AI Enhanced Intrusion Detection and Prevention Systems (IDS/IPS)

DR. DEEPAK TOMAR[1], DR. KISMAT CHHILLAR[2], PROF. SAURABH SHRIVASTAVA[3]

[1] System Analyst, Bundelkhand University, Jhansi, Uttar Pradesh, India
[2] Assistant Professor, Dept. of Mathematical Sciences and Computer Applications, Bundelkhand University, Jhansi, Uttar Pradesh, India
[3] Professor, Dept. of Mathematical Sciences and Computer Applications, Bundelkhand University, Jhansi, Uttar Pradesh, India

Abstract- The increasing sophistication of cyber threats necessitates a move beyond traditional signature-based intrusion detection systems (IDS) toward more dynamic, data-driven approaches. This paper provides a comprehensive review of machine learning (ML) techniques for real-time network anomaly detection, a critical capability for responding to fast-moving attacks. We analyzed key ML paradigms, including supervised, unsupervised and semi-supervised learning, highlighting their trade-offs, such as the need for labeled data versus the ability to detect zero-day threats. A comparative analysis of traditional ML models (e.g., Random Forest, SVM) and deep learning (DL) architectures (e.g., CNN, LSTM, Autoencoder) reveals that DL models consistently offer superior performance in handling the high-dimensional, complex nature of modern network traffic, albeit with greater computational demands. Finally, we discuss advanced architectures and future research directions, including federated learning for its privacy-preserving and scalable nature and Explainable AI (XAI) for fostering trust and providing actionable insights to human security analysts. The paper concludes that the future of network security lies in the development of hybrid, continuously adaptive systems that balance performance, privacy and interpretability to effectively counter evolving cyber threats.

Index Terms- computer network, Anomaly Detection, Computer Networks Security, Networking.

I. INTRODUCTION

The modern world is profoundly dependent on complex network infrastructures that serve as the backbone for commerce, communication and government operations. This increasing interconnectedness, however, has made network security a paramount concern for organizations, governments, and individuals alike. As cyber threats become more sophisticated, they have evolved from simple, high-volume attacks to targeted and stealthy intrusions that can lie dormant for extended periods before triggering a damaging event. Traditional security measures, such as signature-based intrusion detection systems (IDS), have proven to be insufficient in this dynamic environment. These systems rely on predefined rules and static signatures to identify malicious traffic, making them effective against known threats but fundamentally reactive and unable to adapt to new or unknown attacks. This has created a critical "detection gap" between the speed of evolving threats and the capabilities of conventional defenses. A new approach is urgently required to strengthen network defense mechanisms and enhance responsiveness to emerging threats.

In response to these challenges, machine learning (ML) has emerged as a powerful tool in the field of network anomaly detection. Unlike traditional rule-based systems, ML and deep learning (DL) algorithms are capable of analyzing vast amounts of high-dimensional network traffic data to learn and model "normal" network behavior. By identifying deviations from this dynamically established baseline, these models can flag potential anomalies, including previously unknown attack patterns, with greater accuracy and reduced human intervention. This ability to adapt and generalize from past experiences enables them to continuously refine their detection capabilities over time, a crucial feature for staying ahead of evolving threats. The speed of modern threats necessitates a detection system with minimal latency. Fast-moving attacks, such as distributed denial-of-service (DDoS) attacks or rapidly spreading malware, can cripple services within minutes. A delay of even a few seconds in detection can result in significant damage, extended downtime, or the exfiltration of sensitive data. This is why real-time anomaly detection is not merely a beneficial feature but an essential component for ensuring network resilience and enabling a rapid, effective response.

This paper provides a comprehensive review of ML techniques for real-time network anomaly detection, focusing on the architectural and operational considerations required for practical implementation. The paper is organized as follows: Section 2 discusses about background and related work. Section 3 discusses the primary ML paradigms, outlining their strengths and weaknesses; Section 4 provides a comparative analysis of traditional ML and DL techniques; Section 5 evaluates performance metrics and the importance of benchmark datasets. This section also explores the key challenges in real-time implementation; Section 6 discusses advanced architectures and future research directions, including federated learning and explainable AI; and Section 7 synthesizes the key findings and finally section 8 provides an outlook on the future of the field.

