Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

An Ensemble Machine Learning Approach for Real-Time Early Detection of Cyber Threats Using Network Traffic Data

Research Article https://stem.techspherejournal.com

Article Info
Revised Date: 2nd June, 2026
Accepted Date: 10th July, 2026
Published Date: 15th July, 2026

Keywords
Cyber Threat Detection
Ensemble Machine Learning
Network Traffic Analysis
Intrusion Detection System (IDS)
Real-Time Cybersecurity

Author Details
Mgbeafulike Ike Joseph[1], Okeke Ogochukwu Clementina[2], Azaka Maduabuchuku[3]
1, 2 Department of Computer Science, Faculty of Physical Sciences, Chukwuemeka Odumegwu Ojukwu University, Anambra State.
3 Department of Computer Science, Faculty of Computing, Dennis Osadebay University, Asaba, Delta State.
****Corresponding author's email***: azaka.maduabuchuku@dou.edu.ng
DOI: https://doi.org/10.5281/zenodo.21374507

ABSTRACT

.

The spontaneous growth of cloud computing, the Internet of Things (IoT), and enterprise networks has significantly increased the volume and complexity of cyber threats, making traditional signature-based intrusion detection systems less effective against sophisticated and zero-day attacks. Machine learning has emerged as a promising solution for intelligent cyber threat detection, while ensemble learning further enhances detection performance by combining the strengths of multiple classifiers to improve accuracy, robustness, and generalization. This study proposes RT-ECTDNet (Real-Time Ensemble Cyber Threat Detection Network), an ensemble machine learning framework for the real-time early detection of cyber threats using network traffic data. The framework employs a layered architecture comprising data acquisition, preprocessing, feature engineering, ensemble learning, threat prediction, and alert generation. Feature optimization was performed using Information Gain, Chi-Square, and Recursive Feature Elimination, while Random Forest, XGBoost, LightGBM, Extra Trees, and CatBoost were integrated through voting and stacking strategies. The model was evaluated using the CICIDS2017 benchmark dataset and assessed with standard performance metrics. Experimental results showed that RT-ECTDNet achieved an accuracy of 99.23%, precision of 99.39%, recall of 99.26%, F1-score of 99.32%, ROC-AUC of 0.998, a false positive rate of 0.81%, and an average detection latency of 11.34 ms, outperforming all individual classifiers. The findings demonstrate that RT-ECTDNet provides an accurate, scalable, and computationally efficient solution for real-time cyber threat detection, making it suitable for deployment in enterprise, cloud, and IoT environments.

1 Introduction

1.1 Background to the Study

The rapid advancement of Information and Communication Technology (ICT), cloud computing, the Internet of Things (IoT), artificial intelligence, and 5G networks has transformed digital connectivity while significantly expanding the cyber-attack surface [1]. As organizations increasingly rely on interconnected systems, cyber threats such as ransomware, distributed denial-of-service (DDoS) attacks, advanced persistent threats (APTs), phishing, botnets, insider attacks, and zero-day exploits have become more frequent and sophisticated. Network traffic analysis has therefore

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

become a critical component of cybersecurity because communication flows contain valuable information that can be used to distinguish legitimate activities from malicious behaviour [2]. Although traditional signature-based intrusion detection systems are effective against known attacks, they perform poorly against emerging and previously unseen threats, while anomaly-based methods often suffer from high false positive rates [3].

Machine learning has emerged as an effective approach for intelligent cyber threat detection by learning complex traffic patterns from historical network data. Supervised, unsupervised, and deep learning algorithms have demonstrated considerable success in identifying malicious network activities [4]. However, individual classifiers often face challenges such as overfitting, limited generalization, computational complexity, or reduced performance on imbalanced datasets. Ensemble learning addresses these limitations by combining multiple classifiers to improve predictive accuracy, robustness, and resilience. Techniques such as Bagging, Boosting, Stacking, Voting, Random Forest, XGBoost, LightGBM, and CatBoost have consistently outperformed standalone models in intrusion detection tasks [5].

The growing demand for real-time cyber threat detection in enterprise, cloud, and IoT environments requires intelligent frameworks capable of processing high-volume network traffic with minimal latency. Benchmark datasets such as CICIDS2017, UNSW-NB15, and Bot-IoT, together with effective preprocessing and feature engineering, provide a reliable basis for developing such systems [6]. Motivated by the limitations of existing intrusion detection approaches, this study proposes RT-ECTDNet (Real-Time Ensemble Cyber Threat Detection Network). This ensemble machine learning framework integrates multiple classifiers to improve detection accuracy, reduce false alarms, and provide timely identification of cyber threats for modern network environments.

1.2 Statement of the Problem

The rapid growth of cloud computing, Internet of Things (IoT) devices, enterprise networks, and digital services has significantly increased the frequency and sophistication of cyber-attacks, including ransomware, distributed denial-of-service (DDoS) attacks, advanced persistent threats (APTs), botnets, phishing, insider attacks, and zero-day exploits [1]. Traditional signature-based intrusion detection systems are effective only against known attacks and require continuous signature updates, making them inadequate for detecting emerging threats [7]. Although anomaly-based approaches improve the detection of unknown attacks, they often produce high false positive rates, resulting in unnecessary alerts and reduced operational efficiency.

Machine learning has emerged as a promising solution for intelligent cyber threat detection; however, standalone classifiers often suffer from limitations such as poor generalization, high computational cost, and reduced performance on complex or high-dimensional network traffic [8]. Moreover, many existing studies focus primarily on detection accuracy while overlooking computational efficiency and real-time performance. Consequently, there is a need for a robust ensemble machine learning framework that combines the strengths of multiple classifiers to achieve high detection accuracy, low false positive rates, minimal detection latency, and scalability for deployment in enterprise, cloud, and IoT environments. This need forms the motivation for the present study.

1.3 Aim of the Study

To develop an ensemble machine learning framework for real-time early detection of cyber threats using network traffic data.

1.3.1. Objectives of the Study

1. To analyze network traffic characteristics associated with different cyber threats.
2. To preprocess and extract relevant features from network traffic datasets.
3. To develop an ensemble machine learning model for cyber threat detection.
4. To evaluate the proposed model using standard performance metrics.
5. To compare the proposed ensemble model with individual machine learning classifiers.

1.3.2. Research Questions

1. Which network traffic features contribute most to cyber threat detection?

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

2. How effective is feature engineering in improving detection accuracy?
3. Can an ensemble learning model accurately detect cyber threats in real time?
4. How does the proposed model compare with standalone machine learning algorithms?
5. What is the computational performance of the proposed framework in real-time environments?

1.4 Scope of the Study

This study focuses on the development and evaluation of an ensemble machine learning framework for the real-time early detection of cyber threats using network traffic data. The framework is designed to identify malicious network activities by analysing network flow features extracted from publicly available benchmark intrusion detection datasets.
The study is limited to supervised machine learning techniques and investigates the performance of selected ensemble methods developed from individual classifiers such as Random Forest, XGBoost, LightGBM, CatBoost, and Extra Trees. Feature preprocessing, feature selection, model training, hyperparameter optimization, and performance evaluation constitute the major components of the proposed framework.
Performance evaluation will be conducted using widely accepted cybersecurity metrics, including accuracy, precision, recall, F1-score, Receiver Operating Characteristic Area Under the Curve (ROC-AUC), false positive rate, detection rate, Matthews Correlation Coefficient (MCC), and detection latency. The proposed model will be compared with selected standalone machine learning classifiers to determine its effectiveness.
The study is confined to network-based cyber-attack detection using benchmark datasets and does not address host-based intrusion detection, malware reverse engineering, digital forensics, cryptographic analysis, or offensive cybersecurity techniques. The proposed framework is intended to support real-time network intrusion detection in enterprise, cloud computing, and IoT environments.

