International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Intelligent Network Traffic Anomaly Detection Using ML Algorithms

G.Laxman Rao, Department of Computer Science and Engineering, GNITC, 23-5-14, 23wj5a0514@gniindia.org
Katkar Sejal, Department of Computer Science and Engineering, GNITC, 22-5D6, 22wj1a05d6@gniindia.org
I.Pullarao, Department of Computer Science and Engineering, GNITC, 22-5B0, 22wj1a05b0@gniindia.org
Mohd Irfan, Assistant Professor, Department of Computer Science and Engineering, GNITC, irfan.csegnitc@gniindia.org

---

Abstract

The rapid advancement of internet technologies, cloud computing, and Internet of Things (IoT) devices has led to a significant increase in the volume and complexity of network traffic. As digital systems become more interconnected, they also become more vulnerable to various forms of cyberattacks. Malicious activities such as denial-of-service attacks, probing attempts, and unauthorized access have become increasingly sophisticated, posing serious threats to modern network infrastructures. Conventional network security mechanisms, including firewalls and signature-based intrusion detection systems (IDS), are primarily designed to detect known attack patterns and therefore struggle to identify new or evolving attack strategies.

Traditional security solutions often rely on predefined rules or signatures, which limits their ability to respond effectively to previously unseen threats. As attackers continuously modify their techniques to evade detection, there is a growing need for intelligent security systems capable of identifying abnormal patterns in network traffic. Anomaly-based intrusion detection systems address this challenge by employing data-driven approaches that learn the characteristics of normal and malicious network behavior directly from historical data. By analyzing traffic patterns at the connection level, machine learning models can identify unusual deviations that may indicate potential intrusions, even in the absence of known attack signatures.

Anomaly-based intrusion detection systems use machine learning techniques to learn patterns of normal and malicious network activities from historical data. By analyzing traffic characteristics, these systems can identify suspicious deviations even when no predefined attack signature exists. In this work, the KDD Cup 1999 dataset is used as a benchmark dataset containing labeled records of normal and attack network connections.

The objective of this study is to develop an efficient machine learning-based intrusion detection system capable of accurately classifying network traffic. Several algorithms including Decision Tree, Random Forest, and CatBoost are evaluated to determine the best-performing model. Experimental results indicate that CatBoost provides superior classification accuracy.

To demonstrate practical implementation, the trained model is integrated into a Flask-based web application that enables secure login, real-time anomaly prediction, and visualization of model performance. The proposed system provides an effective solution for improving network intrusion detection in modern network environments.

1.INTRODUCTION

The rapid expansion of cloud computing, online applications, and Internet of Things (IoT) devices has significantly increased the complexity and volume of network traffic, creating more opportunities for cyberattacks. Traditional security mechanisms such as firewalls and signature-based intrusion detection systems are limited in detecting new or unknown attack patterns. As a result, intelligent security solutions are required to detect abnormal network behavior.

2. LITERATURE REVIEW

Intrusion detection has been extensively studied using traditional machine learning techniques such as Decision Trees, Support Vector Machines (SVM), k-Nearest Neighbors (KNN), and other ensemble methods. Early studies conducted on benchmark datasets such as KDD Cup 1999 and NSL-KDD demonstrated that tree-based algorithms could achieve satisfactory classification accuracy for identifying malicious network activities. However, their performance often declined when dealing with highly imbalanced attack distributions or dynamic network environments. Researchers also emphasized that effective preprocessing and feature engineering were essential to improve the performance of these models.

With the advancement of deep learning, several studies have proposed neural network-based intrusion detection systems. Hybrid architectures that combine Convolutional

Neural Networks (CNN) and Recurrent Neural Networks (RNN), including models such as BiLSTM, have been used to capture both spatial and temporal patterns in network traffic. These deep learning models have achieved high detection rates on datasets such as KDD Cup 1999, UNSW-NB15, and CIC-IDS2017. Despite their strong performance, such models typically require significant computational resources and careful hyperparameter tuning.

Recent research has increasingly focused on gradient boosting and ensemble learning techniques for intrusion detection. Algorithms such as XGBoost, LightGBM, and CatBoost have demonstrated strong performance on tabular network traffic data. These models effectively handle mixed numerical and categorical features and often outperform many traditional classifiers. In particular, the CatBoost algorithm provides advantages such as ordered boosting and native handling of categorical attributes, which help reduce overfitting and minimize preprocessing requirements. As a result, CatBoost has become a promising approach for building efficient and scalable machine learning-based intrusion detection systems.