II. RELATED WORK

The concept of network traffic anomaly detection has evolved significantly over the past decades. Initially, network monitoring was a manual process, relying on human expertise to identify and respond to unusual patterns. This evolved into basic thresholding, where predefined limits were set for network metrics like traffic volume or packet rates. While this offered some automation, it was often rigid and prone to false positives. The next major leap came with the integration of machine learning techniques.

Machine learning offered a more adaptive and robust approach by learning complex patterns from data. This was a crucial development, as modern cyber threats became more sophisticated and targeted, sometimes lying dormant for extended periods before triggering a damaging event. Machine learning models could handle the complex, high-dimensional nature of modern network traffic, distinguishing normal from abnormal behavior and adapting dynamically to new threats. This capability addressed the limitations of traditional rule-based methods that struggled against unknown, zero-day attacks. The shift from static to dynamic analysis through ML laid the groundwork for the advanced systems used today.

The field of network anomaly detection has been a subject of extensive research, particularly in the application of machine learning and deep learning. A number of studies have focused on comparative analyses of different ML techniques. For example, research has shown that deep learning models, particularly Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, generally outperform traditional ML approaches in cyber threat detection, achieving accuracy rates exceeding 98%. Specifically, LSTMs have been found to achieve the highest recall and F1-score on benchmark datasets like CICIDS2017, demonstrating their superior ability to detect complex, multi-stage intrusions.

Furthermore, researchers have explored distributed and privacy-preserving approaches like federated learning. One study proposed a federated deep learning model for network anomaly detection that keeps training data decentralized, which prevents attackers from exploiting it. The results showed that this federated approach outperformed traditional centralized models in accuracy and efficiency, highlighting its scalability and robustness for large networks.

Other research has addressed the computational and operational challenges of real-time implementation. Studies have proposed novel methods for real-time traffic analysis using hardware acceleration, such as ASIC-based approaches, to handle repetitive tasks and improve performance by as much as 9x with a single processor. The use of real-time streaming frameworks like Apache Spark has also been explored, with models trained to provide real-time alerts on network attacks by processing data on the fly. This body of work underscores the ongoing effort to not only develop more effective algorithms but also to engineer robust and scalable systems capable of real-time operation.

III. MACHINE LEARNING PARADIGMS FOR ANOMALY DETECTION

A. *Supervised Learning*

Supervised learning models are trained on labeled datasets that contain examples of both normal and anomalous traffic. The model learns patterns from this data to classify new, unseen traffic into predefined categories. Common algorithms in this category include Support Vector Machines (SVM), Decision Trees (DT) and Random Forests (RF). This approach can achieve high accuracy and precision in detecting known attack types. However, its primary limitation is the reliance on extensive, high-quality and up-to-date labeled data. In real-world environments, obtaining such data is a significant challenge, as the process of manually labeling network traffic is laborious and time-consuming. Furthermore, because these models are trained on historical data, they are fundamentally limited in their ability to detect novel or zero-day attacks that do not exist in their training set. They are highly effective for detecting and classifying known threats but can be brittle when confronted with an evolving threat landscape. Figure 1 gives an overview of supervised learning for network anomaly detection.

