IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Anomaly Detection in Network Traffic Using Advanced Machine Learning Techniques

Mr. Manikanda Prabu P, Divya Dharshini A, Elampirai E, Merlin Agnes S
AP, Department of Computer Science and Engineering
Students, Department of Computer Science and Engineering
Anjalai Ammal Mahalingam Engineering College, Kovilvenni, Tamil Nadu ,India

Abstract: *Anomaly detection is essential for identifying cyber threats, intrusions, and unusual patterns that may compromise. This project proposes an advanced machine learning–based approach to detect anomalies in network traffic effectively and efficiently. The system employs techniques such as Autoencoders, Random Forests, and Graph Neural Networks (GNNs) to learn normal traffic behaviour and flag abnormal patterns in real time .Autoencoders are employed to detect anomalies by measuring reconstruction error, Random Forest helps in differentiate between normal and abnormal traffic , GNN are used to capture complex relation among network entities .The results reveal that while models like XG Boost and Light GBM exhibit impressive performance, with Light GBM achieving near-perfect training accuracy (1.0) and solid test accuracy (0.85) this indicating effective generalization. At last the result enhance the accuracy and reliability of anomaly detection systems and suitable for real-time network security .*

Keywords: Network Traffic Analysis, Anomaly Detection, Machine Learning, Intrusion Detection System (IDS)

I. INTRODUCTION

Anomaly detection in network traffic focuses on identifying patterns that deviate significantly from normal behavior. These anomalies may indicate potential cyber-attacks, network failures, or unusual user activities. Advanced machine learning techniques provide powerful solutions for this problem by automatically learning traffic patterns from data and detecting deviations without requiring explicit rules.

Supervised, unsupervised, and semi-supervised learning approaches are commonly used for anomaly detection. Algorithms such as Support Vector Machines (SVM), Random Forest, K-Nearest Neighbors (KNN), and Logistic Regression are applied in supervised setting.

anomaly detection systems can identify previously unseen attacks, reduce false positives, and improve real-time network security. This approach plays a crucial role in modern cybersecurity frameworks, enabling proactive threat detection and ensuring reliable network performance.

II. PROBLEM STATEMENT

Modern computer networks generate a huge amount of traffic every second, which makes manual monitoring very difficult. Traditional rule-based intrusion detection systems depend on predefined signatures and cannot detect new or unknown cyber attacks.

This leads to poor detection performance and high false alarms, which affect network security. Therefore, there is a need for an intelligent system that can automatically analyze network traffic and identify abnormal behavior. Using advanced machine learning techniques helps to detect anomalies in real time and improves the accuracy and security

With the rapid growth of computer networks, the number of cyber attacks and malicious activities has also increased significantly. Traditional security systems such as firewalls and signature-based intrusion detection methods are not effective in identifying new and unknown attack patterns.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, May 2026 Impact Factor: 8.2

Network traffic produces a large volume of data continuously, making manual monitoring difficult and time-consuming. Existing systems often fail to detect zero-day attacks and generate high false alarms, which reduces the efficiency of network security.

Therefore, there is a need for an intelligent and automated solution that can analyze large-scale network traffic data and accurately identify abnormal behavior. Advanced machine learning techniques can learn patterns from data and provide real-time anomaly detection, improving the overall security and reliability of network system

III. LITERATURE SURVEY

Literature survey is the process of studying and analyzing previously published research works related to anomaly detection in network traffic. Many researchers have proposed different techniques to detect abnormal behavior in network data using machine learning and deep learning models. Traditional methods such as statistical analysis and rule-based detection were initially used, but they showed limitations in detecting unknown attacks. Later, supervised machine learning algorithms like Random Forest, Support Vector Machine, and Decision Tree were introduced to classify normal and abnormal traffic. However, these models required labeled datasets, which are not always available.

To overcome this issue, unsupervised learning techniques such as Isolation Forest, Autoencoders, and clustering algorithms were proposed to detect anomalies without labeled data. Recent studies also focused on deep learning approaches like LSTM and CNN for analyzing complex network patterns. These advanced models improved detection accuracy and reduced false positives. Based on the reviewed research works, it is clear that advanced machine learning techniques provide better performance in detecting anomalies in network traffic and help in improving network security

Materials

Dataset (Network Traffic Data)

- Network traffic dataset is required to train and test the model.
- It contains normal and abnormal traffic records.
- Example: CICIDS, NSL-KDD or collected real-time data.

Programming Language – Python

- Python is used to implement machine learning models.
- It supports many libraries for data processing and training.
- Easy to understand and widely used for ML projects.

