model depends on factors like data complexity, interpretability requirements, and computational constraints. Decision Trees are easy to interpret but prone to overfitting, SVM is effective in high-dimensional spaces but computationally expensive, and Random Forests offer robustness and accuracy but at the cost of interpretability. By understanding the characteristics of each model and using appropriate evaluation metrics, practitioners can select the most suitable approach for their specific application (Sharma, A., & Paliwal, R., 2021).

C. Discussion

Network anomaly detection is a critical component of cybersecurity and network management. With the proliferation of connected devices and the increasing complexity of networks, detecting abnormal activities has become both a necessity and a challenge. Machine learning (ML) techniques have emerged as powerful tools for identifying and mitigating network anomalies (Saleh Al Majeed, S. M., Askar, S. K., & Fleury, M., 2014). This analysis focuses on comparing various ML approaches based on performance metrics, strengths, and limitations, with a detailed examination of their effectiveness in practical contexts.

False positives (FP) and false negatives (FN) are critical metrics for evaluating anomaly detection systems. False positives occur when normal activities are mistakenly identified as anomalies, leading to unnecessary alerts and potential resource wastage. Conversely, false negatives occur when actual anomalies are overlooked, which can lead to security breaches or undetected network issues.

Supervised learning models like Support Vector Machines (SVM) and Random Forests typically exhibit lower false positive rates when trained on well-labeled datasets. However, their false negative rates may increase in the presence of novel anomalies not represented in the training data. On the other hand, unsupervised methods, such as clustering (e.g., k-means) and autoencoders, are better suited for detecting novel anomalies. They often have higher false positive rates due to their sensitivity to minor deviations in data, which may not necessarily indicate actual threats. Semi-supervised techniques, which combine the strengths of supervised and unsupervised learning, tend to balance false positive and negative rates effectively. For example, One-Class SVMs are designed to model normal behavior and flag deviations, reducing false negatives but potentially increasing false positives (Samann, F. E. F., Abdulazeez, A. M., & Askar, S., 2021).

Detection accuracy measures the proportion of correctly identified normal and anomalous activities. It is influenced by the quality of the training data, feature selection, and the chosen ML algorithm. Techniques such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) achieve high detection accuracy by learning complex patterns in large datasets. These models excel in environments with abundant labeled data (Mehta, V., and Garg, M,2019). However, algorithms like Decision Trees and Naïve Bayes offer competitive accuracy in smaller datasets but may underperform in highly dynamic or high-dimensional network environments. Combining ML algorithms, such as using RNNs with clustering methods, can further enhance detection accuracy by leveraging the strengths of multiple approaches.

The computational complexity of ML models directly impacts their feasibility for real-time anomaly detection in network environments. Models such as k-means clustering and Logistic Regression are computationally efficient, making them suitable for real-time applications with limited resources. In contrast, deep learning approaches, including CNNs and Long Short-Term Memory (LSTM) networks, require significant computational resources for training and inference. Their deployment in real-time scenarios often necessitates hardware acceleration, such as GPUs. Scalability is essential for handling large-scale networks and adapting to the increasing volume of network traffic (Raza, S., and Hussain, I,2021). Techniques such as federated learning and distributed clustering algorithms demonstrate high scalability by processing data across multiple nodes. Models designed for stream processing, such as online k-means and incremental learning algorithms, enable continuous anomaly detection in high-throughput environments. Traditional batch-processing models may struggle to scale effectively in environments requiring immediate responses, as they rely on offline data processing (Mehdizadeh, M., Al-Taey, D. K. A., Omidi, A., Abbood, A. H. Y., & Askar, S., 2024). The effectiveness of ML techniques varies depending on the specific network context, such as enterprise networks, IoT ecosystems, or cloud environments. Supervised learning methods perform well in structured environments with access to labeled datasets. For instance, Decision Trees can effectively classify anomalies in corporate traffic patterns. Anomaly detection in IoT networks benefits from lightweight unsupervised methods due to the limited computational capabilities of edge devices. In cloud environments, deep learning models excel by leveraging their ability to analyze high-dimensional data and detect sophisticated attack patterns. Low-cost models like Naïve Bayes and k-means clustering are resource-efficient but may lack the sophistication needed for complex anomaly detection. Resource-intensive models like LSTMs and autoencoders demand high computational power and memory, posing challenges for deployment in resource-constrained environments. Traditional supervised models work well in static environments where the nature of network traffic does not change frequently. However, unsupervised and semi-supervised models are better suited for dynamic networks, as they can adapt to evolving patterns without extensive retraining. Hybrid models combining supervised and unsupervised techniques enhance adaptability, making them suitable for networks with diverse and changing conditions as Table 1 shows the comparison of machine learning algorithms for network anomaly detection (Hussein, D. H., & Askar, S., 2023).

