International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 11, Issue 4s, April-2026, http://ijmec.com/, ISSN: 2456-4265

Full Length Article

Intelligent Network Traffic Anomaly Detection Using Machine Learning Algorithms

Mohd Ameen Uddin Malik[1], Syed Raza Hussaini[2], Mohammed Noman[3], Mr. Mohammed Rahmat Ali[4]
[1,2,3]B.E.Students; Department of Computer Science & Engineering ISL Engineering College, Hyderabad, India
[4]Assistant Professor, Department of Computer Science & Engineering ISL Engineering College, Hyderabad, India
Mail Id; ameenuddinsaif123@gmail.com , sdraza58@gmail.com , mohammednoman9785@gmail.com , mdrahmatali@islec.edu.in

Accepted 24-04-2026

ABSTRACT

With the rapid growth of internet communication and cloud-based services, network security has become a critical challenge due to the increasing number of cyber threats and malicious attacks. Traditional intrusion detection systems rely mainly on predefined signatures and rules, making them less effective against modern and evolving attacks. To address these limitations, this paper presents an intelligent network traffic anomaly detection system using machine learning algorithms. The proposed system utilizes the KDD Cup 1999 dataset for training and evaluation. Data preprocessing techniques such as feature encoding and feature selection are applied to improve prediction performance. Two machine learning algorithms, Random Forest and CATBoost Classifier, are implemented for anomaly detection. The trained models are integrated into a Flask-based web application that supports both CSV file upload analysis and real-time manual prediction. Experimental results demonstrate that the proposed system achieves high accuracy exceeding 99% while maintaining reliable prediction performance. The system also provides visualization features for improved analysis and interpretation of network traffic behaviour. Overall, the proposed approach offers a scalable, efficient, and user-friendly solution for intelligent intrusion detection and network security enhancement.

Keywords: *Machine Learning, Network Security, Intrusion Detection System, CATBoost, Random Forest, Anomaly Detection, Flask Web Application, KDD Cup 1999 Dataset.*

INTRODUCTION

The rapid expansion of internet services, cloud computing, and digital communication has significantly increased the complexity of network infrastructures. As organizations increasingly depend on network-based systems, cybersecurity threats such as denial-of-service attacks, unauthorized access, probing, and malware attacks have become more frequent and sophisticated. Protecting sensitive information and ensuring secure communication has therefore become an essential requirement for modern organizations.

Traditional intrusion detection systems mainly depend on rule-based and signature-based detection methods. Although these systems are effective against known attacks, they often fail to detect zero-day attacks and evolving malicious activities. Furthermore, maintaining updated attack signatures becomes increasingly difficult as cyber threats continue to evolve.

Machine learning techniques provide a more intelligent and adaptive approach to anomaly detection by learning patterns from historical network traffic data. These models can identify unusual behaviour and classify network traffic as either normal or malicious with high accuracy. In recent years, machine learning-based intrusion detection systems have gained significant importance due to their ability to handle large datasets and improve threat detection capabilities.

In this paper, an intelligent network traffic anomaly detection system is proposed using Random Forest and CATBoost Classifier algorithms. The system is trained using the KDD Cup 1999 dataset and deployed as a Flask-based web application. The application allows users to perform batch prediction using CSV uploads and real-time prediction through manual input. Visualization modules are also included to provide better understanding of prediction results and model performance.

LITERATURE REVIEW

Several research studies have explored machine learning and deep learning approaches for network anomaly detection.

A. A. Jihado and A. S. Girsang proposed a hybrid intrusion detection model using Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) for capturing spatial and temporal network traffic features. Their

https://doi.org/10.63665/IJMEC.1104s.15
ISSN: 2456-4265
IJMEC 2026

approach achieved high detection accuracy but required significant computational resources.

L. I. Khalaf et al. presented a deep learning-based anomaly detection system for cyber threat identification. Their work highlighted the effectiveness of deep learning techniques in detecting large-scale network anomalies. However, the model required large training datasets and increased processing time.

S. Gunupusala and S. C. Kaila focused on multi-class anomaly detection using machine learning classifiers. Their research demonstrated that feature selection and balanced datasets significantly influence prediction performance.