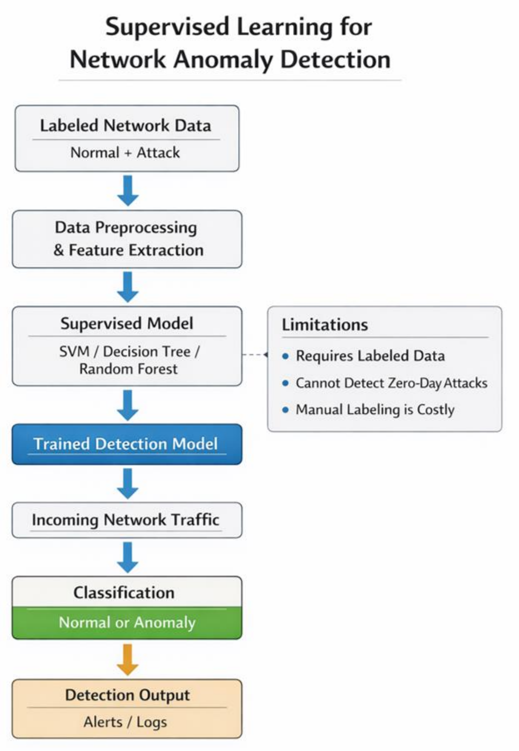


Fig. 1. Supervised Learning for Network Anomaly Detection

B. *Unsupervised Learning*

Unsupervised learning algorithms operate without the need for labeled data. Instead of learning to classify predefined categories, they learn the underlying structure of the data and model what constitutes "normal" network behavior. Any data points that significantly deviate from this learned distribution are flagged as outliers or potential anomalies. Clustering algorithms like K-means and DBSCAN are commonly used examples. Figure 2 clearly illustrates about various machine learning paradigms for network anomaly detection.

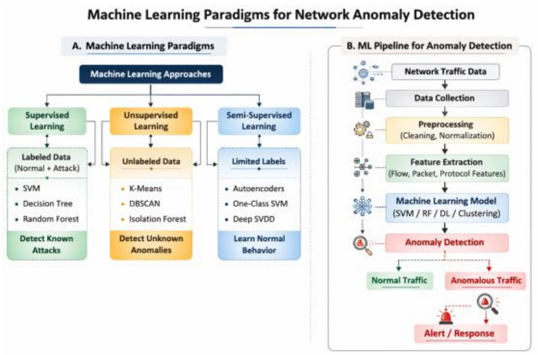


Fig. 2. Machine Learning Paradigms for Network Anomaly Detection

The utility of this paradigm is its unique ability to detect unknown, previously unseen anomalies, a crucial capability for identifying zero-day threats. Unsupervised models lack objective accuracy metrics and can be prone to higher false positive rates, as normal but unusual events (e.g., a large software update or a legitimate spike in traffic) may be incorrectly flagged as anomalies. A system that generates too many false alarms can erode the trust of security analysts and lead to alert fatigue.

C. *Semi-supervised Learning*

Semi-supervised learning offers a pragmatic middle ground between the two approaches, leveraging a small amount of labeled data to guide the detection process within a large pool of unlabeled data. A common implementation involves training a model to understand the "normal" behavior of the system using a small, labeled dataset and then applying this understanding to a large stream of unlabeled data to detect deviations. One-Class SVM and Autoencoders are prime examples of this approach. This method directly addresses the practical challenge of data labeling, improving detection accuracy compared to purely unsupervised methods while avoiding the impractical demands of fully supervised approaches. By building a more robust baseline from a small, labeled set of normal traffic, a semi-supervised model can reduce false positives while retaining the flexibility to detect novel attacks. The choice of the learning paradigm, therefore, represents a strategic decision that balances the competing requirements of precision and adaptability in a dynamic environment. Figure 3 shows different machine learning models used for anomaly detection.

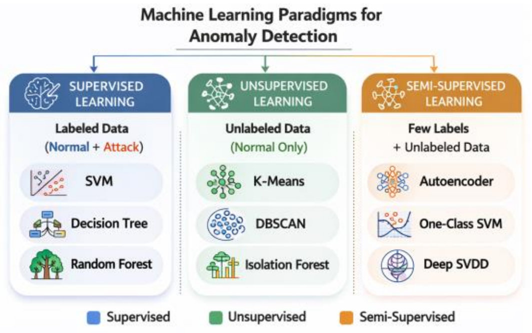


Fig. 3. Machine Learning for Anomaly Detection

IV. A COMPARATIVE ANALYSIS OF KEY TECHNIQUES

The implementation of ML for network anomaly detection involves selecting and configuring specific models, each with unique strengths and weaknesses. A comparative analysis of these techniques is essential for understanding their suitability for real-time systems.