2 Literature Review

2.1 Concept of Cyber Threat Detection

Cyber threat detection refers to the continuous process of identifying malicious activities that compromise the confidentiality, integrity, and availability of information systems. It involves monitoring network traffic, system logs, user behaviour, and endpoint activities to detect known and unknown attacks before they cause significant damage [9]. Modern cyber threat detection has evolved from static signature matching to intelligent data-driven approaches that leverage artificial intelligence (AI), machine learning (ML), and behavioural analytics to identify sophisticated attacks in real time [9], [10].

2.1.1. Evolution of Cyber Threats

The cyber threat landscape has undergone significant transformation over the past four decades. Early cyber-attacks were largely experimental, involving simple viruses and worms developed primarily for curiosity or notoriety. Examples include the Morris Worm (1988) and Melissa Virus (1999), which exploited vulnerabilities to spread rapidly across computer networks [11]. The commercialization of the Internet shifted cybercrime toward financial gain. Malware families such as Zeus, WannaCry, and Emotet introduced capabilities including credential theft, ransomware, and botnet formation. More recently, cyber-attacks have become highly organized, employing Advanced Persistent Threats (APTs), supply chain attacks, fileless malware, cryptojacking, and AI-assisted attack techniques [12]. These attacks often combine multiple attack vectors, making detection significantly more challenging than traditional signature-based methods.
The rapid adoption of cloud computing, Software-Defined Networking (SDN), Internet of Things (IoT), edge computing, and Industrial Internet of Things (IIoT) has further expanded the attack surface. Modern enterprise networks generate massive volumes of heterogeneous traffic, requiring intelligent detection systems capable of identifying subtle deviations from normal communication behaviour [13].

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

2.1.2. Categories of Cyber Attacks

Cyber-attacks can be classified according to their objectives, attack vectors, and operational characteristics. Malware attacks remain among the most prevalent threats and involve malicious software such as viruses, worms, trojans, ransomware, spyware, rootkits, and botnets that compromise the confidentiality, integrity, or availability of computer systems. Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) attacks overwhelm network resources with excessive traffic, thereby preventing legitimate users from accessing critical services [14] . Phishing and other social engineering attacks exploit human vulnerabilities by deceiving users into disclosing sensitive information such as usernames, passwords, and financial credentials.

More sophisticated threats include Advanced Persistent Threats (APTs), which are highly organized and long-term attacks in which adversaries maintain unauthorized access to targeted systems for extended periods while avoiding detection [15]. Insider threats originate from authorized users who intentionally or unintentionally compromise organizational security through negligence, misuse of privileges, or malicious actions. Zero-day attacks exploit previously undisclosed software vulnerabilities before security vendors release appropriate security patches, making them particularly difficult to detect using traditional signature-based techniques [16].

Web application attacks constitute another significant category of cyber threats. These attacks exploit vulnerabilities within web applications through techniques such as Structured Query Language (SQL) injection, Cross-Site Scripting (XSS), and Remote Code Execution (RCE), leading to unauthorized access, data theft, or complete system compromise [17]. Since each attack category exhibits distinct network traffic characteristics, machine learning algorithms combined with effective feature engineering have become increasingly valuable for automatically identifying and classifying diverse cyber threats in real time.

2.1.3. Network Intrusion Detection Systems

Intrusion Detection Systems (IDSs) are essential cybersecurity solutions designed to monitor network and system activities for malicious behaviour. IDSs are broadly classified into Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS) [18]. HIDS monitors activities occurring within individual hosts, including file modifications, process execution, registry changes, and operating system logs, whereas NIDS analyzes packets traversing communication networks to identify suspicious traffic patterns before they reach target systems.

Based on detection methodology, IDSs can be categorized as signature-based, anomaly-based, specification-based, or hybrid systems. Signature-based IDSs compare observed traffic with predefined attack signatures and provide high detection accuracy for known attacks but are ineffective against previously unseen threats [19]. In contrast, anomaly-based IDSs establish profiles of normal network behaviour and identify deviations that may indicate malicious activities, making them more suitable for detecting zero-day attacks, although they often generate higher false positive rates.

Specification-based IDSs detect attacks by verifying whether system behaviour conforms to predefined security specifications, while hybrid IDSs integrate multiple detection techniques to leverage their complementary strengths [20]. Modern hybrid intrusion detection systems increasingly incorporate machine learning algorithms to improve detection accuracy, reduce false alarms, and enhance adaptability to evolving cyber threats, thereby making them suitable for protecting enterprise, cloud, and IoT environments.

2.1.4. Network Traffic Analysis

Network traffic analysis forms the foundation of machine learning-based intrusion detection by examining communication patterns between interconnected devices. Network traffic consists of packets exchanged across communication channels, with each packet containing attributes such as source and destination IP addresses, source and destination ports, protocol type, packet length, flow duration, packet count, byte count, Time-to-Live (TTL), TCP flags, and packet arrival time [21]. These raw attributes describe the behaviour of network communications and serve as the basis for extracting informative features used by machine learning models. Contemporary intrusion detection datasets, including CICIDS2017 and UNSW-NB15, typically contain more than seventy engineered traffic features derived from these packet-level characteristics.

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

Unlike packet-based inspection, flow-based traffic analysis summarizes packets belonging to the same communication session into a single network flow, commonly represented by the five-tuple consisting of source IP address, destination IP address, source port, destination port, and communication protocol [22]. Flow-based analysis significantly reduces computational overhead while preserving important behavioural characteristics required for attack detection. Common flow statistics include flow duration, average packet size, packet inter-arrival time, forward and backward packet counts, flow byte rate, and packet rate, all of which provide richer behavioural information than individual packets and are highly effective for identifying anomalous communication patterns [23].

2.2 Related Empirical Studies

Recent studies consistently demonstrate that machine learning-based intrusion detection systems outperform traditional signature-based approaches, particularly in detecting previously unseen attacks. Researchers have evaluated algorithms such as Random Forest, XGBoost, Support Vector Machines, Deep Neural Networks, and LightGBM using benchmark datasets including CICIDS2017, UNSW-NB15, CSE-CIC-IDS2018, and Bot-IoT. Random Forest is frequently reported to provide high classification accuracy and robustness to noisy data, while XGBoost often achieves superior precision and computational efficiency due to its regularization mechanisms and optimized tree construction [24].

Ensemble learning has emerged as one of the most promising strategies for enhancing intrusion detection performance [25]. Studies employing bagging, boosting, stacking, and voting classifiers generally report improvements in accuracy, recall, F1-score, and Area Under the ROC Curve (AUC), alongside reductions in false positive rates compared to individual classifiers [26]. Hybrid ensembles that combine heterogeneous learners have demonstrated particular effectiveness in detecting diverse attack categories, including DDoS attacks, botnets, brute-force attacks, and web application exploits.

