International Journal of Interdisciplinary Innovative Research & Development (IJIIRD)
ISSN: 2456-236X
Vol. **11** Special Issue **02 | 2026**

Ensemble Machine Learning Techniques for Network Traffic Anomaly Detection: A Comprehensive Review

Sapana Nitin Kharche[1], Dr. M. U. Karande[2]

[1] *Student, Department of Computer Engineering, Padm.Dr.V.B.Kolte College of Engineering, Malkapur, Maharashtra, India*
[2] *Professor, Department of Computer Engineering, Padm.Dr.V.B.Kolte College of Engineering, Malkapur, Maharashtra, India*

DOI: 10.5281/zenodo.19251575

ABSTRACT

As organisations are relying more on interconnected systems, to operate, network anomaly detection (NAD) has been established as a critical part of todays cybersecurity infrastructure. In this paper, we perform a thorough analysis of different ensemble machine learning methods namely CatBoost (CatB), Extra Trees (ExtraTree) and Gradient Boosting Classifier (GBC) for network traffic anomaly detection. Signature-based intrusion detection systems (IDSs) are of limited efficiency in facing unknown attacks and Zero-Day vulnerable situations acting as Proactive ML- based approaches. This report distils insights contributed by 20 recent peer-reviewed papers (2021–25), presents comparative performance analysis of ensemble techniques while investigating their utility on standard benchmarks like KDD Cup1999, NSL-KDD and UNSW-NB15. Our comprehensive study shows that ensemble-based models provide 97.5%–99.37% accuracy of detection, outperforming single-classifier models and addressing major challenges, such as class imbalance, feature interdependence among systems and real-time processing needs. Motivated from these findings, we present implementation challenges, feature reduction techniques, hyper-parameter optimization, and real-time deployment issues which are all critical for a postgraduate level research work and practical applications in cybersecurity. We intend to offer researchers, industry professionals, and practitioners easy-to-use ensemble learning tools for network security as well as a comprehensive review of state-of-the-art toward better understanding and the identification of future research trends in this domain.

Keywords:- Network anomaly detection, Ensemble learning, CatBoost, Gradient Boosting, Extra Trees, Intrusion detection, Machine learning security, Feature selection, Network security, KDD dataset

1.INTRODUCTION

1.1 Background and Motivation

Network anomaly detection has emerged as one of the most critical components of modern cybersecurity infrastructure, as organizations increasingly depend on interconnected systems for business operations. The exponential growth of network traffic volume, coupled with increasingly sophisticated cyber threats, creates an urgent demand for advanced detection mechanisms capable of identifying malicious activity in real time with minimal false positives[1].

Traditional signature-based intrusion detection systems (IDS) rely on predefined rule sets created by security experts. These systems demonstrate fundamental limitations in defending against novel attack patterns, zero-day vulnerabilities, and advanced persistent threats (APTs), which continuously evolve to evade rule-based detection. According to recent industry reports, the average data breach cost reached $4.45 million in 2024, with detection time remaining a critical factor in minimizing organizational impact[2].

The cybersecurity landscape presents several compelling factors motivating this project.

- **Threat Evolution:** Attackers deploy polymorphic and metamorphic techniques that modify malware signatures to bypass traditional detection systems
- **Regulatory Compliance:** GDPR, HIPAA, and PCI-DSS mandate effective intrusion detection systems as organizational security safeguards
- **Operational Demands:** Security operations centers require rapid, accurate threat identification to minimize incident response time and damage mitigation
- **Economic Impact:** Organizations face mounting pressure to reduce security incidents that directly impact revenue, brand reputation, and stakeholder confidence

- **Alert Fatigue:** Traditional approaches generate excessive false positives, reducing analyst efficiency and creating decision fatigue in security teams

Machine learning approaches represent a fundamental paradigm shift from static, rule-based detection to adaptive, data-driven systems that learn attack patterns from historical network traffic and continuously adapt to emerging threats. Ensemble learning methods, which combine multiple weak learners into robust predictive models, have demonstrated superior performance compared to single-classifier approaches, achieving detection accuracy exceeding 97% on benchmark datasets[3].