A. Traditional Machine Learning Models

Traditional ML models, such as Random Forest (RF), Decision Trees (DT), and Support Vector Machines (SVM), have been widely used in early IDS implementations. These models offer several advantages, including faster training times, lower computational resource demands and greater interpretability compared to their deep learning counterparts. Random Forests, in particular, are known for their competitive accuracy and are well-suited for environments with limited computational resources. Studies have also shown that SVM-based algorithms can exhibit superior performance and shorter training times compared to some artificial neural network (ANN) based models. However, these models can struggle to scale with the massive volume, complexity, and high-dimensional nature of modern network traffic data. Figure 4 displays the performance of traditional ML models for IDS.

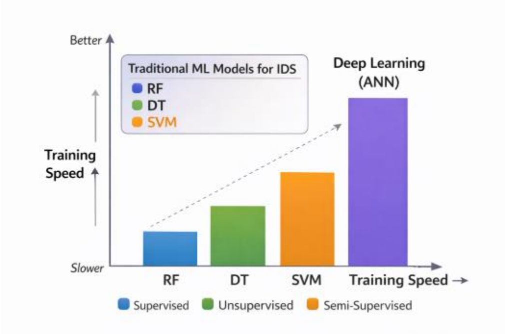


Fig. 4. Traditional ML Models for IDS

B. Deep Learning Architectures

Deep learning (DL) models, a specialized subset of ML, are characterized by multiple hidden layers that enable them to learn complex, hierarchical representations directly from raw data. This ability eliminates the need for manual feature engineering, a labor-intensive and time-consuming process. Deep learning models, particularly CNNs and LSTMs, consistently outperform traditional ML approaches in cyber threat detection, achieving accuracy rates exceeding 98% in some evaluations.

a. Convolutional Neural Networks (CNNs)

Convolutional Neural Networks, originally developed for image analysis, can be adapted for network traffic by treating traffic flows as "images" or data matrices. They excel at identifying spatial patterns and are particularly effective for detecting volumetric attacks like DDoS. The architecture, which includes convolutional and pooling layers, allows them to reduce dimensionality and extract meaningful features without losing the spatial relationships between data points, thereby improving computational efficiency.

b. Recurrent Neural Networks (RNNs) and LSTMs

Given the sequential, time-series nature of network traffic data, Recurrent Neural Networks (RNNs) and their more advanced variant, Long Short-Term Memory (LSTM) networks, are exceptionally well-suited for this domain. Figure 5 illustrates the performance of different deep learning techniques for network anomaly detection.

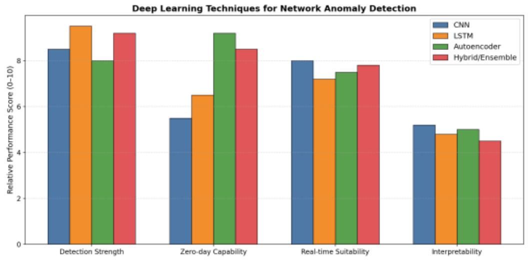


Fig. 5. Deep Learning Techniques for Network Anomaly Detection

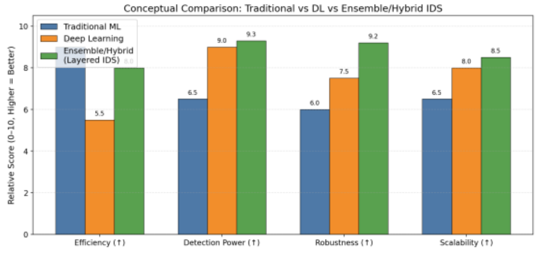
The core of an LSTM lies in its unique gating mechanisms and memory cells, which allow it to capture and retain long-term dependencies in the data without suffering from the vanishing gradient problem. This makes them ideal for detecting subtle, multi-stage attacks that unfold over time, where the sequence of events is more telling than a single data point. Studies confirm that LSTMs achieve the highest recall and F1-score and outperform other approaches on benchmark datasets like CICIDS2017. The selection between a CNN and an LSTM is not a universal choice but a strategic decision that depends on the nature of the attack to be detected. CNNs perform well on volumetric attacks, where the key pattern is spatial and related to volume, while LSTMs excel in detecting complex, multi-stage intrusions where temporal analysis is paramount.