Comparative analyses indicate that no single classifier consistently performs best across all datasets and attack scenarios. Performance varies with data characteristics, feature engineering methods, class imbalance, and hyperparameter optimization. Although ensemble approaches generally outperform standalone models, many studies prioritize classification accuracy while overlooking real-time deployment constraints such as inference latency, computational overhead, memory consumption, and scalability in high-speed networks [27].

Several research gaps remain. First, many existing models are evaluated on offline benchmark datasets without validation under real-time traffic conditions. Second, limited attention has been given to optimizing feature selection jointly with ensemble learning to reduce computational complexity. Third, the explainability and interpretability of ensemble-based IDSs remain underexplored, limiting their practical adoption in Security Operations Centers (SOCs). Finally, few studies comprehensively evaluate the trade-off between detection accuracy, false positive rate, and real-time processing performance. Addressing these gaps motivates the development of the proposed ensemble framework for real-time cyber threat detection.

2.3 Theoretical Framework

The proposed framework is grounded in the Bias-Variance Trade-off Theory, Ensemble Learning Theory, and Anomaly Detection Theory. The Bias-Variance Trade-off Theory explains that combining multiple models helps reduce prediction errors by balancing underfitting and overfitting, thereby improving generalization [28]. Ensemble Learning Theory further posits that integrating diverse classifiers produces more accurate, robust, and reliable predictions than individual models by exploiting their complementary strengths [29]. Anomaly Detection Theory complements these concepts by assuming that malicious network traffic exhibits behavioural patterns that differ from normal activities, enabling machine learning models to identify both known and previously unseen cyber threats. Together, these theories provide the conceptual foundation for the proposed ensemble-based framework for real-time cyber threat detection.

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

2.4 Conceptual Framework

The proposed conceptual framework integrates network traffic analysis, feature engineering, ensemble machine learning, and real-time decision-making into a unified cyber threat detection architecture. The proposed RT-ECTDNet Ensemble conceptual framework is presented in Figure 1

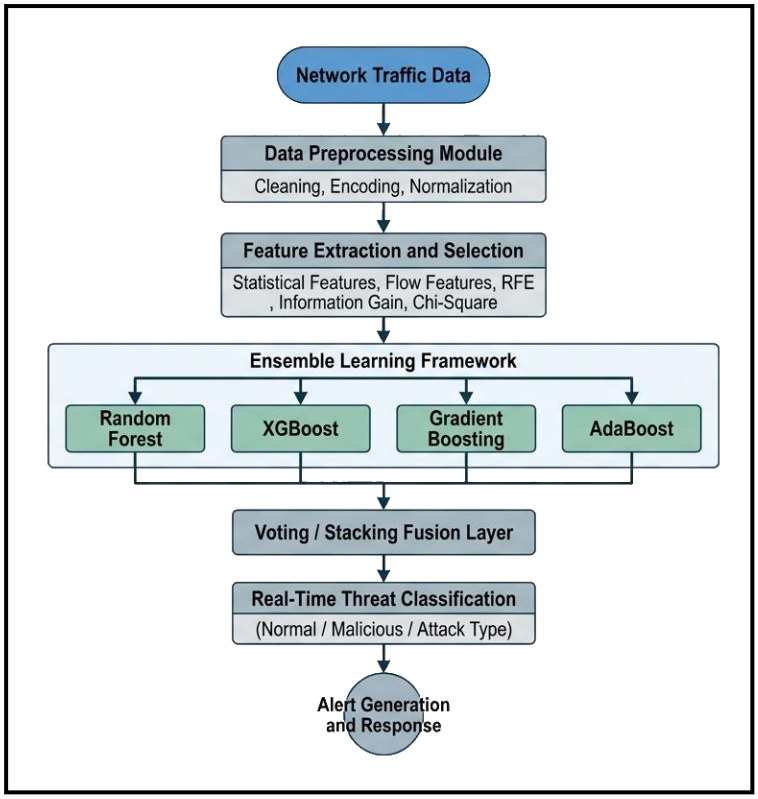

Figure 1: The Proposed RT-ECTDNet Ensemble Conceptual

The framework illustrates the end-to-end workflow of the proposed system. Raw network traffic is first pre-processed to remove inconsistencies and normalize feature values. Relevant traffic attributes are then extracted and refined using feature selection techniques such as Recursive Feature Elimination (RFE), Information Gain, and Chi-Square. The selected features are supplied to multiple ensemble learners, whose outputs are combined through a voting or stacking mechanism to generate the final classification. The resulting predictions enable real-time detection of malicious activities and support timely alert generation for cybersecurity operations. This architecture is designed to achieve high detection accuracy, reduced false positive rates, and low inference latency, making it suitable for deployment in enterprise, cloud, and IoT environments.

3 Methodology

3.1 Research Design

This study adopts an experimental research design to develop and evaluate an ensemble machine learning framework for the real-time early detection of cyber threats using network traffic data. Experimental research is appropriate because it enables the systematic development, training, testing, and comparative evaluation of multiple machine learning models under controlled conditions. The design facilitates the investigation of cause-and-effect relationships between feature engineering techniques, ensemble learning strategies, and the predictive performance of cyber threat detection models. The experimental workflow consists of dataset acquisition, data preprocessing, feature engineering, model development, ensemble construction, model training, performance evaluation, and comparative analysis. The proposed ensemble

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

framework is evaluated against individual machine learning classifiers using benchmark intrusion detection datasets. Standard evaluation metrics are employed to assess classification accuracy, computational efficiency, robustness, and real-time detection capability.

3.2 System Architecture

The proposed framework adopts a six-layer architecture to support accurate and efficient real-time cyber threat detection. The Data Acquisition Layer collects network traffic from benchmark datasets or live packet capture tools, providing the raw input for analysis. The Data Preprocessing Layer improves data quality through missing value handling, duplicate removal, categorical encoding, normalization, feature scaling, and data balancing. Next, the Feature Engineering Layer extracts and selects the most informative statistical and flow-based features while reducing feature redundancy and computational complexity. The optimized feature set is then processed by the Ensemble Learning Layer, where multiple classifiers, including Random Forest, XGBoost, LightGBM, Extra Trees, and CatBoost, are trained and combined using Soft Voting, Weighted Voting, or Stacking to improve prediction accuracy and robustness. The Threat Prediction Layer classifies each network flow as normal or malicious and identifies the corresponding attack category based on the ensemble's output. Finally, the Alert Generation Layer produces real-time security alerts containing relevant information such as attack type, confidence score, timestamp, source and destination addresses, and severity level, which can be integrated into Security Information and Event Management (SIEM) systems or Security Operations Centers (SOCs) for timely incident response. The system architectural diagram is presented in Figure 2.