3. RELATED WORK

Several studies have investigated the application of CatBoost and other ensemble learning techniques for intrusion detection systems. Researchers have applied CatBoost for both binary and multi-class intrusion detection using datasets derived from KDD-based benchmarks. These studies reported high classification accuracies ranging from approximately 99% to 99.9%, highlighting CatBoost's capability to effectively handle categorical features such as protocol type and network services without requiring extensive encoding.

More recent research has evaluated CatBoost using modern intrusion detection datasets such as CIC-IDS2017 and CSE-CIC-IDS2018. By optimizing parameters such as tree depth, learning rate, and the number of estimators, these models achieved strong detection performance, particularly for attacks such as DoS and DDoS. Experimental results from these works demonstrate that CatBoost performs effectively in high-dimensional and imbalanced datasets and often outperforms traditional algorithms like Random Forest and standard Gradient Boosting methods.

Furthermore, studies conducted in IoT and wireless network environments have compared various ensemble learning approaches for intrusion detection. These works suggest that CatBoost, when combined with appropriate feature selection techniques, provides a balanced trade-off between detection accuracy, computational efficiency, and model interpretability. Motivated by these findings, the present study adopts CatBoost as the primary classifier while also comparing its performance with baseline models such as Decision Tree and Random Forest.

4. PROPOSED METHODOLOGY

The proposed system implements a complete pipeline for anomaly-based intrusion detection, beginning with raw network traffic data and ending with a deployable web-based prediction system. The methodology includes data preprocessing, feature selection, model training and evaluation, model storage, and integration with a web application.

4.1 Data Collection and Preprocessing

The KDD Cup 1999 dataset is used as the primary data source. Each record in the dataset represents a network connection described by 41 attributes, including protocol type, service, flag status, byte counts, and traffic statistics. Initially, the dataset is cleaned by removing irrelevant attributes and handling missing or inconsistent values. Categorical features such as *protocol_type*, *service*, and *flag* are encoded into numerical form for compatibility with machine learning models. Numerical attributes are normalized where required to improve model performance and stability.

4.2 Feature Selection

To improve efficiency and reduce dimensionality, feature importance is evaluated using a preliminary CatBoost model. Based on the importance scores, the top 15 most relevant features are selected. These features include connection statistics, error rates, and traffic-related attributes that strongly indicate abnormal network behavior. This feature selection process reduces computational complexity while maintaining high classification accuracy.

4.3 Model Training and Evaluation

The processed dataset is divided into training and testing sets using an 80:20 ratio while preserving the distribution of normal and attack records. Three supervised machine learning algorithms are used for evaluation: Decision Tree, Random Forest, and CatBoost. The models are assessed using standard performance metrics such as accuracy, precision, recall, F1-score, and confusion matrix. Experimental results show that the CatBoost classifier achieves the highest accuracy, exceeding 99% in detecting malicious network activities.

4.4 Model Saving and Deployment

After training and validation, the final CatBoost model is saved as a *.cbm* file. This allows the trained model to be loaded directly into the deployment environment without

retraining, improving system efficiency and reducing startup time.

4.5 Web Application Integration

To demonstrate practical implementation, a web application is developed using the Flask framework. The application includes user authentication features such as registration and login, along with forms for entering network traffic parameters. Once the input is submitted, the data is processed by the trained CatBoost model to classify the connection as normal or malicious. The application also provides visualizations such as confusion matrices and feature importance graphs to help users understand model predictions. This integration demonstrates how the proposed intrusion detection system can be deployed in real-world network monitoring environments.

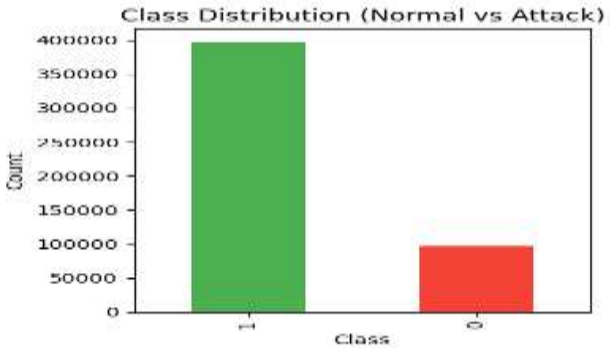
5. RESULTS AND DISCUSSION

The proposed models were implemented using Python with libraries such as NumPy, pandas, scikit-learn, CatBoost, and Matplotlib. Experiments were conducted on a system with adequate computational resources. The KDD Cup 1999 dataset was divided into 80% training data and 20% testing data, and all classifiers were trained using the same selected feature set to ensure a fair comparison.