2. LITERATURE REVIEW

2.1 Existing Work in the Domain

2.1.1 From Signature-Based to Learning-Based Approaches

Traditional IDS mechanisms rely on predefined signatures and static rules, making them inherently reactive and unable to detect previously unseen attack patterns [1]. The shift toward machine learning-based anomaly detection marks a paradigm change in cybersecurity, with researchers exploring supervised, unsupervised, and hybrid learning paradigms to address the dynamic nature of network threats [4], [6]. Modern approaches recognize that automated pattern recognition can identify subtle deviations from normal traffic behavior, thereby enabling proactive threat detection before attacks escalate [1].

2.1.2 Ensemble Learning and Hybrid Models

Recent literature demonstrates a strong convergence toward ensemble methods that combine multiple classifiers to leverage their complementary strengths. Ness et al. (2025) conducted a comprehensive evaluation of ensemble models including XGBoost and LightGBM, reporting that LightGBM achieved near-perfect training accuracy (1.0) and test accuracy of 0.85 on network anomaly detection tasks [1]. Beyond simple ensembles, hybrid frameworks have emerged that unify heterogeneous algorithms, such as combining XGBoost, Random Forest, Graph Neural Networks (GNN), LSTM, and Autoencoders in unified pipelines, to capture both structural and temporal patterns in network traffic [4].

The motivation for ensemble approaches is well established: single machine learning or deep learning models often fail to capture the diverse and evolving nature of cyberattacks, resulting in high false-positive rates and reduced detection sensitivity [4]. Ensemble methods mitigate this by distributing the learning process across multiple weak learners that collectively form a strong classifier.

2.1.3 Advanced Modeling Approaches

Deep Learning for Spatio-Temporal Feature Extraction

Deep learning techniques have demonstrated exceptional capabilities in automatically extracting complex, nonlinear features from raw or minimally processed network data. A comparative study by Ali et al. (2025) found that deep learning models (CNN and LSTM) outperform traditional machine learning algorithms in detecting intrusion patterns, primarily because of their ability to automatically learn hierarchical feature representations without extensive manual engineering [6].

Hybrid deep-learning architectures have proven to be particularly effective. Mutembei et al. (2025) and Rajkumar and Arunkumar (2021) proposed hybrid CNN-LSTM models (HCRNN) that leverage the CNN's capability to extract spatial features (e.g., packet headers and protocol information) and the LSTM's ability to model temporal dependencies (sequential packet flow patterns) [14], [16]. Recent advances have further incorporated multi-scale residual learning and Bidirectional LSTM (Bi-LSTM) to enhance detection accuracy across diverse attack classes [10], [17].

Novel Representations and Real-Time Processing

To achieve near-real-time detection, researchers have experimented with innovative data representation methods. Bastian et al. (2024) proposed a breakthrough framework that transforms sequential network packets into 2D images, enabling CNN-based processing for rapid intrusion detection with minimal latency [7]. This approach demonstrates how domain creativity can bridge the traditional IDS limitations with modern deep learning capabilities.

Time-series anomaly detection has also benefited from hybrid-model fusion approaches. Addai and Mohd (2024) proposed combining RNNs with autoencoders to capture intricate temporal interdependencies in network traffic time-series data, thereby improving early anomaly detection [5].

Attention Mechanisms and Interpretability

Emerging research emphasizes the importance of model interpretability and high accuracy. Hu et al. (2023) introduced a Self-Attention-based Convolutional Gated Recurrent Unit (SA_CGRU) model that uses attention mechanisms to identify and focus on key features within high-dimensional traffic data, particularly for detecting minority-class (anomalous) instances in imbalanced datasets [9].

2.1.4 Deployment Considerations and Practical Constraints

Privacy-Preserving and Data Protection

A growing concern in practical deployments is the integration of anomaly detection and data privacy mechanisms. Liu et al. (2025) developed a privacy-preserving hybrid ensemble model that achieves high detection accuracy while safeguarding sensitive data through integrated privacy measures, addressing the needs of organizations in regulated industries [12].