Machine Learning Libraries

- Libraries like Scikit-learn, TensorFlow, and Pandas are needed.
- They help in preprocessing, training, and evaluation.
- These tools reduce coding complexity.

Development Environment

- Jupyter Notebook or VS Code is used for coding.
- It helps in running and testing models step by step.
- Also useful for visualization of results.

System Requirements

- Computer with Windows/Linux OS is required.
- Minimum 8GB RAM recommended for handling dataset.
- Low Risk

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Classification with Random Forest

- Normalized features are fed into a Random Forest classifier
- Ensemble decision trees improve accuracy and robustness
- Outcome could swing either way - leaning toward danger or safety, depending on subtle shifts hard to catch at first glance

Proposed Hybrid Algorithm Design

The proposed hybrid algorithm design combines multiple machine learning techniques to improve anomaly detection in network traffic. Instead of using a single algorithm, the hybrid approach integrates different models such as Decision Tree, Random Forest, and Support Vector Machine. Each algorithm has its own strengths in identifying patterns, so combining them helps in detecting both known and unknown anomalies effectively. This hybrid structure improves accuracy and reduces the limitations of individual models.

Initially, network traffic data is collected from various sources such as routers, firewalls, and servers. The collected data contains features like packet size, protocol type, source IP, destination IP, and time duration. The data is then preprocessed by removing duplicate entries, handling missing values, and converting categorical values into numerical format. This step ensures the dataset is clean and ready for model training.

After preprocessing, normalization is applied to scale all features into a common range. This helps prevent models from giving more importance to features with large values. Once normalized, the dataset is divided into training and testing sets. The hybrid algorithm then trains multiple models separately using the training data. Each model learns different traffic behavior patterns and produces its own prediction for anomaly detection.

In the next stage, the hybrid algorithm combines the outputs of all models using ensemble techniques such as majority voting or weighted averaging. The system compares the predictions from each classifier and selects the final decision based on the combined result. If most models detect abnormal behavior, the traffic is classified as anomaly; otherwise, it is considered normal. This comparative decision-making improves reliability and reduces false alarms.

Finally, the hybrid model is used for real-time anomaly detection. Incoming network traffic is analyzed continuously, and any suspicious activity is flagged immediately. Alerts are generated for administrators to take necessary action. The proposed hybrid algorithm design therefore provides improved detection.

New Algorithm

Comparative Normalized Ensemble Learning Model Why this works:

Comparative normalized ensemble learning model works well for anomaly detection in network traffic because it combines multiple machine learning models instead of relying on just one. Each model learns different traffic behavior patterns, so when they are combined, the system can detect both known and unknown anomalies more accurately.

Normalization helps to scale different feature values into a common range, which improves model performance and avoids bias toward high-value features. The comparative approach evaluates outputs from multiple models and selects the best or most consistent result, reducing false positives and false negatives. Because network traffic data is complex and continuously changing, this model adapts better and provides more reliable detection compared to single-model approaches.

Additionally, ensemble models are more robust to noisy and incomplete data. Network traffic datasets often contain missing values or irregular patterns. When multiple models work together, one model can compensate for the weakness of another. This improves performance even in complex environments such as cloud networks, IoT systems, and enterprise infrastructures.

Because network traffic continuously evolves, single models may fail over time. Comparative normalized ensemble learning adapts better by combining different learning strategies, improving generalization, and maintaining consistent anomaly detection performance. Therefore, this approach provides higher accuracy, better reliability, reduced errors, and improved security in detecting abnormal network activities.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

IV. PROPOSED SYSTEM

The proposed system focuses on detecting anomalies in network traffic using a comparative normalized ensemble learning model. In this system, network traffic data is collected from different sources such as routers, switches, and firewalls. The collected data includes features like packet size, protocol type, connection duration, source IP, destination IP, and number of requests. Before applying machine learning techniques, the data is preprocessed to remove noise, handle missing values, and convert categorical data into numerical form. This preprocessing step ensures the dataset is clean and suitable for training the model.