c. Autoencoders

As an unsupervised deep learning technique, autoencoders are trained to reconstruct a data input. A model trained exclusively on a dataset of normal traffic will learn to reconstruct it with high fidelity. When presented with anomalous traffic, however, the model will struggle to reconstruct the data accurately, resulting in a high reconstruction error. This error can be used as a metric to flag potential anomalies, making autoencoders a powerful tool for detecting zero-day threats in an unsupervised manner.

d. Ensemble and Hybrid Models

To mitigate the weaknesses of individual models, researchers are increasingly exploring ensemble and hybrid approaches. Combining different models can lead to improved performance and robustness, offering a balanced approach to network security. One promising approach is a layered IDS architecture that uses a traditional ML model for initial filtering and a more complex DL model for in-depth analysis. This balance of efficiency and accuracy can create a more resilient system capable of handling a broader range of threats. The following table provides a high-level comparison of the techniques discussed. Figure 7 shows the comparison of various DL models and ensemble model for IDS.



Note: Scores are illustrative and derived from the narrative (not measured experimental results).

Fig. 7. Comparison of Traditional vs DL vs Ensemble IDS

V. PERFORMANCE EVALUATION AND BENCHMARKING

Evaluating the effectiveness of real-time anomaly detection systems is a complex task that requires careful consideration of performance metrics and the quality of benchmark datasets.

A. Evaluation Metrics

For datasets with a severe class imbalance, where normal traffic instances vastly outnumber anomalies, a simple metric like accuracy can be misleading. A model that classifies all traffic as normal could achieve an accuracy of 99% but fail to detect any anomalies. Therefore, it is essential to use a multi-metric evaluation approach that provides a more nuanced understanding of model performance.

a. Precision

This metric measures the proportion of correctly identified anomalies out of all cases the model flagged as anomalous. It quantifies false positives and is critical in scenarios where false alarms are costly, such as in industrial sensor systems or enterprise security. Figure 8 shows the precision-recall curve.

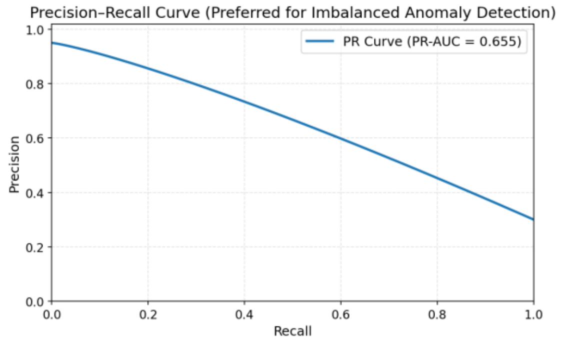


Fig. 8. Precision-Recall Curve

b. *Recall (Sensitivity)*

Recall measures the fraction of true anomalies that were successfully detected. It highlights a model's ability to find all the positive instances and is crucial in domains like network security where minimizing false negatives (missed attacks) is paramount.

c. *F1 Score*

The F1 score is the harmonic mean of precision and recall. It provides a single, balanced score that is particularly useful when dealing with class imbalance and when there is a trade-off between precision and recall. Figure 9 shows the ROC curve.

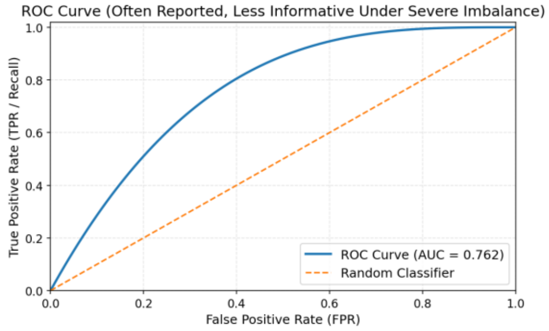


Fig. 9. ROC Curve

For a comprehensive view of model performance, especially across different classification thresholds, AUC-ROC (Area Under the Receiver Operating Characteristic curve) is widely used. However, for highly imbalanced datasets, the PR-AUC (Precision-Recall curve) is often a "better comparative visualization of model performance", as it focuses on the trade-offs most relevant to anomaly detection.