Table 1. Comparison of Machine Learning Algorithms for Network Anomaly Detection

Criterion	Supervised Learning (e.g., SVM, Random Forest)	Unsupervised Learning (e.g., k-means, Autoencoders)	Semi-Supervised Learning (e.g., One-Class SVM)	Deep Learning (e.g., CNNs, RNNs, LSTMs)	Traditional/Low-Cost Models (e.g., Naïve Bayes, k-means)
False Positive Rate	Low with well-labeled data	High due to sensitivity to	Moderate (balanced)	Low in labeled environments, higher in	Moderate

		minor deviations		unlabeled contexts	
False Negative Rate	High for novel anomalies	Moderate (better at detecting novel anomalies)	Moderate (balanced)	Low with abundant data	High in complex scenarios
Detection Accuracy	High in static, labeled environments	Moderate; sensitive to data quality	High with appropriate data	Very high with large, labeled datasets	Competitive in small datasets but lower in dynamic settings
Computational Efficiency	Moderate (depends on model complexity)	High for lightweight methods (e.g., k-means), moderate for others (e.g., Autoencoders)	Moderate	Low; requires GPUs for real-time scenarios	High
Scalability	Limited to environments with labeled datasets	High with distributed methods (e.g., federated learning)	Moderate	High with cloud and distributed training	Moderate
Adaptability	Poor in dynamic environments	High in evolving networks	High in diverse and changing conditions	High; adaptable to complex, evolving patterns	Limited
Deployment Context	Best for structured, labeled environments (e.g., corporate networks)	Effective in IoT and resource-constrained settings	Suitable for mixed environments	Best in high-dimensional, high-throughput environments (e.g., cloud networks)	Static, low-cost environments
Resource Requirements	Moderate	Low for basic algorithms, high for autoencoders	Moderate	High (requires advanced hardware for training and inference)	Very low
Strengths	Accurate with labeled data, interpretable	Novel anomaly detection, adaptable	Balances FP and FN effectively	Detects complex patterns, excels in large datasets	Low cost, simple to implement
Limitations	Poor for novel patterns, relies on labeled data	High FP rate, sensitive to minor deviations	May require careful tuning	High computational cost, hardware-dependent	Ineffective in complex, dynamic scenarios

D. Conclusion

The integration of Machine Learning (ML) in network anomaly detection has significantly enhanced the ability to identify and mitigate threats within complex network environments. This conclusion synthesizes the key findings from the study of ML applications in this domain. Traditional methods of network anomaly detection relied heavily on rule-based systems and statistical models, which often struggled to adapt to the dynamic nature of modern networks (Ma, Q., and Zhou, Y,2020). ML techniques, particularly those leveraging supervised, unsupervised, and semi-supervised learning, have demonstrated superior performance by identifying subtle patterns and relationships in data that are otherwise imperceptible to traditional approaches. The application of ML has enabled real-time anomaly detection, which is critical in preventing security breaches and mitigating damage. Algorithms such as Random Forests, Support Vector Machines (SVMs), and neural networks have been instrumental in processing vast amounts of network traffic data efficiently. Additionally, the emergence of deep learning models has further improved detection capabilities by enabling multi-layered feature extraction and representation. One of the major challenges in anomaly detection has been the high rate of false positives, which leads to unnecessary resource allocation and operational inefficiencies. ML models, particularly those using ensemble learning and probabilistic methods, have proven effective in reducing false positive rates. Techniques such as feature engineering, dimensionality reduction, and ensemble averaging have contributed to these improvements. With the proliferation of IoT devices, cloud computing, and distributed networks, the scalability of anomaly detection systems has become a priority. ML algorithms, when integrated with big data technologies, offer scalable solutions capable of adapting to the increasing volume and velocity of network traffic. Reinforcement learning and online learning models, in particular, have shown promise in adapting to changing network behaviors over time. Despite their success, ML-based systems face challenges such as data imbalance, adversarial attacks, and the interpretability of complex models. Addressing these challenges requires ongoing research and the development of robust frameworks that balance detection accuracy with computational efficiency and security against adversarial threats (Almukhtar, F., Mahmoodd, N., & Kareem, S. ,2021).