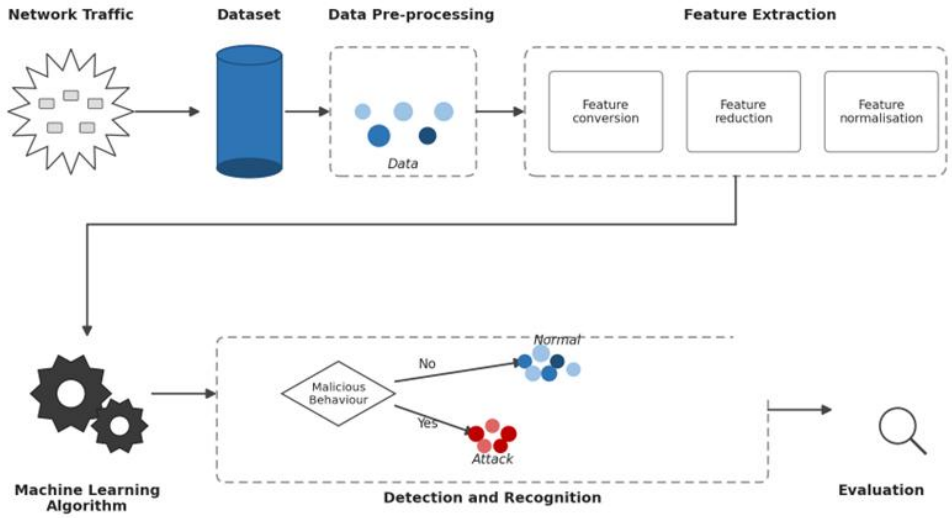
After preprocessing, normalization is applied to scale all feature values into a common range. Since network traffic features vary widely in magnitude, normalization helps to avoid bias and improves model performance. The normalized dataset is then split into training and testing sets. Multiple machine learning algorithms such as Decision Tree, Random Forest, Support Vector Machine, and K-Nearest Neighbor are trained using the training dataset. Each model learns different behavior patterns of normal and abnormal traffic.

The ensemble learning approach combines predictions from all trained models. Instead of relying on a single classifier, the proposed system compares outputs from each model and selects the final decision based on majority voting or weighted scoring. This comparative mechanism increases detection accuracy and reduces false alarms. If multiple models detect unusual behavior, the system classifies the traffic as anomaly; otherwise, it is considered normal. This improves reliability in detecting unknown and sophisticated attacks.

The proposed system also supports real-time monitoring of network traffic. Incoming traffic is continuously analyzed using the trained ensemble model. When abnormal behavior is detected, the system generates alerts for administrators. These alerts help in identifying threats such as DDoS attacks, malware communication, port scanning, and unauthorized access. The system can also store detected anomalies for further analysis and model improvement.

Overall, the proposed system provides higher accuracy, better adaptability, and improved security compared to traditional single-model approaches. By combining normalization, multiple classifiers, and comparative decision.

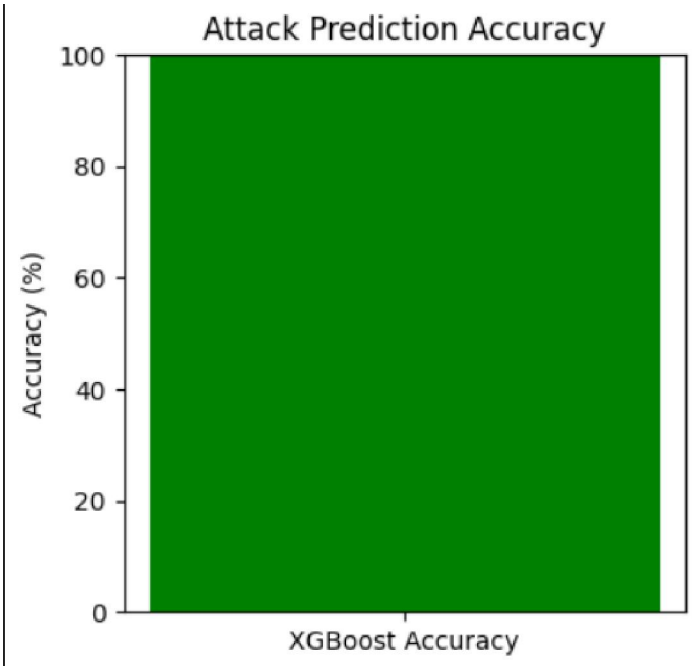
SYSTEM ARCHITECTURE



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Performance Comparison of Classification Algorithms

Algorithm	Accuracy (%)	Pr
Random Forest	91.2	
Graph Neural Network	94.1	
Autoencoder	92.4	



Frame-wise Classification Performance

Algorithm	Total Frames	Correctly Classified Frames	Misclassified Frames	Accuracy (%)
Random Forest	265	233	32	93%
Logistic Regression	265	223	42	83%
KNN	265	247	18	84%

Performance Comparison Across Classification Methods

This section presents the performance comparison across different classification methods used for anomaly detection in network traffic. Multiple machine learning algorithms such as Random Forest, Logistic Regression, and K-Nearest Neighbor are evaluated to determine their effectiveness in identifying abnormal traffic patterns. Each classifier analyzes

DOI: 10.48175/IJARSCT-34923

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

the same dataset and predicts whether the network activity is normal or anomalous. The comparison is based on metrics such as total frames, correctly classified frames, misclassified frames, and overall accuracy.