B. *Benchmark Datasets*

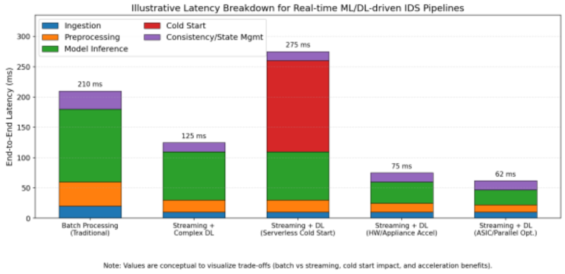
The quality of benchmark datasets is fundamental for reliable model evaluation. Historically, the KDD Cup 99 dataset was a prominent benchmark, but it is now considered outdated and problematic. It contains a massive number of redundant records that bias learning algorithms toward frequent attack types, preventing them from learning about rare but more dangerous ones. Its successor, NSL-KDD, addresses the redundancy issue but still "may not be a perfect representative of existing real networks". More recent datasets, such as UNSW-NB15 and CIC-IDS2017/2018, are considered more representative of modern network traffic and attacks. They feature a broader range of attack types (e.g., Fuzzers, DoS, Web Attacks, Botnets) and are used to evaluate state-of-the-art models. The characteristics of real-world network traffic are constantly changing due to evolving user behavior, network configurations, or new attack patterns, a phenomenon known as "concept drift". The very concept of "normal" and "anomalous" behavior is not static. This makes a model trained on a static, outdated dataset inherently flawed, as it is built to solve a problem that no longer exists in its original form. The utility of a real-time system is a continuous function of its ability to adapt to a changing environment, a property that is impossible to measure with a one-time evaluation on a static dataset.

C. *Challenges for Real-Time Implementation*

Beyond algorithmic performance, the practical deployment of real-time ML systems for network anomaly detection faces significant engineering and operational challenges that must be addressed for successful adoption.

a. *Computational Overhead and Latency*

The demand for low latency in a real-time system is non-negotiable. However, complex ML and DL models are computationally intensive, requiring significant resources and introducing latency that is unacceptable for time-sensitive applications. Traditional batch processing is fundamentally unsuited for the continuous nature of network data streams. Furthermore, a critical system design challenge is the "cold start" problem in serverless environments, where the time required to initialize a function can add significant delays for large models. Building reliable streaming pipelines also requires careful management of trade-offs between processing speed and data consistency. Figure 10 illustrates the latency breakdown for real time ML/DL driven IDS paradigms.



Note: Values are conceptual to visualize trade-offs (batch vs streaming, cold start impact, and acceleration benefits).

Fig. 10. Latency Breakdown for Real-Time ML/DL Driven IDS Pipelines

To mitigate these issues, specialized frameworks and hardware solutions are being explored. Platforms like Progress Flowmon offer hardware and virtual appliance support designed to handle millions of traffic flows per second. Research also explores ASIC-based hardware acceleration for repetitive tasks in traffic simulation and hyper-parallel optimization techniques to train CNNs, enabling high accuracy without compromising real-time performance. This highlights that the successful implementation of a real-time, ML-driven IDS is not merely an algorithmic problem but a sophisticated systems engineering challenge that requires a holistic, end-to-end approach.

b. Data Imbalance

Network traffic datasets are characterized by a severe class imbalance, where normal traffic instances vastly outnumber anomalous ones. This can cause ML models to become biased toward the majority class, leading to poor detection performance for rare but critical attacks, such as web attacks, which may be significantly underrepresented in the dataset. One of the most common techniques to address this is the Synthetic Minority Over-sampling. Technique (SMOTE), which generates synthetic samples for the underrepresented minority classes to balance the data distribution. Studies have shown that applying SMOTE can significantly improve a model's performance in detecting minority class attacks, leading to a more reliable and effective intrusion detection system.