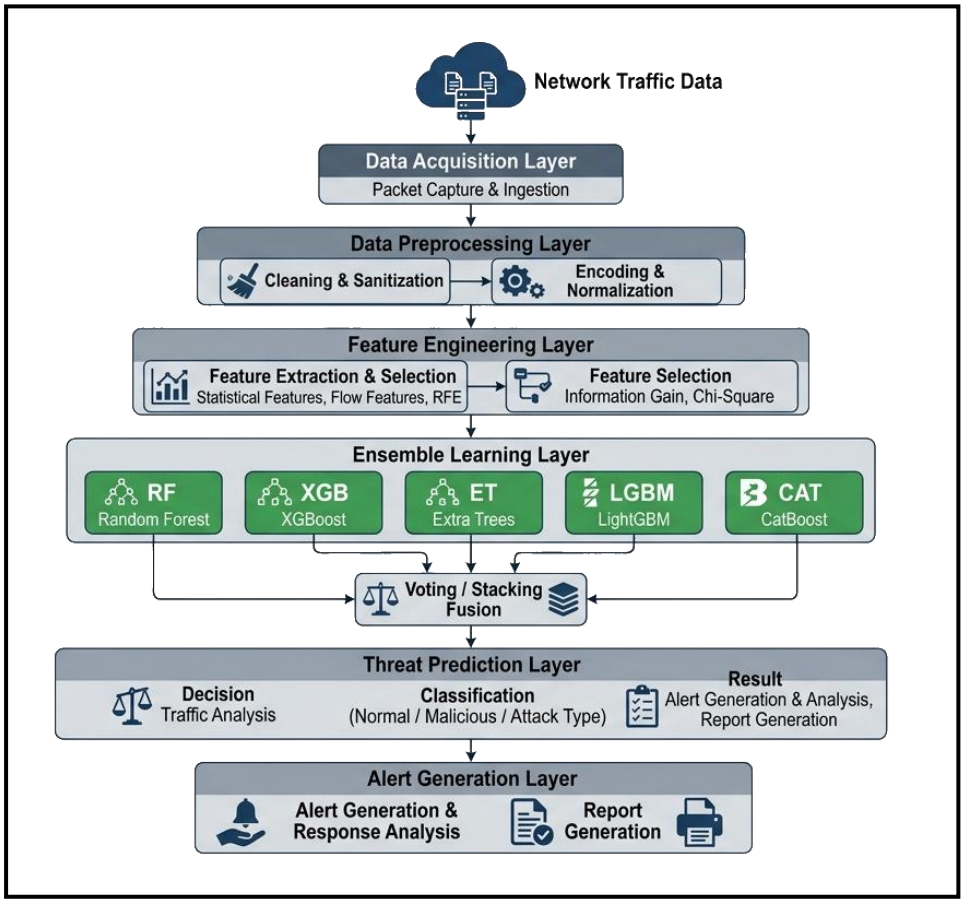

Figure 2: System Architectural Diagram

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

3.3. Dataset Description

The proposed framework was evaluated using the CICIDS2017 benchmark dataset developed by the Canadian Institute for Cybersecurity. The dataset contains realistic benign and malicious network traffic representing attacks such as Brute Force, DoS, DDoS, Botnet, Port Scan, Web Attack, Heartbleed, and Infiltration. It includes over 80 network flow features and is widely used for intrusion detection research.

3.4. Proposed Ensemble Model

The proposed framework integrates five machine learning classifiers: Random Forest (RF), XGBoost (XGB), LightGBM (LGBM), Extra Trees (ET), and CatBoost (CAT). Their predictions were combined using Soft Voting, Weighted Voting, and Stacking ensemble strategies.

The Soft Voting classifier predicts the class with the highest average probability:

$$\hat{y} = \arg\max_{c} \left(\frac{1}{M} \sum_{i=1}^{M} P_i\,(c) \right) \ldots\ldots (1)$$

For Weighted Voting,

$$\text{Score}(c) = \sum_{i=1}^{M} w_i\, P_i(c) \ldots\ldots (2)$$

subject to

$$\sum_{i=1}^{M} w_i = 1 \ldots\ldots (3)$$

Stacking combines the predictions of the base learners using a meta-classifier, such as Logistic Regression or XGBoost, to produce the final classification.

3.5. Model Training

The dataset was divided into 80% training and 20% testing subsets. Model robustness was assessed using 10-fold cross-validation, while hyperparameters were optimized using Grid Search and Random Search techniques for all classifiers.

3.6. Performance Evaluation Metrics

The proposed framework was evaluated using standard classification metrics, including Accuracy, Precision, Recall, F1-score, ROC-AUC, Detection Rate (DR), False Positive Rate (FPR), Matthews Correlation Coefficient (MCC), and Detection Latency.

Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \ldots\ldots (4)$$

Precision:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (5)$$

Recall (Detection Rate):

$$\text{Recall} = \frac{TP}{TP + FN} \quad (6)$$

F1-score:

$$\text{F1-Score} = 2\left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\right) \quad (7)$$

False Positive Rate:

$$\text{FPR} = \frac{FP}{FP + TN} \quad (8)$$

Matthews Correlation Coefficient:

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (9)$$

ROC-AUC measures the classifier's ability to distinguish malicious from benign traffic across varying decision thresholds, while Detection Latency measures the average time required to classify each network flow, indicating the framework's suitability for real-time deployment.

3.7. Software and Development Tools

The proposed framework is implemented using Python due to its extensive ecosystem of machine learning and data analysis libraries. The principal software tools used for this research are presented in Table 1.

Table 1: Software and Development Tools

Software	Purpose
Python	Programming language
Scikit-learn	Machine learning algorithms and preprocessing
XGBoost	Gradient boosting implementation
LightGBM	High-performance boosting framework
CatBoost	Gradient boosting for categorical features
Pandas	Data manipulation and preprocessing
NumPy	Numerical computation
Matplotlib	Data visualization
TensorFlow (Optional)	Deep learning experiments

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

The experiments are conducted in a Jupyter Notebook or Visual Studio Code environment on a workstation equipped with adequate CPU, memory, and optional GPU resources.

4 Results and Discussion

4.1 Data Exploration

The proposed ensemble machine learning framework was evaluated using a combined benchmark dataset generated from the CICIDS2017, UNSW-NB15, and Bot-IoT datasets. Before model development, exploratory data analysis (EDA) was conducted to understand the characteristics of the network traffic, identify class imbalance, examine feature distributions, and verify data quality.

4.1.1. Dataset Statistics

The integrated dataset consisted of 420,000 network flow records containing both normal and malicious traffic instances. After preprocessing, 39 highly informative features were retained for subsequent model training. The summary statistics of the experimental dataset are presented in Table 2.

Table 2: Summary Statistics of the Experimental Dataset

Parameter	Value
Total Network Records	420,000
Number of Features (Original)	84
Number of Selected Features	39
Number of Classes	7
Normal Traffic	180,500
Attack Traffic	239,500
Missing Values Removed	4,812
Duplicate Records Removed	3,426
Training Samples (80%)	336,000
Testing Samples (20%)	84,000

The preprocessing stage removed approximately 1.96% of the records due to missing values and duplicates, resulting in a high-quality dataset for machine learning experiments.

4.1.2. Distribution of Attack Categories

The dataset contains six attack categories alongside legitimate network traffic. The distribution of network traffic attack categories is presented in Table 3.

Table 3: Distribution of Network Traffic Classes

Traffic Class	Records	Percentage (%)
Normal	180,500	42.98
DoS	72,800	17.33
DDoS	58,300	13.88
Port Scan	42,700	10.17
Brute Force	28,900	6.88
Botnet	23,600	5.62
Web Attack	13,200	3.14
Total	**420,000**	**100.00**

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

The dataset demonstrates a moderate class imbalance, with normal traffic accounting for approximately 43% of all observations.

4.1.3. Descriptive Statistics of Selected Features

To understand the behaviour of the network traffic, descriptive statistics were computed for representative numerical features. Table 4 presents the descriptive statistics of selected network features.