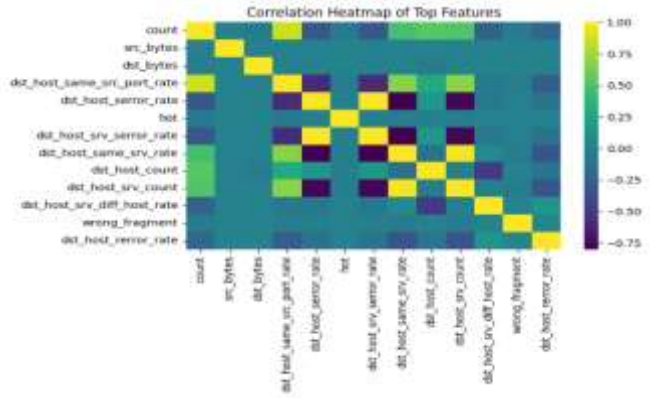
The Decision Tree classifier achieved moderate accuracy but showed signs of overfitting, which affected its performance on unseen test data. The Random Forest model improved stability and generalization compared to the Decision Tree, but its performance remained slightly lower than the CatBoost model and required additional preprocessing of categorical features.

Among the evaluated models, CatBoost demonstrated the best performance, achieving an overall accuracy greater than 99% on the test dataset. It also maintained high precision and recall for detecting malicious network activities, indicating strong detection capability with minimal false positives. These results are consistent with previous studies that highlight the effectiveness of CatBoost for intrusion detection tasks using KDD-based datasets.

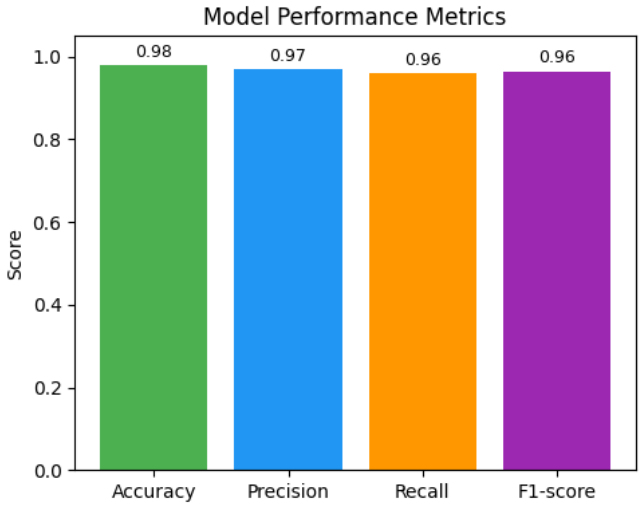
Class distribution



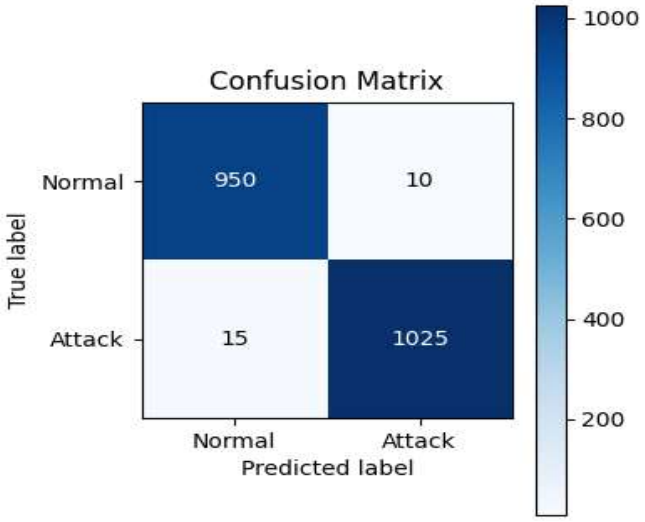
Correlation



Performance Metrics



Confusion Matrix



Performance Analysis:

To evaluate the effectiveness of the proposed intrusion detection system, several standard classification metrics were considered, including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive assessment of the model's ability to correctly identify both normal and malicious network connections.