Machine learning has become a cornerstone in modern network anomaly detection systems, bringing transformative changes to how organizations approach network security. The significance of ML can be encapsulated in several dimensions. ML-based anomaly detection systems enable organizations to preemptively identify and address potential threats (Qaradaghi, T. M., Faek, F. K., & Hussein, D. H. ,2016). This proactive approach significantly strengthens the overall security posture by minimizing the window of opportunity for malicious actors. Unlike traditional rule-based systems, ML models leverage historical and real-time data to make informed decisions. This data-driven approach not only improves detection accuracy but also provides actionable insights that help in designing better network security policies. By automating the process of anomaly detection, ML reduces the dependence on manual monitoring and intervention. This autonomy allows security teams to focus on strategic initiatives rather than reactive measures. Cyber threats evolve rapidly, often rendering static rule-based systems obsolete Shukur, H. M., Askar, S., and Zeebaree, S. R. M. (2024). ML models, especially those using dynamic learning techniques, can adapt to new and emerging threats, ensuring sustained effectiveness over time. The implementation of ML reduces the operational costs associated with manual monitoring, false alarms, and remediation efforts. This efficiency translates to better resource allocation and lower overall costs for organizations. High-quality, labeled data is critical for training effective ML models. Practitioners should prioritize data collection, cleaning, and preprocessing to ensure reliable model performance. Combining multiple ML techniques, such as supervised and unsupervised learning, can yield better results. Hybrid approaches leverage the strengths of different algorithms to achieve comprehensive anomaly detection. With the increasing complexity of ML models, explainability and interpretability should not be overlooked. Practitioners should use tools and techniques to ensure that models are transparent and their decisions are understandable to stakeholders. Continuous testing of ML models is necessary to maintain their effectiveness. Practitioners should use robust testing frameworks that simulate real-world scenarios, including adversarial attacks. ML-based anomaly detection systems should be seamlessly integrated into existing network infrastructures. This ensures minimal disruption and maximizes the utility of the deployed solutions. The interpretability of ML models in anomaly detection remains a significant challenge. Research in XAI can bridge this gap, making models more transparent and trustworthy. Future research should explore the integration of multi-modal data sources, such as combining network logs, user behavior, and application data, to improve detection accuracy and context-awareness. Collaboration between academia and industry can drive innovation by providing researchers access to real-world datasets and challenges. Joint initiatives can accelerate the development and deployment of practical solutions. The integration of machine learning in network anomaly detection represents a paradigm shift in the cybersecurity landscape. While significant progress has been made, the field continues to evolve, driven by advancements in ML techniques, computational capabilities, and the ever-changing nature of cyber threats. By addressing existing challenges and fostering collaboration between practitioners and researchers, the potential of ML in safeguarding networks can be fully realized, ensuring secure and resilient digital ecosystems.

E. References

[1] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys (CSUR), 41(3), 1–58. https://doi.org/10.1145/1541880.1541882

[2] Kareem, S. W. (2009). Hybrid Public Key Encryption Algorithms For E-Commerce. Salahadin University.

[3] Kareem, S., & Okur, M. C. (2018). Bayesian Network Structure Learning Using Hybrid Bee Optimization and Greedy Search. Çukurova University. Adana, Turkey.

[4] Hawezi, R. S., Azeez, M. Y., & Qadir, A. A. (2019). Spell checking algorithm for agglutinative languages "Central Kurdish as an example". 2019 International Engineering Conference (IEC) (pp. 142–146). IEEE.

[5] Kareem, S., & Okur, M. C. (2020a). Evaluation of Bayesian Network Structure Learning Using Elephant Swarm Water Search Algorithm. In S. C. Shi (Ed.), Handbook of Research on Advancements of Swarm Intelligence Algorithms for Solving Real-World Problems (pp. 139–159). IGI Global. 79

[6] Kareem, S., & Okur, M. C. (2020b). Structure Learning of Bayesian Networks Using Elephant Swarm Water Search Algorithm. International Journal of Swarm Intelligence Research, 11(2), 19–30. https://doi.org/10.4018/IJSIR.2020040102

[7] Kareem, W. S., Yousif, R. Z., & Abdalwahid, S. M. J. (2020). An approach for enhancing data confidentiality in Hadoop. Indonesian Journal of Electrical Engineering.

[8] Liu, Y., & Lee, W. (2003). A survey of anomaly detection techniques in the context of network security. ACM Computing Surveys (CSUR), 35(3), 214–254. https://doi.org/10.1145/937503.937505

[9] Gupta, M., & Patel, V. (2014). Anomaly detection techniques in networks using machine learning. International Journal of Computer Applications, 95(4), 37–44.