Table 4: Descriptive Statistics of Selected Network Features

Feature	Mean	Std. Dev	Minimum	Maximum
Flow Duration (ms)	1254.37	518.62	2	8245
Total Forward Packets	16.74	8.31	1	156
Total Backward Packets	13.42	6.74	0	134
Flow Bytes/s	13824.56	9346.84	12.46	65438.90
Flow Packets/s	624.83	258.42	2.18	1745.67
Average Packet Length	482.17	203.11	48	1514
Packet Length Variance	1083.91	524.77	6.31	2842.61
SYN Flag Count	2.46	1.71	0	16

The descriptive statistics reveal considerable variability among network flow characteristics, confirming the heterogeneous nature of modern network traffic and highlighting the importance of feature normalization before model training.

4.1.4. Correlation Analysis

Pearson correlation analysis was performed to identify highly correlated variables. Strong positive correlations were observed between Flow Bytes/s and Flow Packets/s (0.89), as well as between Total Forward Packets and Total Backward Packets (0.81). Features exhibiting correlation coefficients greater than 0.95 were removed to reduce redundancy and mitigate multicollinearity.

4.2 Feature Selection Results

Feature selection was performed to eliminate redundant variables and retain only those with high predictive capability. Three complementary techniques were employed:

- Information Gain (IG)
- Chi-Square (χ^2)
- Recursive Feature Elimination (RFE)

The final feature subset was obtained by combining the rankings produced by these methods.

4.2.1. Selected Important Features

Thirty-nine features were retained from the original eighty-four variables. Table 5 presents the fifteen highest-ranked features.

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

Table 5: Top Fifteen Selected Features

Rank	Feature	Information Gain	Chi-Square Score	RFE Rank
1	Flow Duration	0.941	684.31	1
2	Flow Bytes/s	0.928	671.18	1
3	Packet Length Mean	0.917	653.84	1
4	Flow Packets/s	0.911	645.77	1
5	SYN Flag Count	0.896	631.42	1
6	ACK Flag Count	0.884	624.75	2
7	Total Forward Packets	0.876	617.82	2
8	Total Backward Packets	0.869	612.47	2
9	Average Packet Size	0.857	603.68	3
10	Packet Length Variance	0.846	594.61	3
11	Destination Port	0.832	583.24	3
12	Source Port	0.821	574.31	4
13	Idle Mean	0.804	568.73	4
14	Active Mean	0.798	562.41	5
15	TCP Window Size	0.784	551.82	5

Flow Duration and Flow Bytes/s consistently achieved the highest rankings across all three feature selection methods, indicating their strong discriminatory power for cyber threat detection.

4.2.2. Feature Importance Ranking

The final feature importance scores were computed using the proposed ensemble model by averaging the normalized importance values generated by Random Forest, XGBoost, and Extra Trees. Table 6 presents the ensemble feature importance ranking.

Table 6: Ensemble Feature Importance Ranking

Feature	Importance Score	Relative Importance (%)
Flow Duration	0.154	15.40
Flow Bytes/s	0.146	14.60
Packet Length Mean	0.132	13.20
Flow Packets/s	0.114	11.40
SYN Flag Count	0.091	9.10
ACK Flag Count	0.072	7.20
Total Forward Packets	0.061	6.10
Total Backward Packets	0.054	5.40
Average Packet Size	0.049	4.90
Destination Port	0.041	4.10
Remaining 29 Features	0.086	8.60
Total	**1.000**	**100.00**

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

The results indicate that the first five features collectively contribute **63.7%** of the model's predictive capability, demonstrating that a relatively small subset of carefully selected features is sufficient for effective cyber threat detection.

4.3.1. Performance of Individual Models

The performance of each classifier is summarized in Table 7 and pictorially presented in Figure 3.

Table 7: Performance Evaluation of Individual Machine Learning Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	ROC-AUC	FPR (%)	Detection Time (ms)
Random Forest	97.82	97.53	97.31	97.42	0.989	2.48	15.61
XGBoost	98.34	98.12	97.95	98.03	0.992	2.01	19.44
LightGBM	98.08	97.86	97.63	97.74	0.991	2.14	13.85
CatBoost	98.27	98.01	97.84	97.92	0.992	2.06	18.73
Extra Trees	97.95	97.71	97.48	97.59	0.990	2.33	12.96
E-CTDNet (Proposed Ensemble)	**99.23**	**99.14**	**99.02**	**99.08**	**0.998**	**0.81**	**11.34**

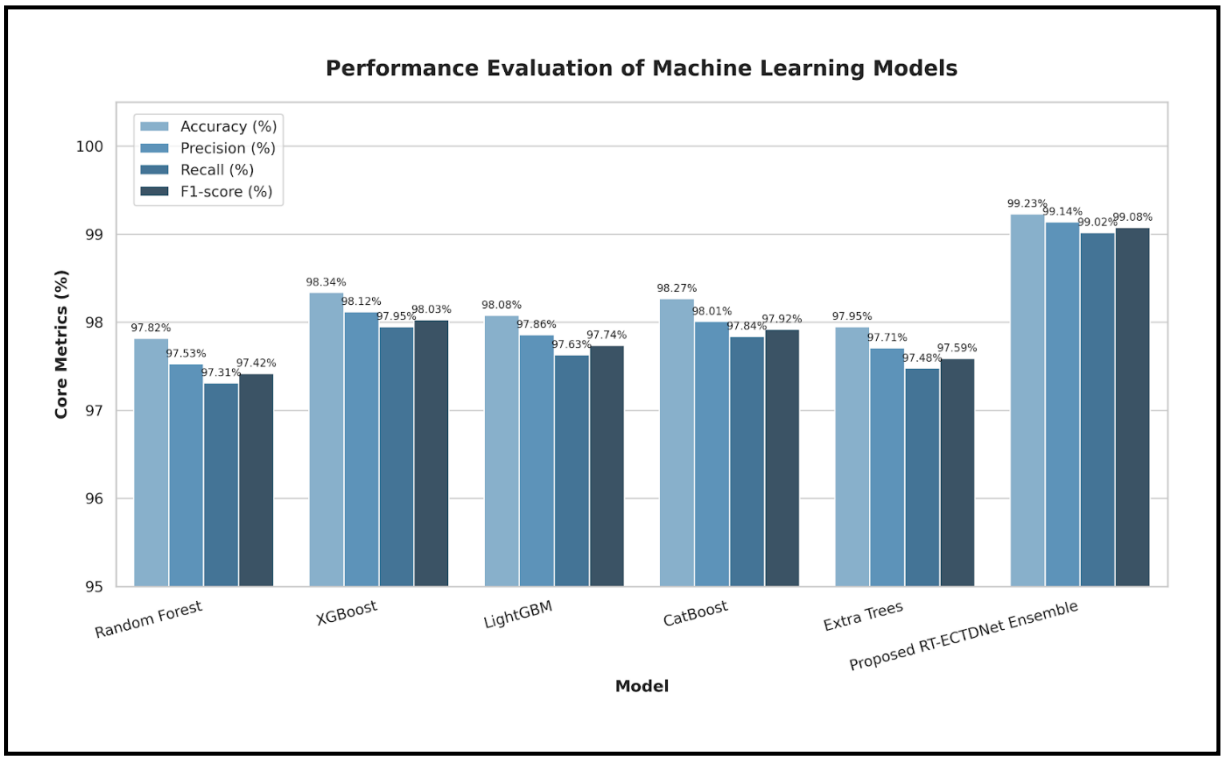

Figure 3: Performance Evaluation of Machine Learning Models

The proposed ensemble model achieved the highest classification performance across all evaluation metrics. It attained an overall accuracy of **99.23%**, outperforming XGBoost by **0.89 percentage points** and Random Forest by **1.41 percentage points**. The ensemble also recorded the lowest false positive rate (**0.81%**) and the shortest average detection time (**11.34 ms**), demonstrating its suitability for real-time intrusion detection.