K. Lu explored statistical and machine learning approaches for network traffic anomaly analysis. The study emphasized the need for intelligent systems capable of adapting to evolving cyber threats.

A. Alfardus and D. B. Rawat investigated machine learning-based anomaly detection for securing in-vehicle communication networks. Their work demonstrated the flexibility and applicability of machine learning approaches across multiple cybersecurity domains.

Although many existing systems provide strong performance, several limitations remain, including high computational complexity, preprocessing overhead, and limited adaptability to real-time systems. The proposed system addresses these challenges by combining Random Forest and CATBoost models with an efficient Flask-based deployment framework.

METHODOLOGY

The proposed system follows a machine learning-based methodology for detecting anomalies in network traffic. The complete workflow consists of data preprocessing, feature selection, model training, prediction generation, and visualization.

Dataset Collection

The KDD Cup 1999 dataset is used for training and evaluation. The dataset contains labelled network traffic records representing both normal and malicious network activities. Features include protocol type, service, flag, byte count, error rate, and connection statistics.

Data Preprocessing

Categorical attributes such as protocol type, service, and flag are encoded using label encoding techniques. Irrelevant features are removed, and the dataset is cleaned to improve model performance.

Feature Selection

Feature importance analysis is performed to identify the most significant attributes contributing to anomaly detection. Top important features are selected to reduce computational complexity and improve accuracy.

Model Training

Two machine learning algorithms are implemented:

Random Forest

CATBoost Classifier

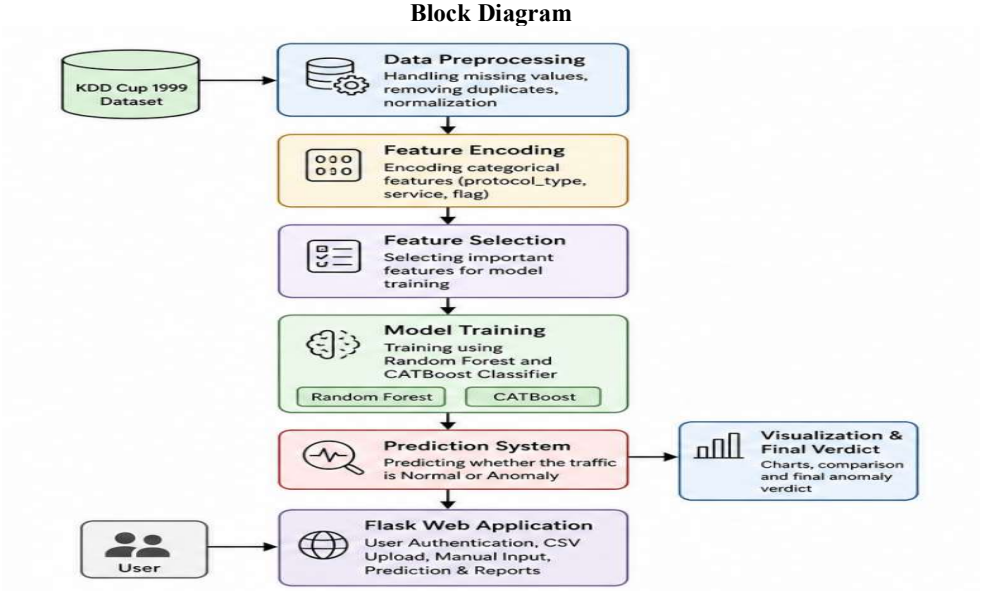
The dataset is divided into training and testing sets using an 80:20 ratio. Both models are trained and evaluated using performance metrics such as accuracy, precision, recall, and F1-score.

Web Application Deployment

The trained models are integrated into a Flask-based web application. The application supports:

- CSV-based batch prediction
- Manual input prediction
- Visualization dashboards
- Prediction result analysis

Block Diagram



https://doi.org/10.63665/IJMEC.1104s.15
ISSN: 2456-4265
IJMEC 2026

IMPLEMENTATION

The implementation phase involves training machine learning models and integrating them into the web application.