c. The Challenge of Concept Drift

Concept drift refers to the non-stationarity of network traffic data over time, where the distribution of the data changes due to evolving user behavior, network configurations, or the emergence of new attack patterns. A model trained on past data may lose its effectiveness as the underlying patterns change, leading to performance degradation. This challenge forces a shift from a static model to a dynamic, self-adapting system. Drift can be detected by continuously monitoring model quality metrics (e.g., accuracy, F1-score) or by tracking changes in the model's predictions over time. Once detected, the model must be adapted through incremental or full retraining with new data. Approaches like the use of "sliding windows" and algorithms like the Hoeffding Tree enable models to continuously update without a full retraining cycle, which is crucial for real-time systems. This necessitates a fundamental shift in system design, from a traditional batch-processing pipeline to a continuous loop of ingestion, transformation, inference, and response, extending the problem to the hardware and architectural level.

V. ADVANCED ARCHITECTURES AND FUTURE DIRECTIONS

Future research is focused on pushing the boundaries of ML for network security by addressing issues of privacy, scalability, and interpretability. The convergence of these trends points toward a new generation of systems that are not only robust but also trusted and privacy-preserving.

A. Distributed and Collaborative Learning (Federated Learning)

In a federated learning (FL) framework, multiple clients (e.g., network devices, edge servers) collaboratively train a global model without exchanging their raw data. Only model updates (e.g., weights, gradients) are transmitted to a central server, which aggregates them to improve the shared model. This approach is a powerful solution to the scalability and privacy challenges inherent in large, decentralized networks like IoT systems. It removes dependence on a central server, which addresses the single point of failure that plagues centralized architectures, and prevents attackers from exploiting centralized training data. Case studies in healthcare and mobile applications demonstrate FL's ability to maintain high performance while preserving data privacy by keeping sensitive information on the device.

B. Real-Time Streaming Frameworks

To handle the massive volume of continuous network data, real-time streaming frameworks are essential. Platforms like Apache Flink and Spark Streaming are specifically designed for this purpose. These frameworks enable applications like fraud detection and network monitoring to process data on the fly. They offer features like Complex Event Processing (CEP) to identify complex patterns and support for incremental learning, allowing models to be updated with new data in real-time without restarting the system.

C. The Rise of Explainable AI (XAI)

As complex "black box" DL models become the norm, there is a growing need for transparency and interpretability. Explainable AI (XAI) is a field dedicated to providing insights into how and why an AI model makes a particular decision. XAI is crucial for network security, as a security professional needs to understand the reasons behind an alert to take effective, actionable steps. While a deep learning model may accurately flag an anomaly, its true value is in enabling a human to respond to a threat. XAI helps to build trust in automated systems and elevate the role of the analyst from a simple monitor to a strategic responder. It is being applied to various security tasks, including intrusion detection, malware analysis, and identifying zero-day vulnerabilities. The convergence of federated learning and Explainable AI represents a broader movement toward ethical, trust-based, and human-centric cybersecurity systems. While Federated Learning solves the privacy and scalability problems by keeping data decentralized, XAI addresses the trust deficit by making the model's decisions interpretable. These two concepts, though technically distinct, converge to create a new vision for cybersecurity: a system that is not only robust and scalable but also privacy-preserving and trusted by the human operators who rely on it.