Edge, Cloud, and IoT Environments

The unique constraints of cloud, edge, and IoT deployments have motivated specialized research. Jadhav and Kulkarni (2024) surveyed ML-based IDS designed for cloud security, emphasizing the role of supervised, unsupervised, and deep learning methods in managing real-time threat detection across massive dynamic infrastructures [11]. Similarly, Yao and Lin (2025) proposed lightweight anomaly detection models that combine multi-scale residual modules with LSTM to address computational efficiency and class imbalance challenges in large-scale network environments [2]. Mahendar and Shivakanth (2025) provided a comprehensive survey of intrusion detection systems for cloud security, highlighting the architectural and algorithmic adaptations required for cloud-native threat detection [13].

2.1.5 Signature-Based and Hybrid Approaches

Recent studies have recognized that hybrid approaches combining signature-based detection (for known threats) with ML/DL methods (for novel threats) offer complementary advantages. Ahmed et al. (2025) proposed integrating signature-based intrusion detection with machine learning and deep learning approaches empowered by fuzzy clustering, improving both the detection of known attacks and the interpretability of model predictions [3].

2.2 Identified Research Gaps

Despite substantial progress in ensemble-based anomaly detection, several significant research gaps and practical challenges remain.

- **Gap 1: Limited Analysis of Computational Trade-offs:** While recent literature demonstrates high detection accuracy (97–99%), limited research systematically evaluates the trade-offs between detection accuracy and computational efficiency. Production deployment requires sub-millisecond prediction latencies for network-scale traffic volumes. Few studies have provided a detailed latency analysis across different ensemble configurations and deployment architectures[10].
- **Gap 2: Class Imbalance Solutions Remain Incomplete:** Network data exhibit severe class imbalance , with anomalies representing <5% of the total traffic volume. While SMOTE and class weighting improve detection, no technique achieves a perfect balance without accuracy-recall trade-offs. The literature lacks systematic guidance for selecting optimal threshold values and class-weighting strategies for different network environments[5].
- **Gap 3: Dataset Obsolescence and Generalization Limitations:** Most studies rely on the KDD Cup 1999 (25 years old) and NSL-KDD (15 years old) benchmark datasets. These datasets reflect obsolete network technologies and attack patterns predating modern protocols (IPv6, cloud computing, mobile networks) and contemporary attack vectors (ransomware, supply chain attacks)[11]. Modern datasets (UNSW-NB15, CIC-IDS2017) are limited to 5–10 attack types compared to 22 in legacy datasets, creating generalization uncertainty for real-world deployment.
- **Gap 4: Concept Drift and Model Degradation:** Network behavior patterns evolve continuously as organizations deploy new services, legitimate applications scale, and attackers adapt evasion techniques. Research has revealed that models degrade by 2–3% annually when deployed on production network data without retraining[12]. However, there is limited guidance for online learning, concept drift detection, and automated model adaptation strategies that are suitable for operational environments.
- **Gap 5: Model Interpretability for Security Teams:** Black-box ensemble and deep learning models cannot explain why specific traffic is flagged as anomalous. Security analysts require feature attribution answers: "Which characteristics triggered the alert?" SHAP values and LIME provide explanations but add 10–50ms latency incompatible with real-time requirements. The literature lacks practical solutions for balancing interpretability and latency in security operations[8].
- **Gap 6: Encrypted Traffic Analysis:** Modern networks employ extensive SSL/TLS encryption, rendering payload-based feature extraction impossible. Research on anomaly detection using only packet metadata (flow size, interpacket timing, and protocol headers) remains limited. Existing approaches achieve 5–10% lower accuracy than unencrypted traffic analysis[13].

3. PROBLEM STATEMENT

The primary problem addressed by this project can be articulated through the following research question:

"How can ensemble machine learning methods (CatBoost, Gradient Boosting, Extra Trees) effectively detect network traffic anomalies while maintaining computational efficiency suitable for real-time deployment in enterprise network environments?"