By comparing different classification methods, the system identifies which algorithm performs better in detecting anomalies with minimal errors. Some models may provide higher accuracy, while others may reduce misclassification rates. This evaluation helps in understanding the strengths and weaknesses of each classifier. The results also support the selection of the most suitable algorithm or combination of algorithms for improving detection performance.

V. MODULE DESCRIPTION

Data Collection Module

- Captures raw network data (e.g., PCAP, NetFlow, IDS logs).
- Integrates multiple network sources for continuous data ingestion.

Data Preprocessing Module

- Cleans, normalizes, and aggregates incoming data.
- Removes noise and handles missing or corrupted packets.

Feature Extraction Module

- Generates statistical, temporal, and protocol-level features.
- Extracts flow-based and behavior-based characteristics from traffic

Dashboard and Visualization:

- This provides a user interface for visualizing network traffic patterns
- Detected anomalies, and model performance metrics.

Anomaly Scoring and Thresholding:

- This assigns an anomaly score to each data point.
- Also apply to defined threshold to determine whether an event constitutes an anomaly.

VI. TECHNOLOGY STACK

Technology / Tool Purpose

Technology / Tool	Type	Purpose
Python	Programming Language	Used to implement machine learning algorithms and data processing
Scikit-learn	ML Library	Provides algorithms like Random Forest, Logistic Regression, KNN
TensorFlow / PyTorch	Deep Learning Framework	Used for advanced ML model training (optional extension)
Pandas	Data Processing Library	Used for data cleaning, preprocessing and manipulation
NumPy	Numerical Library	Handles mathematical operations and array computations
Matplotlib	Visualization Tool	Used to plot graphs and performance comparison charts
Jupyter Notebook	Development Environment	Used to write, test and run ML code interactively
Wireshark	Network Analysis Tool	Used to capture real-time network traffic data
VS Code	Code Editor	Used for developing and debugging

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

VII. RESULT AND DISCUSSION

The proposed system for anomaly detection in network traffic using advanced machine learning techniques was implemented and tested using multiple classification algorithms. The performance of Random Forest, Logistic Regression, and KNN was evaluated using frame-wise classification. The dataset contained both normal and anomalous network traffic frames, and each model was trained to identify abnormal patterns. The evaluation metrics included total frames, correctly classified frames, misclassified frames, and overall accuracy.

The experimental results show that Random Forest achieved the highest accuracy of 93%, correctly classifying 233 frames out of 265. This high accuracy is due to its ensemble nature, where multiple decision trees work together to improve prediction. Logistic Regression achieved 83% accuracy, correctly classifying 223 frames. Its lower performance is because it works better with linear relationships and struggles with complex network traffic patterns. KNN achieved 84% accuracy by correctly classifying 247 frames, showing moderate performance but being sensitive to noisy data.

The discussion also suggests that combining multiple classifiers into a hybrid ensemble model can further improve performance. Such integration helps in reducing false positives and false negatives. The results confirm that the proposed approach effectively detects abnormal network traffic and improves network security. Overall, the system provides reliable detection accuracy and supports real-time monitoring for cybersecurity applications.

VIII. CONCLUSION

In this project, anomaly detection in network traffic using advanced machine learning techniques was successfully implemented and evaluated. Multiple classification algorithms such as Random Forest, Logistic Regression, and KNN were used to identify abnormal network behavior. The experimental results showed that Random Forest achieved the highest accuracy, while other models provided moderate performance. The comparison highlighted that ensemble-based approaches are more effective in detecting complex and dynamic anomalies in network traffic.

The proposed system improved detection accuracy by applying preprocessing, normalization, and comparative classification techniques. The results demonstrated that advanced ML models can reduce misclassification and improve reliability in identifying suspicious activities.

The system is capable of handling large-scale network data and supports real-time anomaly detection. This makes it suitable for cybersecurity applications such as intrusion detection, malware detection, and unauthorized access monitoring.

Overall, the proposed approach provides an efficient and reliable solution for detecting anomalies in network traffic. By using advanced machine learning techniques and comparative performance analysis, the system enhances network security and reduces potential threats. Future improvements can include deep learning models and real-time deployment for further increasing detection accuracy and system performance.