VI. CONCLUSION

This report presented an overview of machine learning approaches for real-time network anomaly detection, emphasizing the transition from traditional signature-based systems to data-driven techniques. While conventional machine learning models remain useful in resource-constrained environments, deep learning architectures such as CNNs and LSTMs demonstrate stronger performance by effectively analyzing high-dimensional and sequential network traffic. The selection of an appropriate architecture depends on the specific threat context, with CNNs suited for volumetric attacks and LSTMs effective for detecting multi-stage intrusions. Real-time detection introduces challenges including computational overhead, data imbalance, and concept drift, which require systems capable of continuous learning and adaptation. Modern streaming frameworks and adaptive learning methods play an important role in addressing these issues. Emerging directions such as federated learning and Explainable AI further strengthen collaborative, privacy-preserving analysis and improve transparency for security professionals. Overall, the future of network security depends on hybrid and scalable architectures that integrate multiple machine learning techniques to create adaptive, reliable, and trustworthy defense systems.

VII. FUTURE SCOPE

The field of real-time network anomaly detection continues to evolve, with several important research challenges that require further attention. Federated learning presents a promising approach for addressing privacy and scalability in distributed environments, yet its application in network anomaly detection remains relatively underexplored. Future research should focus on developing robust defenses against adversarial threats in federated systems, including model poisoning and backdoor attacks. Additionally, the emergence of quantum computing introduces new risks to existing cryptographic and distributed security architectures, highlighting the need for continued innovation in secure design. Explainable AI is also essential for improving transparency and trust in deep learning based detection models, but its practical integration into network security workflows remains limited. Research opportunities include applying XAI to digital forensics, identifying zero-day vulnerabilities, and strengthening models against adversarial evasion. Another persistent challenge is concept drift, where evolving network behavior reduces model effectiveness. Developing adaptive systems capable of continuous learning and retraining is therefore a key priority. Future work may also enhance model performance through architectural improvements, such as incorporating attention mechanisms into LSTM models to better capture relevant patterns in network traffic and improve anomaly detection accuracy.

REFERENCES

[1] M. Natkaniec, K. Kosek-Szott, S. Szott and G. Bianchi, "A Survey of Medium Access Mechanisms for Providing QoS in Ad-Hoc Networks," *IEEE communications surveys & tutorials*, vol. 15, no. 2, pp. 592-620, 2012.

[2] G. Sunkara, "The Role of AI and Machine Learning in Enhancing SD-WAN Performance," *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, vol. 14, no. 4, pp. 1-9, 7 December 2022.

[3] M. Karakus and A. Durresi, "Quality of service (QoS) in software defined networking (SDN): A survey," *Journal of Network and Computer Applications*, vol. 80, no. 1, pp. 200-218, 15 February 2017.

[4] K. Bouraqia, E. Sabir, M. Sadik and L. Ladid, "Quality of experience for streaming services: measurements, challenges and insights," *IEEE Access*, vol. 8, no. 1, pp. 13341-13361, 2020.

[5] Z. Mammeri, "Framework for parameter mapping to provide end-to-end QoS guarantees in IntServ/DiffServ architectures," *Computer Communications*, vol. 28, no. 9, pp. 1074-1092, 2 June 2005.

[6] S. R. Lima, P. Carvalho and V. Freitas, "Admission control in multiservice IP networks: architectural issues and trends," *IEEE Communications Magazine*, vol. 45, no. 4, pp. 114-121, 16 April 2007.

[7] A. Mohamad and H. A. Hussein, "Control Dynamic System and Qos Manager Agent Over Ipv6 Networks: Intserv and Diffserv Approach in Access Nodes," *ResearchSquare*, vol. 1, no. 1, pp. 1-36, 2023.

[8] A. Bahnasse, F. E. Louhab, H. A. Oulahyane, M. Talea and A. Bakali, "Novel SDN architecture for smart MPLS traffic engineering-DiffServ aware management," *Future Generation Computer Systems*, vol. 87, no. 1, pp. 115-126, 1 October 2018.

[9] M. Radivojević and M. Petar, "Quality of Service Implementation," *The Emerging WDM EPON*, vol. 1, no. 1, pp. 35-66, 13 May 2017.

[10] L. Han, Y. Qu, L. Dong and R. Li, "Flow-Level QoS Assurance via IPv6 In-Band Signalling," in *27th Wireless and Optical Communication Conference (WOCC 2018)*, Hualien, Taiwan, 2018.