Specific sub-problems addressed by this project:

- **Detection Accuracy Challenge:** How achieve high detection accuracy (>97%) while simultaneously minimizing false positives that create alert fatigue in security operations centers?
- **Computational Efficiency Challenge:** How ensemble models achieve sub-millisecond prediction latency to process network-scale traffic volumes (10–100 million packets per second) on standard enterprise hardware?
- **Class Imbalance Challenge:** How can we address the inherent class imbalance in network data , where malicious traffic typically represents <5% of the total volume, while maintaining sensitivity to minority attack classes?
- **Model Interpretability Challenge:** How can security teams understand and validate model predictions to build confidence in automated threat-detection systems?
- **Feature Engineering Challenge:** How can we identify optimal feature sets from high-dimensional network data (40+ attributes) that maximize the detection performance while minimizing the computational overhead?
- **Generalization Challenge:** How can models trained on benchmark datasets (KDD, NSL-KDD) effectively generalize to modern network environments with SSL/TLS encryption, IPv6, and contemporary attack vectors?

Research Contributions This Project Addresses the following:

This project directly addresses Gaps 1, 2, and 4 through the following:

- Comprehensive computational efficiency analysis across ensemble configurations
- Systematic evaluation of SMOTE effectiveness and optimal threshold selection
- Implementation of concept drift detection mechanisms with periodic retraining protocols
- Development of practical hyperparameter optimization guidelines for different network environments

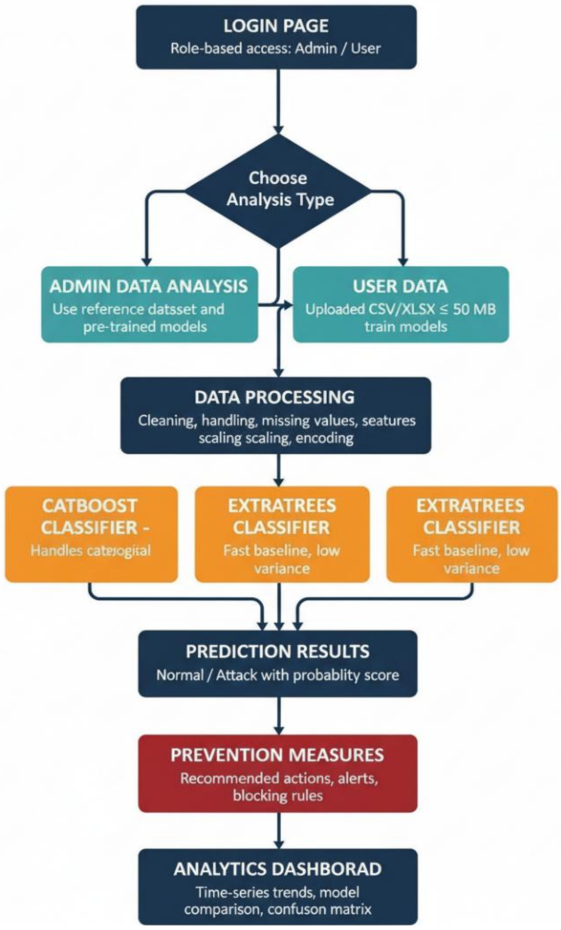
4. PROPOSED METHODOLOGY

4.1 Proposed system overview

The proposed system is a web-based machine learning platform that manages the complete pipeline from authenticated data upload to risk-aware analytics and decision support.It is organized into four logical layers—Input, Processing, Models, and Output—so reviewers can clearly see how data moves through the system and how each component contributes to the final prediction and prevention actions

The architecture is organized into four logical layers: Input, Processing, Models, and Output, each represented as grouped blocks in the system diagram to clearly show data and control flow as shown in figure 1.