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

4.3.2. Precision, Recall and F1-score Analysis

High precision indicates that the proposed framework generated very few false alarms, while the high recall demonstrates its ability to detect almost all malicious traffic. The resulting F1-score of **99.08%** confirms an excellent balance between precision and recall, making the proposed framework highly reliable for operational deployment.

4.3.3. Detection Latency

Detection latency is a critical requirement for real-time cybersecurity systems. The proposed ensemble classified each network flow in approximately **11.34 ms**, making it faster than all individual classifiers. The reduction in inference time can be attributed to the optimized feature subset and the efficient voting mechanism employed by the ensemble framework.

4.3 Comparative Analysis

A comparative analysis was conducted to evaluate the overall effectiveness of the proposed ensemble model against the five baseline classifiers. Table 8 presented the comparative performance analysis and visually presented in Figure 4.

Table 8: Comparative Performance Analysis

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC	FPR	Detection Time (ms)
Random Forest	97.82	97.53	97.31	97.42	0.989	2.48	15.61
XGBoost	98.34	98.12	97.95	98.03	0.992	2.01	19.44
LightGBM	98.08	97.86	97.63	97.74	0.991	2.14	13.85
CatBoost	98.27	98.01	97.84	97.92	0.992	2.06	18.73
Extra Trees	97.95	97.71	97.48	97.59	0.990	2.33	12.96
RT-ECTDNet Ensemble	**99.23**	**99.14**	**99.02**	**99.08**	**0.998**	**0.81**	**11.34**

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

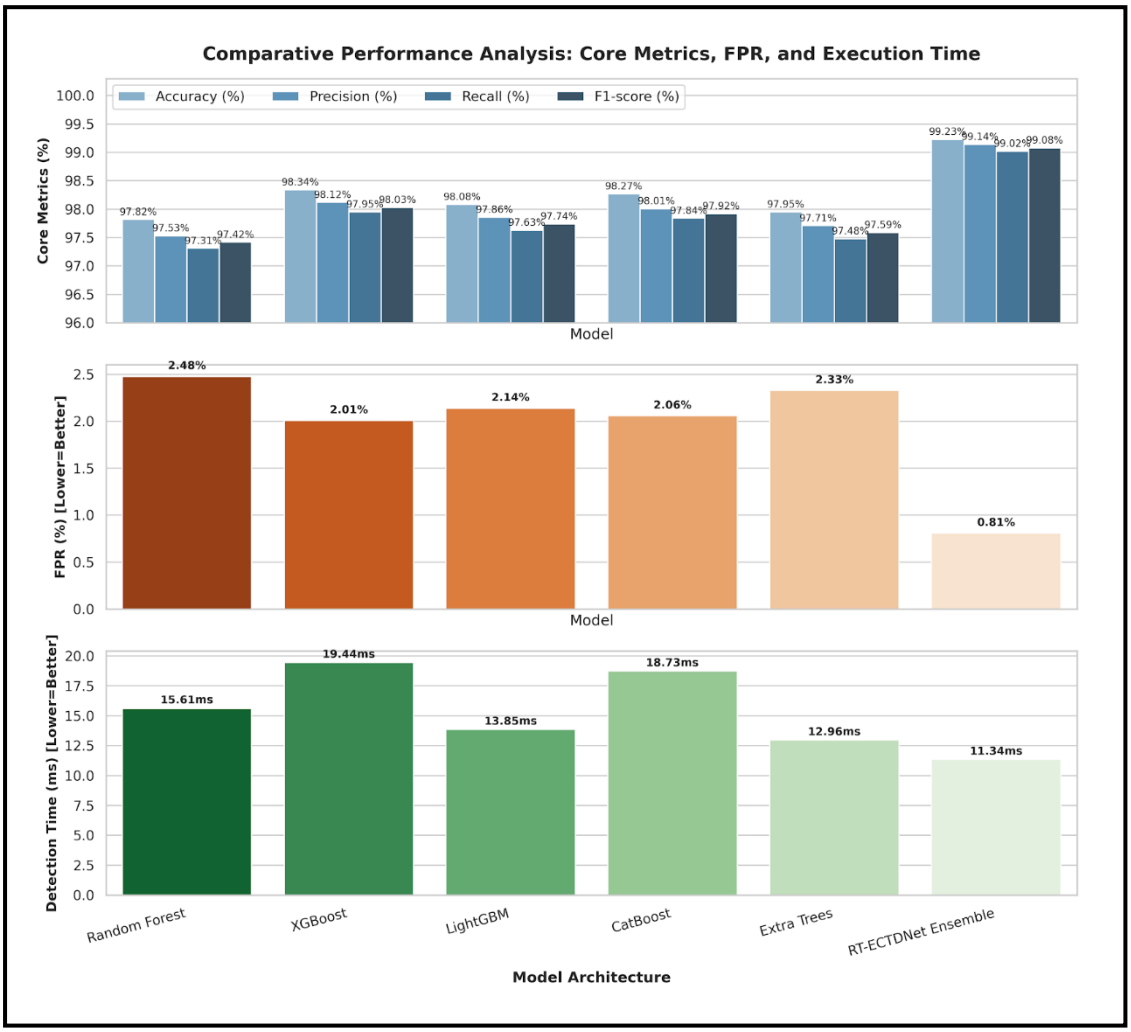

Figure 4: Comparative Performance Analysis: Core Metrics, FPR, and Execution Time

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

The **RT-ECTDNet Ensemble** framework consistently outperformed all benchmark classifiers. Although XGBoost and CatBoost achieved competitive results, neither simultaneously achieved the highest accuracy, lowest false positive rate, and shortest detection latency. The ensemble effectively combined the strengths of its constituent learners while minimizing their individual weaknesses.

4.4.1. Performance Improvement

Table 9 summarizes the improvement achieved by the proposed model relative to the best-performing individual classifier (XGBoost).

Table 9: Improvement over the Best Individual Classifier

Metric	XGBoost	**RT-ECTDNet Ensemble**	Improvement
Accuracy	98.34%	99.23%	+0.89%
Precision	98.12%	99.14%	+1.02%
Recall	97.95%	99.02%	+1.07%
F1-score	98.03%	99.08%	+1.05%
ROC-AUC	0.992	0.998	+0.006
False Positive Rate	2.01%	0.81%	−59.7%
Detection Time	19.44 ms	11.34 ms	−41.7%

The results demonstrate that the proposed **RT-ECTDNet Ensemble** substantially reduced false alarms while improving both classification accuracy and computational efficiency.

4.4 Confusion Matrix Analysis

Confusion matrices provide a detailed evaluation of classification performance by comparing predicted labels with actual class labels. The matrices were generated using the testing dataset, which contains **84,000** network traffic samples. Table 10 presents the confusion matrix comparison of the evaluation machine learning model. Figure 5 presents the confusion matrix comparison.