Random Forest Algorithm

Random Forest is an ensemble learning algorithm that constructs multiple decision trees and combines their predictions to improve accuracy and reduce overfitting.

Advantages of Random Forest

- High classification accuracy
- Reduced overfitting
- Better handling of large datasets
- Robust performance for classification tasks

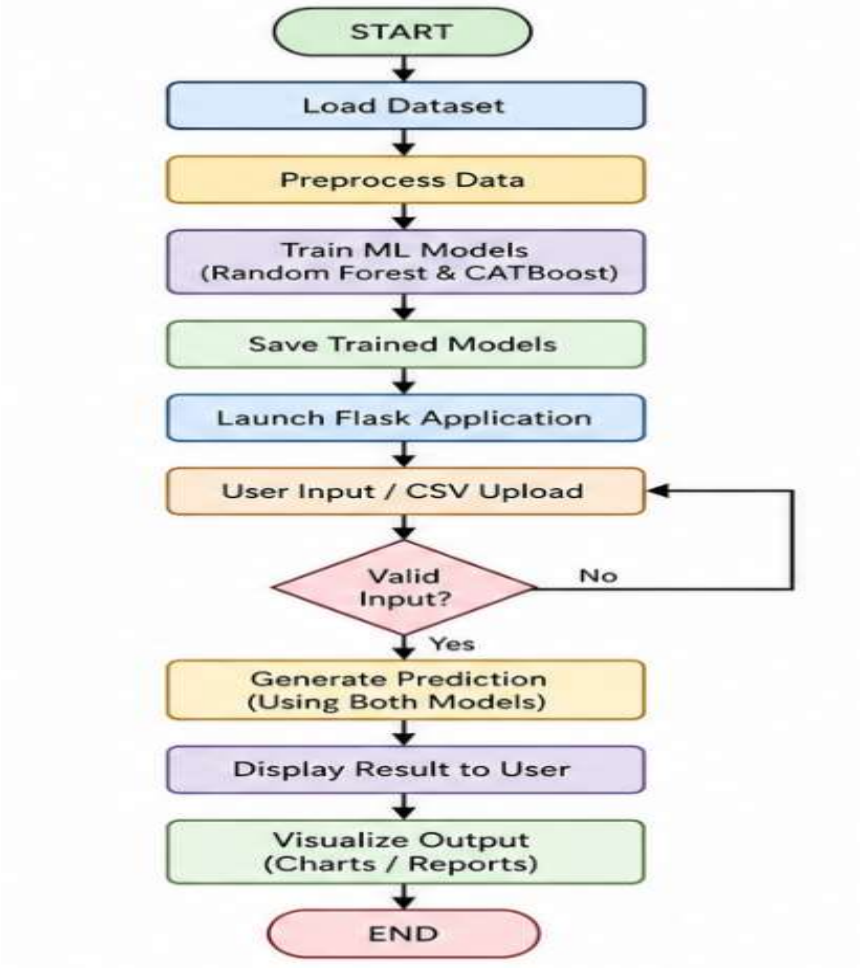
CATBoost Classifier

CATBoost is an advanced gradient boosting algorithm designed for handling categorical and numerical data efficiently. It uses ordered boosting techniques to reduce prediction bias and improve generalization.

Advantages of CATBoost

- Efficient handling of categorical features
- Reduced preprocessing requirements
- Improved prediction accuracy
- Better generalization performance

Flowchart



Flask Web Application

The web application is developed using Flask framework. The system contains the following modules:

- Login Authentication Module
- Dashboard Module
- CSV Upload Prediction Module
- Manual Prediction Module
- Visualization Module

The application allows users to upload CSV files or manually enter network traffic parameters to obtain anomaly detection results.

TESTING

https://doi.org/10.63665/IJMEC.1104s.15
ISSN: 2456-4265
IJMEC 2026

Software testing is performed to verify the functionality and performance of the anomaly detection system.

Unit Testing
Each module such as login authentication, prediction module, CSV upload handling, and visualization is tested individually.