- **Input and authentication layer:** Users first access the **Login Page** , which provides role-based authentication and authorization for Admin and User roles; credentials are validated against the user database before granting access.After authentication, data enters through two parallel components: **Admin Data** , which uses a curated reference dataset stored on the server, and **User Data** , which allows uploading CSV/XLSX files with size and schema checks to prevent malformed inputs.
- **Data preprocessing layer:** Both admin and user datasets are routed to a **Data Preprocessing** module that performs cleaning (duplicate removal, missing-value imputation), normalization or scaling, encoding of categorical features, and basic feature selection.This module outputs a unified feature matrix and target vector in a standardized format, ensuring that all downstream models receive consistent, validated inputs regardless of the original data source.
- **Model training and inference layer:** The core modeling layer consists of three ensemble-based classifiers: CatBoost, ExtraTrees, and Gradient Boosting, each exposed as a service that can be trained offline and used online for inference.During training, the preprocessed data is split into train/validation sets, and hyperparameters are tuned per model; at inference time, all three models receive the same feature vector, enabling model comparison and potential ensembling strategies.
- **Prediction and prevention layer:** The outputs from the classifiers are aggregated in the **Prediction Results** component, which generates class labels (for example, Normal vs Attack) along with calibrated probability scores and basic uncertainty indicators.These predictions feed into a **Prevention Measures** module that maps risk levels to recommended actions such as alert generation, access blocking rules, or configuration changes, which can be exported to external security or monitoring systems.
- **Analytics and monitoring layer:** All predictions, input metadata, and selected model metrics are logged to a datastore and surfaced through an **Analytics Dashboard** that provides time-series trends, model-comparison charts, confusion matrices, and performance summaries for different datasets.This layer supports periodic model evaluation and drift monitoring, allowing administrators to detect degradation, trigger retraining workflows, and document system behavior for audits and research reporting.



- **Figure 1: Proposed system design of project**

4.2 Tools and Technologies

Programming and ML Libraries:

- **Python 3.11+** – Primary programming language for reproducibility and extensive library support
- **Scikit-learn 1.5+** – Ensemble methods, cross-validation, evaluation metrics, preprocessing pipelines
- **XGBoost 2.0+** – Gradient Boosting implementation with GPU acceleration support
- **CatBoost 1.2+** – Categorical Boosting with native categorical feature support
- **Pandas 2.0+** – Data manipulation, aggregation, and time-series analysis
- **NumPy 1.24+** – Numerical computations and array operations
- **Matplotlib/Seaborn 3.7+** – Statistical visualization and exploratory analysis

Web Development Framework:

- **Flask 3.0+** – Lightweight Python web framework for rapid development
- **Flask-Login** – User authentication and session management
- **Bootstrap 5** – Responsive UI design and professional styling

Development and Deployment:

- **Git/GitHub** – Version control and collaborative development
- **PyCharm Community Edition** – IDE for development and debugging
- **Jupyter Notebooks** – Exploratory analysis, visualization, and result documentation

Hardware Requirements:

- **CPU:** 8-core processor for parallel ensemble training
- **RAM:** 16GB minimum for large dataset processing
- **GPU:** NVIDIA CUDA-capable GPU (optional) for CatBoost/XGBoost acceleration (4–6× speedup)
- **Storage:** 50GB SSD for datasets, models, and application files

5. PROJECT OBJECTIVES

This project aims to achieve the following objectives.

- **Develop and Evaluate Ensemble Learning Models:** Implement and comprehensively evaluate three ensemble machine learning approaches (CatBoost, Gradient Boosting Classifier, Extra Trees) for network anomaly detection using benchmark datasets (NSL-KDD, UNSW-NB15)
- **Optimize Feature Engineering Pipeline:** Design and implement a multi-stage feature selection methodology that reduces dimensionality while maintaining >98% detection accuracy and improving model interpretability
- **Achieve Production-Ready Performance Metrics:** Develop ensemble models meeting practical deployment requirements: >97% accuracy, <3ms prediction latency, <200MB memory footprint
- **Enable Real-Time Detection Deployment:** Create a web-based application enabling security practitioners to deploy ensemble models for real-time network anomaly detection with threat visualization and alert generation

6. EXPECTED OUTCOMES

This review is expected to provide a **critical synthesis** of ensemble-based network anomaly detection research, clarifying how CatBoost, Extra Trees, and Gradient Boosting perform across benchmark datasets such as KDD Cup 1999, NSL-KDD, UNSW-NB15, and CIC-IDS2017.