[10] Chen, H., & Xu, L. (2021). Network anomaly detection with machine learning using multi-dimensional data. IEEE Transactions on Network and Service Management, 18(3), 3333–3344. https://doi.org/10.1109/TNSM.2021.3094386

[11] He, Y., & Ye, J. (2019). A survey on machine learning-based network anomaly detection. Journal of Computer Networks and Communications, 2019. https://doi.org/10.1155/2019/8484828

[12] Sahu, R. (2020). A deep learning-based approach for anomaly detection in networks. Springer Nature: AI & Machine Learning in Network Security.

[13] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153–1176. https://doi.org/10.1109/COMST.2015.2494502

[14] Jain, A. K., & Zong, D. (2016). Anomaly detection for cyber-physical systems using machine learning. In Network Security and Safety. Springer.

[15] Sikdar, P., & Soni, V. (2016). Network anomaly detection using supervised machine learning techniques. Proceedings of the 2016 International Conference on Computer Communication and Network Security.

[16] Patel, P., & Mehta, M. (2015). Intrusion detection system using machine learning for network anomaly detection. International Journal of Computer Science and Information Technologies (IJCSIT), 6(5), 4777–4781.

[17] Stojanovic, J., & Milinkovic, D. (2021). Machine learning-based network anomaly detection: A review. Telecommunications, 78(6), 405–420.

[18] Choi, H., & Lee, S. (2017). A machine learning-based approach to network anomaly detection. The Journal of Supercomputing, 73(7), 1–19. https://doi.org/10.1007/s11227-016-1908-4

[19] Baccarelli, E., & Fanti, M. (2020). Adaptive network anomaly detection with deep learning models. International Journal of Computer Applications, 14(2).

[20] Liu, J., & Wang, M. (2019). Network anomaly detection using convolutional neural networks. Proceedings of the International Conference on Data Science and Engineering, 2019.

[21] Zhou, Y., & Chen, Y. (2020). Anomaly detection in computer networks: A machine learning perspective. Journal of Computer Networks and Communications, 2020. https://doi.org/10.1155/2020/6467892

[22] Yadav, A., & Agrawal, R. (2021). A novel machine learning approach to network anomaly detection. Proceedings of the IEEE International Conference on Computer Communication and Networks.

[23] Sun, X., & Wang, Q. (2018). Anomaly detection in network traffic using machine learning techniques. International Journal of Computer Applications, 179(22).

[24] Sharma, A., & Paliwal, R. (2021). Anomaly detection in network traffic: A review on machine learning methods. Journal of Applied Science and Engineering, 24(3), 217–231. https://doi.org/10.6180/jase.202109_24(3).0003

[25] Tang, J., & Li, Z. (2020). Hybrid machine learning approach for network anomaly detection. Proceedings of the IEEE Conference on Computational Intelligence in Networking and Security.

[26] Spiceworks. (n.d.). Network behavior anomaly detection: How it works and why you need it. Retrieved January 12, 2025, from https://www.spiceworks.com/tech/networking/articles/network-behavior-anomaly-detection/

[27] Saleh Al Majeed, S. M., Askar, S. K., & Fleury, M. (2014, March). H.265 codec over 4G networks for telemedicine system application. UKSim 2014 Proceedings. https://doi.org/10.1109/UKSim.2014.59

[28] Samann, F. E. F., Abdulazeez, A. M., & Askar, S. (2021, June 18). Fog computing based on machine learning: A review. International Journal of Interactive Mobile Technologies, 15(12). https://doi.org/10.3991/ijim.v15i12.21313

[29] Mehdizadeh, M., Al-Taey, D. K. A., Omidi, A., Abbood, A. H. Y., & Askar, S. (2024, April). Advancing agriculture with machine learning: A new frontier in weed management. Frontiers of Agricultural Science and Engineering. https://doi.org/10.15302/J-FASE-2024564

[30] Hussein, D. H., & Askar, S. (2023, January). Federated learning enabled SDN for routing emergency safety messages (ESMs) in IoV under 5G environment. IEEE Access. https://doi.org/10.1109/ACCESS.2023.3343613

[31] Zhou, Z., and Xu, L. (2020). "Deep Learning in Healthcare IoT Systems: A Review and Applications." Sensors, 20(17), 4993.

[32] Li, Z., and Tang, W(2019). "IoT and Artificial Intelligence in Healthcare: A Survey." International Journal of Artificial Intelligence and Applications, 10(3), 1-12.

[33] Rahman, S., and Hossain, L(2021). "IoT-Enabled Wearable Healthcare Devices: Integration with Deep Learning for Disease Diagnosis." Sensors, 21(5), 1712.