Functional Testing
Functional testing ensures that all features work according to system requirements. The following functionalities are validated:

- Login validation
- CSV upload prediction
- Manual anomaly prediction
- Result display
- Visualization output

Integration Testing
Integration testing verifies communication between Flask application modules, machine learning models, preprocessing functions, and visualization components.

Performance Testing
Performance testing evaluates system response time and prediction efficiency for both manual and batch prediction.

RESULTS AND DISCUSSION

The proposed anomaly detection system achieved excellent performance during experimental evaluation.

Model Accuracy

Algorithm	Accuracy
Random Forest	**99.6%**
CATBoost Classifier	**99.42%**

The results indicate that both models perform effectively in classifying network traffic anomalies.

Prediction Results
The system successfully classifies network traffic as either Normal or Potential Attack using both Random Forest

Visualization Results
The Flask web application provides graphical visualizations including:

- Accuracy comparison charts
- Attack distribution graphs
- Confusion matrix visualization
- Feature importance analysis

Discussion
Experimental analysis demonstrates that the proposed system provides high accuracy and reliable performance while maintaining low prediction latency. CATBoost showed excellent handling of categorical features, whereas Random Forest provided stable baseline classification performance.

CONCLUSION

This paper presented an intelligent network traffic anomaly detection system using machine learning algorithms. The system utilizes the KDD Cup 1999 dataset along with preprocessing and feature selection techniques to improve anomaly detection performance.

Random Forest and CATBoost Classifier algorithms were implemented and evaluated for network intrusion detection. Experimental results demonstrated that the proposed system achieved high accuracy exceeding 99% while providing reliable classification performance.

A Flask-based web application was also developed to provide a practical and user-friendly interface for network traffic analysis through CSV uploads and real-time prediction. The visualization features further improve interpretability and system usability. Overall, the proposed system demonstrates the effectiveness of machine learning techniques for intelligent intrusion detection and provides a scalable solution for improving cybersecurity.

FUTURE SCOPE

The proposed system can be further enhanced in several ways:

- Implementation of multi-class attack classification
- Real-time live packet monitoring
- Integration of deep learning models such as CNN and LSTM
- Cloud deployment for enterprise-level scalability
- Integration with Intrusion Prevention Systems (IPS)
- Development of mobile-friendly dashboards
- Advanced visualization and analytics modules
- Improved authentication and role-based access control

These enhancements can improve the overall scalability, efficiency, and real-time detection capabilities of the system.

REFERENCES

[1] U. Fiore, F. Palmieri, A. Castiglione, and A. De Santis, “Network anomaly detection with the restricted Boltzmann machine,” Neurocomputing, vol. 122, pp. 13–23, 2013.
[2] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection,” ACM Computing Surveys, vol. 41, no. 3, pp. 1–58, 2009.

https://doi.org/10.63665/IJMEC.1104s.15
ISSN: 2456-4265
IJMEC 2026

[3] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," Journal of Network and Computer Applications, vol. 60, pp. 19–31, 2016.
[4] K. Lu, "Network anomaly traffic analysis," Academic Journal of Science and Technology, vol. 10, no. 3, pp. 65–68, 2024.
[5] L. I. Khalaf et al., "Deep learning-based anomaly detection in network traffic for cyber threat identification," 2024.
[6] S. Gunupusala and S. C. Kaila, "Multi-class network anomaly detection using machine learning techniques," Contemporary Mathematics, vol. 5, no. 2, pp. 5–22, 2024.
[7] A. Alfardus and D. B. Rawat, "Machine learning-based anomaly detection for securing in-vehicle networks," Electronics, vol. 13, no. 10, p. 1962, 2024.
[8] M. Tavallaee et al., "A detailed analysis of the KDD CUP 99 data set," IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009.
[9] G. Fernandes et al., "A comprehensive survey on network anomaly detection," Telecommunications Systems, vol. 70, no. 3, pp. 447–489, 2019.
[10] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," Scientific Reports, vol. 14, no. 1, p. 6086, 2024.

https://doi.org/10.63665/IJMEC.1104s.15
ISSN: 2456-4265
IJMEC 2026