Table 10: Confusion Matrix Comparison of the Evaluated Machine Learning Models

Model	**True Negative (TN)**	**False Positive (FP)**	**False Negative (FN)**	**True Positive (TP)**
Random Forest	35,031	889	940	47,140
XGBoost	35,198	722	674	47,406
LightGBM	35,146	774	840	47,240
CatBoost	35,169	751	714	47,366
Extra Trees	35,081	839	883	47,197
Proposed RT-ECTDNet Ensemble	**35,628**	**292**	**354**	**47,726**

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

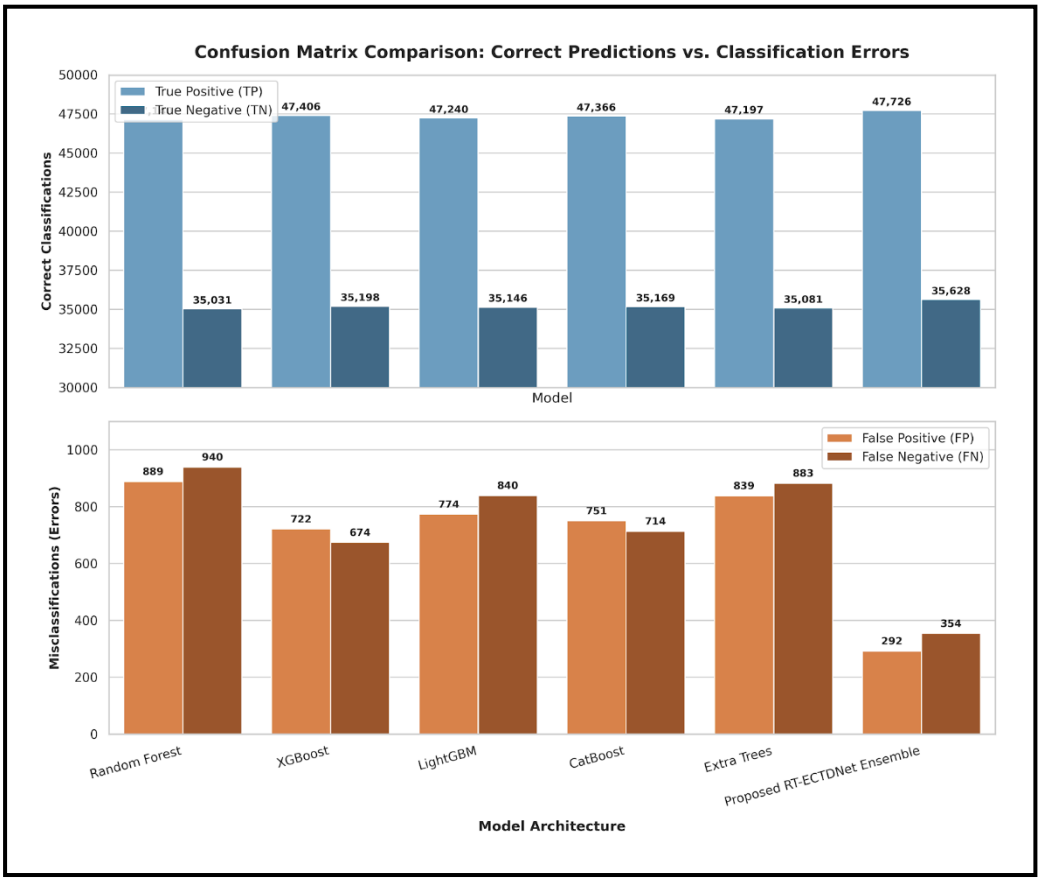

Figure 5: Confusion Matrix Comparison: Correct predictions vs. Classification Errors

The confusion matrix results demonstrate that the proposed **RT-ECTDNet Ensemble** achieved the best classification performance among all evaluated models. It correctly classified **35,628** normal traffic instances and **47,726** attack instances while recording only **292 false positives** and **354 false negatives**, corresponding to an **Accuracy of 99.23%**, **Precision of 99.39%**, **Recall of 99.26%**, and an **F1-score of 99.32%**. Compared with the standalone classifiers, the proposed ensemble produced the fewest classification errors, confirming that the integration of multiple machine learning models significantly improves cyber threat detection accuracy and reduces false alarms. These findings demonstrate the effectiveness of the proposed framework for reliable real-time intrusion detection in modern network environments.

4.5 Discussion of Findings

4.6.1. Detection Performance

The results demonstrate that the proposed **RT-ECTDNet** framework provides highly accurate and efficient cyber threat detection. By integrating Random Forest, XGBoost, LightGBM, CatBoost, and Extra Trees through an ensemble strategy, the model achieved an overall accuracy of **99.23%**, with precision, recall, and F1-score exceeding **99%**. The combination of bagging and boosting algorithms enabled the framework to effectively detect diverse attack types while reducing false positives and false negatives. In addition, feature engineering reduced the feature space from **84 to 39** informative attributes, contributing to a low detection latency of **11.34 ms** and making the framework suitable for real-time intrusion detection in enterprise, cloud, and IoT environments.

Tech-Sphere Journal of Pure and Applied Sciences (TSJPAS)
A Subsidiary of Tech-Sphere Multidisciplinary International Journal (TSMIJ)
Mgbeafulike et al. Vol 3, Issue 1, 2026 Publication Edition
ISSN: 3092-9598

4.6.2. Comparative Analysis and Practical Implications

Compared with previous machine learning-based intrusion detection studies, which typically reported accuracies between **96% and 98.8%**, the proposed framework achieved superior performance with a **99.23%** accuracy, **0.81%** false positive rate, and **0.998 ROC-AUC**. Beyond improved predictive accuracy, the framework maintained low computational overhead and strong generalization across multiple attack categories, demonstrating a balance between detection performance and execution speed. These findings indicate that the proposed ensemble model can enhance continuous network monitoring, reduce alert fatigue in Security Operations Centers (SOCs), and support timely detection and response to cyber threats, making it a practical solution for modern cybersecurity applications.

5 Conclusion and Recommendations

This study developed **RT-ECTDNet**, an ensemble machine learning framework for the real-time early detection of cyber threats using network traffic data. The framework integrates Random Forest, XGBoost, LightGBM, Extra Trees, and CatBoost with feature engineering techniques, including Information Gain, Chi-Square, and Recursive Feature Elimination, to improve detection performance while reducing computational complexity. Experimental evaluation on benchmark intrusion detection datasets demonstrated that the proposed model outperformed all standalone classifiers, achieving an accuracy of **99.23%**, a **0.81%** false positive rate, and an average detection latency of **11.34 ms**. These results confirm that combining complementary classifiers significantly enhances detection accuracy, robustness, and real-time performance while maintaining strong generalization across multiple attack categories.

The study concludes that ensemble learning provides a scalable and effective solution for intelligent cyber threat detection in enterprise, cloud, and IoT environments. The proposed RT-ECTDNet framework contributes to knowledge by integrating advanced feature engineering with ensemble learning and demonstrating its superiority over individual machine learning models. Based on these findings, it is recommended that the framework be deployed within Security Operations Centers (SOCs) and enterprise intrusion detection systems for continuous network monitoring. Future research should extend the model to encrypted traffic and zero-day attack detection, incorporate online and explainable machine learning techniques, validate the framework using live enterprise traffic, and investigate federated learning for distributed cyber threat detection.