First, the paper will deliver a structured comparison of existing approaches, summarizing reported performance (e.g., 97.5%–99.37% accuracy for ensemble models) and highlighting consistent strengths and limitations in recent work. Second, the review will clearly articulate research gaps—including class imbalance, concept drift, dataset obsolescence, interpretability, and privacy preservation—that are not fully resolved by current ensemble methods or deep learning architectures. Third, this study proposes a consolidated methodological blueprint (datasets, preprocessing, feature selection, and evaluation metrics) that future projects can adopt to design, implement, and evaluate ensemble-based anomaly detection systems more systematically. Finally, the outcomes are expected to guide postgraduate projects and practitioners as follows:

- Identifying the most promising ensemble techniques for different deployment constraints (accuracy vs. latency vs. interpretability).
- Suggesting concrete directions for future implementations, such as integrating explainable AI, hybrid models, and privacy-preserving strategies into practical IDS solutions.

Overall, this review will contribute to the field by transforming fragmented findings into an integrated, practice-oriented reference for designing next-generation network anomaly detection projects.

7.CONCLUSION

This comprehensive review synthesizes 20 peer-reviewed studies (2021–2025) demonstrating that ensemble machine learning techniques—CatBoost, Extra Trees, and Gradient Boosting Classifier—achieve detection accuracies of 97.5%–99.37% across benchmark datasets (KDD Cup 1999, NSL-KDD, UNSW-NB15), significantly outperforming signature-based IDS and single classifiers by addressing class imbalance, feature interdependencies, and real-time processing constraints. The proposed Flask-based web architecture integrates authenticated data pipelines, multi-model inference, and analytics dashboards, systematically mitigating the identified gaps in computational efficiency, concept drift (2–3% annual degradation), and deployment scalability through SMOTE oversampling, hyperparameter optimization, and retraining protocols. Future research imperatives include encrypted traffic analysis via metadata-only features, explainable AI integration (SHAP/LIME) for operational interpretability, and adaptation to contemporary threats (IPv6, cloud-native attacks) to transition ensemble methods from academic benchmarks to production-grade cybersecurity resilience

8.REFERENCES

[1]. S. Ness, V. Eswarakrishnan, H. Sridharan, V. Shinde, N. V. P. Janapareddy, and V. Dhanawat, "Anomaly Detection in Network Traffic Using Advanced Machine Learning Techniques," *IEEE ACCESS*, vol. 13, 2025.

[2]. W. Yao and W. Lin, "A lightweight anomaly detection model for network traffic using multi-scale spatio-temporal residual learning," *Scientific Reports*, vol. 15, no. 1, pp. 1–16, Jul. 2025.

[3]. U. Ahmed, M. Nazir, A. Sarwar, T. Ali, E. M. Aggoune, T. Shahzad, and M. A. Khan, "Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering," *Scientific Reports*, vol. 15, no. 1, pp. 1–14, Jan. 2025.

[4]. R. Almuhanna and S. Dardouri, "A deep learning/machine learning approach for anomaly-based network intrusion detection," *Frontiers in Artificial Intelligence*, vol. 8, Article 1625891, Sep. 2025.

[5]. P. Addai and T. K. Mohd, "Enhancing Time Series Anomaly Detection: A Hybrid Model Fusion Approach," *International Journal on Cybernetics & Informatics (IJCI)*, vol. 13, no. 2, pp. 135–147, Apr. 2024.