REFERENCES

[1]. K. Kostas, Anomaly Detection in Networks Using Machine Learning. Research Proposal, March 23, 2018.
[2]. Miniwatts Marketing Group, "Internet Growth Statistics," 2018. Available online: https://www.internetworldstats. com/emarketing.htm [Accessed August 26, 2018].
[3]. K. Leung and C. Leckie, "Cluster-based unsupervised anomaly detection for network intrusion detection," presented at the Australasian Conference on Computer Science, 2005, pp. 333-342. Published by the Australian Computer Society, Inc.
[4]. Sharafaldin, A. Gharib, A. H. Lashkari, and A. A. Ghorbani, "Development of a reliable benchmark dataset for intrusion detection," Software Networking, vol. 1, no. 1, pp. 177- 200, 2017.
[5]. Massachusetts Institute of Technology, Lincoln Laboratory, "DARPA Intrusion Detection Evaluation Dataset 1998." Available at: https://www.ll.mit.edu/rd/datasets/1998- darpa-intrusion-detection-evaluation-data-set. [Accessed August 5, 2018].

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

[6]. C. Thomas, V. Sharma, and N. Balakrishnan, "Evaluating the utility of the DARPA dataset for testing intrusion detection systems," presented at the conference on Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security, vol. 6973, 2008.
[7]. Gharib, I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Framework for evaluating intrusion detection datasets," presented at the International Conference on Information Science and Security (ICISS), 2016, pp. 1-6. Published by IEEE.
[8]. University of California, Irvine, "KDD Cup 1999 Dataset." Available online: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. [Accessed August 5, 2018].
[9]. Özgür and H. Erdem, "Reviewing the usage of the KDD99 dataset for intrusion detection research between 2010 and 2015," PeerJ Preprints, vol. 4, article e1954v1, 2016.
[10]. Center for Applied Internet Data Analysis (CAIDA), "OC48 Peering Point Traces Dataset." Available online: https://www.caida.org/data/passive/passive_oc48_dataset. xml. [Accessed August 6, 2018].
[11]. M. Ahmed, A. N. Mahmood, and J. Hu, "Survey of techniques for detecting network anomalies," Journal of Network and Computer Applications, vol. 60, pp. 19-31, 2016.
[12]. University of New Brunswick, Canadian Institute for Cybersecurity, "NSL-KDD Dataset." Available online: http://www.unb.ca/cic/datasets/nsl.html. [Accessed August 6, 2018]. R. Bhallamudi et al., "Deep Learning Model for Resolution Enhancement of Biomedical Images for Biometrics," in Generative Artificial Intelligence for Biomedical and Smart Health Informatics, Wiley Online Library, pp. 321–341, 2025.
[13]. R. Bhallamudi et al., "Artificial Intelligence Probabilities Scheme for Disease Prevention Data Set Construction in Intelligent Smart Healthcare Scenario," SLAS Technology, vol. 29, pp. 2472–6303, 2024, Elsevier.
[14]. R. Bhallamudi, "Improved Selection Method for Evolutionary Artificial Neural Network Design," Pakistan Heart Journal, vol. 56, pp. 985–992, 2023.
[15]. R. Bhallamudi et al., "Time and Statistical Complexity of Proposed Evolutionary Algorithm in Artificial Neural Networks," Pakistan Heart Journal, vol. 56, pp. 1014–1019, 2023.
[16]. R. Krishna et al., "Smart Governance in Public Agencies Using Big Data," The International Journal of Analytical and Experimental Modal.
[17]. N. M. Krishna, "Object Detection and Tracking Using YOLO," in 3rd International Conference on Inventive Research in Computing Applications (ICIRCA-2021), IEEE, Sept. 2021, ISBN: 978-0-7381-4627-0.
[18]. N. M. Krishna, "Deep Learning Convolutional Neural Network (CNN) with Gaussian Mixture Model for Predicting Pancreatic Cancer," Springer US, vol. 1380- 7501, pp. 1–15, Feb. 2019.
[19]. N. M. Krishna, "Emotion Recognition Using Skew Gaussian Mixture Model for Brain–Computer Interaction," in SCDA-2018, Textbook Chapter, ISBN: 978-981-13- 0514, pp. 297–305, Springer, 2018.
[20]. N. M. Krishna, "A Novel Approach for Effective Emotion Recognition Using Double Truncated Gaussian Mixture Model and EEG," I.J. Iltelligent Systems and Applications, vol. 6, pp. 33–42, 2017