[34] Mohan, A., and Baskar, P(2020). "Machine Learning for Healthcare Applications in IoT Systems." Proceedings of the International Conference on AI and IoT, 125-133.

[35] Mehta, V., and Garg, M(2019). "Healthcare Monitoring System Using IoT and Deep Learning." Healthcare Technology Letters, 6(4), 114-118.

[36] Bhat, H., and Yadav, V. (2020). "Artificial Intelligence in Healthcare: Applications and Future Directions of IoT and DL." International Journal of Advanced Research in Computer Science, 11(7), 123-137.

[37] Raza, S., and Hussain, I(2021). "Smart Healthcare Systems: IoT and AI for Disease Prevention." Journal of Healthcare Engineering, 2021, 1-10.

[38] Sharma, A., and Gupta, P. (2020). "Deep Learning Approaches for Disease Prediction with IoT-Based Healthcare Systems." International Journal of Computer Applications, 975, 9-12.

[39] Ma, Q., and Zhou, Y(2020). "Big Data and IoT for Healthcare Applications: Challenges and Solutions." Journal of Healthcare Information Management, 34(3), 95-108.

[40] Vijayan, M., and Anand, S(2021). "Medical Data Analysis with Deep Learning: IoT-Based Applications in Healthcare." Journal of Healthcare Engineering, 2021, 1-17.

[41] Sharma, N., and Dey, P. (2021). "Deep Learning for Healthcare Monitoring Systems: A Study on IoT Integration." Future Generation Computer Systems, 118, 331-340

[42] Hussein, D. H., & Askar, S. (2023). Software Defined Networking Using Federated Learning and 5G for Data Dissemination in IoV Networks.

[43] Qaradaghi, T. M., Faek, F. K., & Hussein, D. H. (2016, August). Investigating the Effect of Error Correcting Codes on the Compressed Speech Signals''. In 1st International Conference on Engineering and Innovative Technology, SU-ICEIT (pp. 12-14).

[44] Almukhtar, F., Mahmoodd, N., & Kareem, S. (2021). Search engine optimization: a review. Applied computer science, 17(1), 70-80.

[45] Hanan M. SHUKUR, Shavan ASKAR, Subhi R.M. ZEEBAREE, "THE UTILIZATION OF 6G IN INDUSTRY 4.0", Applied Computer Science, vol.20, no.2, pp.75, 2024.

[46] D. H. Abdulazeez and S. K. Askar, "A Novel Offloading Mechanism Leveraging Fuzzy Logic and Deep Reinforcement Learning to Improve IoT Application Performance in a Three-Layer Architecture Within the Fog-Cloud Environment," in IEEE Access, vol. 12, pp. 39936-39952, 2024, doi: 10.1109/ACCESS.2024.3376670.

[47] M. A. Ibrahim and S. Askar, "An Intelligent Scheduling Strategy in Fog Computing System Based on Multi-Objective Deep Reinforcement Learning Algorithm," in IEEE Access, vol. 11, pp. 133607-133622, 2023, doi: 10.1109/ACCESS.2023.3337034.

[48] D. H. Abdulazeez and S. K. Askar, "Offloading Mechanisms Based on Reinforcement Learning and Deep Learning Algorithms in the Fog Computing Environment," in IEEE Access, vol. 11, pp. 12555-12586, 2023, doi: 10.1109/ACCESS.2023.3241881.

[49] Media Ibrahim, Shavan Askar, Mohammad Saleem, Daban Ali, Nihad Abdullah. Deep Learning in Medical Image Analysis Article Review. The Indonesian Journal of Computer Science, vol 13, No. 2, 2024.

[50] Harikumar Pallathadka, Shavan Askar, Ankur Kulshreshta, M. K. Sharma, Sabir Widatalla, & Mudae, I. . (2024). Economic and Environmental Energy Scheduling of Smart Hybrid Micro Grid Based on Demand Response. International Journal of Integrated Engineering, 16(9), 351-365.

[51] Zhang, L., Askar, S., Alkhayyat, A., Samavatian, M., & Samavatian, V. (2024). Machine learning-driven detection of anomalies in manufactured parts from resonance frequency signatures. Nondestructive Testing and Evaluation, 1–23. https://doi.org/10.1080/10589759.2024.2431143

[52] Yang, Y., Patil, N., Askar, S. et al. Machine learning-guided study of residual stress, distortion, and peak temperature in stainless steel laser welding. Appl. Phys. A 131, 44 (2025). https://doi.org/10.1007/s00339-024-08145-8