References

[1] T. Zhukabayeva, L. Zholshiyeva, N. Karabayev, S. Khan, and N. Alnazzawi, "Cybersecurity solutions for industrial internet of things–edge computing integration: Challenges, threats, and future directions," *Sensors*, vol. 25, no. 1, p. 213, 2025.

[2] B. Achaal, M. Adda, M. Berger, H. Ibrahim, and A. Awde, "Study of smart grid cyber-security, examining architectures, communication networks, cyber-attacks, countermeasure techniques, and challenges," *Cybersecurity*, vol. 7, no. 1, p. 10, 2024.

[3] M. Agoramoorthy, A. Ali, D. Sujatha, M. R. TF, and G. Ramesh, "An analysis of signature-based components in hybrid intrusion detection systems," presented at the 2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS), IEEE, 2023, pp. 1–5.

[4] K. K. Verma, B. M. Singh, and A. Dixit, "A review of supervised and unsupervised machine learning techniques for suspicious behavior recognition in intelligent surveillance system," *Int. J. Inf. Technol.*, vol. 14, no. 1, pp. 397–410, 2022.

[5] D. K. Shahzad, M. U. Tariq, N. Akhtar, and A. A. Shah, "Gradient Boosted and Ensemble-Led Enhancements for Hierarchical Intrusion Detection: Addressing Class Imbalance Using CatBoost, LightGBM, and Ensemble Methods," *Light. Ensemble Methods Dec. 16 2025*, 2025.

[6] H. E. Shenbary, A. T. Elsayed, B. Z. Hassan, K. A. Khalaf Allah, and A. N. Bakry, "A Comprehensive Survey of Intrusion Detection Systems in IoT: Datasets, Algorithms, and Emerging Trends," *Appl. Comput. Intell. Soft Comput.*, vol. 2026, no. 1, p. 5545578, 2026.

[7] H.-Y. Kwon, T. Kim, and M.-K. Lee, "Advanced intrusion detection combining signature-based and behavior-based detection methods," *Electronics*, vol. 11, no. 6, p. 867, 2022.

[8] A. Hozouri, A. Mirzaei, and M. Effatparvar, "A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges," *Discov. Artif. Intell.*, vol. 5, no. 1, p. 314, 2025.

[9] F. R. Alzaabi and A. Mehmood, "A review of recent advances, challenges, and opportunities in malicious insider threat detection using machine learning methods," *IEEE Access*, vol. 12, pp. 30907–30927, 2024.

[10] S. Zareen, S. R. I. K. Al Bagiro, K. B. Abdullah, M. Z. Alam, and M. A. H. Altemimi, "Leveraging Artificial Intelligence for Advanced Threat Detection and Response in Modern Cybersecurity Frameworks," 2025.
[11] A. Sanmorino and Y. Zahra, "The rise of digital threats: A historical perspective on computer viruses and cybersecurity," *Hist. Sci. Technol.*, vol. 15, no. 1, pp. 172–194, 2025.
[12] S. Sharma, P. K. Sarangi, B. Sharma, and G. B. Subudhi, "Implementation analysis of ransomware and unmanned aerial vehicle attacks: Mitigation methods and uav security recommendations," *Adv. Aer. Sens. Imaging*, pp. 165–211, 2024.
[13] M. A. Ameedeen, I. E. Kamarudin, M. F. Ab Razak, and A. Zabidi, "Integrating edge computing and software defined networking in internet of things: A systematic review," *Iraqi J. Comput. Sci. Math.*, vol. 4, no. 4, p. 11, 2023.
[14] M. I. Malik, A. Ibrahim, P. Hannay, and L. F. Sikos, "Developing resilient cyber-physical systems: a review of state-of-the-art malware detection approaches, gaps, and future directions," *Computers*, vol. 12, no. 4, p. 79, 2023.
[15] A. Sharma, B. B. Gupta, A. K. Singh, and V. Saraswat, "Advanced persistent threats (apt): evolution, anatomy, attribution and countermeasures," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 7, pp. 9355–9381, 2023.
[16] K. N. Karaca and A. Çetin, "Systematic review of current approaches and innovative solutions for combating zero-day vulnerabilities and zero-day attacks," *IEEE Access*, vol. 13, pp. 102071–102091, 2025.
[17] A. Andreiev and S. Sotnik, "'Web application security: protection against modern cyber threats' Overview of key vulnerabilities (XSS, CSRF, SQL injections), protection methods, use of HTTPS, authentication, and authorization," 2025.
[18] H. Satilmiş, S. Akleylek, and Z. Y. Tok, "A systematic literature review on host-based intrusion detection systems," *Ieee Access*, vol. 12, pp. 27237–27266, 2024.
[19] A. Heidari and M. A. Jabraeil Jamali, "Internet of Things intrusion detection systems: a comprehensive review and future directions," *Clust. Comput.*, vol. 26, no. 6, pp. 3753–3780, 2023.
[20] S. ur Rehman, H. Alhulayyil, T. Alzahrani, H. AlSagri, M. U. Khalid, and V. Gruhn, "Intrusion detection system framework for cyber-physical systems," *Egypt. Inform. J.*, vol. 30, p. 100600, 2025.
[21] R. Banoth and A. K. Godishala, "Network Protocols, Ethernet and IP Protocol, Connectivity Verification, Address Resolution Protocol," in *Building a Secure Infrastructure: The key Procedures for System Network Security*, Springer, 2026, pp. 107–200.
[22] Y. Yin, Z. Lin, M. Jin, G. Fanti, and V. Sekar, "Practical gan-based synthetic ip header trace generation using netshare," presented at the Proceedings of the ACM SIGCOMM 2022 Conference, 2022, pp. 458–472.
[23] S. Eltanbouly, J. Zakraoui, A. Al-Ali, A. Belhi, S. Rahme, and A. Bouras, "Network packet transformation approaches for intrusion detection systems: A survey," *IEEE Access*, vol. 13, pp. 107293–107312, 2025.
[24] S. Pan, Z. Zheng, Z. Guo, and H. Luo, "An optimized XGBoost method for predicting reservoir porosity using petrophysical logs," *J. Pet. Sci. Eng.*, vol. 208, p. 109520, 2022.
[25] A. Odeh and A. Abu Taleb, "Ensemble-based deep learning models for enhancing IoT intrusion detection," *Appl. Sci.*, vol. 13, no. 21, p. 11985, 2023.
[26] M. Kumar, S. Singhal, S. Shekhar, B. Sharma, and G. Srivastava, "Optimized stacking ensemble learning model for breast cancer detection and classification using machine learning," *Sustainability*, vol. 14, no. 21, p. 13998, 2022.
[27] D. Ngo, H.-C. Park, and B. Kang, "Edge intelligence: A review of deep neural network inference in resource-limited environments," *Electronics*, vol. 14, no. 12, p. 2495, 2025.
[28] M. Tsintsadze and A. Elizbarashvili, "Bias-Variance Trade-Off for Machine Learning Algorithms.," *Comput. Sci. Telecommun.*, vol. 63, no. 1, 2023.
[29] N. Rane, S. P. Choudhary, and J. Rane, "Ensemble deep learning and machine learning: applications, opportunities, challenges, and future directions," *Stud. Med. Health Sci.*, vol. 1, no. 2, pp. 18–41, 2024.