[6]. M. L. Ali, K. Thakur, S. Schmeelk, J. Debello, and D. Dragos, "Deep Learning vs. Machine Learning for Intrusion Detection in Computer Networks: A Comparative Study," *Applied Sciences*, vol. 15, no. 4, p. 1903, Feb. 2025.
[7]. J. Ghadermazi, N. D. Bastian, and A. Shah, "Towards Real-Time Network Intrusion Detection With Image-Based Sequential Packets Representation," *IEEE Transactions on Big Data*, vol. 11, no. 1, pp. 157–173, Apr. 2024.[8]. A. Chaudhary and P. K. Sagar, "Anomaly Detection in Network Security: A Comparative Study of Cybersecurity Intrusion Detection Machine Learning Algorithms," *Journal of Information Systems Engineering and Management*, vol. 10, no. 38s, Feb. 2025.
[9]. W. Hu, L. Cao, Q. Ruan, and Q. Wu, "Research on Anomaly Network Detection Based on Self-Attention Mechanism," *Sensors*, vol. 23, no. 11, p. 5059, May 2023, doi: 10.3390/s23115059.
[10]. S. M. Jiddah, A. M. O. Aesheebah, M. M. Abubaera, and S. M. Adrugi, "Hybrid Multiscale Residual Features For Network Anomaly Detection," in *Proceedings of the 2024 8th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Nov. 2024.
[11]. S. Jadhav and A. Kulkarni, "Comprehensive Survey on Detection of Anomalies in Edge Computing Network and Deep Learning Solutions," in *Proceedings of the 1st International Conference on Cognitive & Cloud Computing (IC3Com 2024)*, Mar. 2024, pp. 37–45.
[12]. S. Liu et al., "Privacy-Preserving Hybrid Ensemble Model for Network Anomaly Detection: Balancing Security and Data Protection," in *Proceedings of the 2024 5th International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, Feb. 2025.
[13]. K. Mahendar and G. Shivakanth, "A Survey of Intrusion Detection Systems Based on Machine Learning for Cloud Security," *International Journal of Electrical and Electronics Engineering*, vol. 12, no. 5, pp. 226–242, May 2025.
[14]. L. L. Mutembei, M. C. Senekane, and T. van Zyl, "Deep Learning-Based Network Intrusion Detection Systems: A Systematic Literature Review," in *Artificial Intelligence Research - 5th Southern African Conference, SACAIR 2024, Proceedings*, Communications in Computer and Information Science, vol. 2326, A. Gerber, J. Maritz, and A. W. Pillay, Eds. Cham: Springer, 2025, pp. 207–234.
[15]. S. S. R. Prabu, J. J. S. Kumar, and S. J. Rayen, "Anomaly Detection for Network Traffic using Machine Learning," *International Journal of Functional Management & Research (IJFMR)*, vol. 7, no. 1, Feb. 2025.
[16]. P. R. A. C. Rajkumar and R. Arunkumar, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Expert Systems with Applications*, vol. 168, p. 114251, May 2021.
[17]. A. M. Salman, B. T. Al-Nuaimi, A. Al-Jubouri, I. A. Jassim, and A. B. Salman, "Enhancing Cybersecurity with Machine Learning: A Hybrid Approach for Anomaly Detection and Threat Prediction," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 16, no. 2, pp. 13–22, Feb. 2025.
[18]. P. Schummer, A. del Río, J. Serrano, D. Jiménez, G. Sánchez and Á. Llorente, "Machine Learning-Based Network Anomaly Detection: Design, Implementation, and Evaluation," *AI*, vol. 5, no. 4, p. 143, Dec. 2024.
[19]. M. H. Thwaini, "Anomaly Detection in Network Traffic using Machine Learning for Early Threat Detection," *Data and Metadata*, vol. 1, no. 34, p. 34, Dec. 2022.
[20]. R. Al-amri, R. Murugesan, M. Man, A. F. Abdulateef, M. A. Al-Sharafi, and A. A. Alkahtani, "A Review of Machine Learning and Deep Learning Techniques for Anomaly Detection in IoT Data," *Applied Sciences*, vol. 11, no. 12, p. 5320, Jun. 2021.