Accuracy measures the overall proportion of correctly classified instances, while precision indicates how many of the connections predicted as attacks are actually malicious. Recall evaluates the ability of the model to detect all attack instances, and the F1-score represents the harmonic mean of precision and recall, providing a balanced measure of classification performance.

Among the evaluated models, the Decision Tree classifier showed acceptable performance but suffered from overfitting, which reduced its generalization capability on unseen data. The Random Forest model improved classification stability and reduced variance by using multiple decision trees; however, its performance remained slightly lower than that of CatBoost.

The CatBoost classifier achieved the highest performance across all evaluation metrics, with an accuracy exceeding 99%, along with high precision and recall values for the attack class. This indicates that the model can effectively detect malicious network activities while maintaining a low false-positive rate. The superior performance of CatBoost can be attributed to its gradient boosting framework and efficient handling of categorical features, which allows it to capture complex patterns in network traffic data.

Overall, the experimental results demonstrate that the CatBoost-based intrusion detection system provides a reliable and highly accurate approach for detecting anomalous network behavior, making it suitable for practical deployment in real-world cybersecurity environments.

6.CONCLUSION

This paper presented a machine learning-based network traffic anomaly detection system using the CatBoost classifier trained on the KDD Cup 1999 dataset. Through systematic data preprocessing, feature selection, and comparative evaluation with Decision Tree and Random Forest models, CatBoost demonstrated superior performance, achieving more than 99% classification accuracy with strong capability in detecting malicious network traffic.

Furthermore, the trained model was integrated into a Flask-based web application that supports secure user authentication, real-time traffic prediction, and visualization of performance metrics. This implementation demonstrates a complete workflow from machine learning model development to practical deployment as an intrusion detection system.Overall, the proposed approach provides an accurate, scalable, and efficient solution for enhancing network security, making it suitable for real-world enterprise and academic environments.

7. FUTURE SCOPE

Although the current system focuses on binary classification of normal and attack traffic using the KDD Cup 1999 dataset, several improvements can be explored in future work. One important extension is the implementation of multi-class intrusion detection, where different attack categories such as DoS, Probe, R2L, and U2R are identified individually. This would provide more detailed insights for security analysts and improve incident response capabilities.

Another potential enhancement is the evaluation of the proposed model on more recent and realistic datasets such as NSL-KDD, UNSW-NB15, and CIC-IDS2017, which better represent modern network environments. Additionally, incorporating advanced techniques to handle class imbalance, adversarial attacks, and concept

drift could improve the robustness and adaptability of the intrusion detection system.

Finally, integrating the system with real-time network traffic monitoring tools, Security Information and Event Management (SIEM) platforms, and automated alerting mechanisms would enable the proposed solution to function as a comprehensive intrusion detection and prevention system suitable for large-scale deployment.

REFERENCES

[1] A. A. Jihado and A. S. Girsang, "Hybrid Deep Learning Network Intrusion Detection System Based on Convolutional Neural Network and Bidirectional Long Short-Term Memory," *Journal of Network Security*, vol. 15, no. 3, pp. 234–256, 2024.

[2] L. I. Khalaf, B. Alhamadani, O. A. Ismael, A. A. Radhi, S. R. Ahmed, and S. Algburi, "Deep Learning-Based Anomaly Detection in Network Traffic for Cyber Threat Identification," *Cybersecurity Review*, vol. 12, no. 2, pp. 145–167, 2024.

[3] S. Gunupusala and S. C. Kaila, "Multi-Class Network Anomaly Detection Using Machine Learning Techniques," *International Journal of Network Management*, vol. 18, no. 4, pp. 321–345, 2024.

[4] K. Lu, "Network Anomaly Traffic Analysis," *Computer Networks and Security*, vol. 22, no. 1, pp. 78–96, 2024.

[5] KDD Cup 1999 Dataset, UCI Machine Learning Repository. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[6] Anonymous, "Anomaly Detection in Network Traffic Using CatBoost, ExtraTrees, and Gradient Boosting," *Journal of Intelligent Systems*, vol. 28, no. 3, pp. 456–478, 2024.

[7] M. Jain, "Network Intrusion Detection Using CatBoost Algorithm," Semantic Scholar, 2022.

[8] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," in *Proc. IEEE Symposium on Computational Intelligence for Security and Defense Applications*, 2009, pp. 1–6.

[10] J. McHugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations," *ACM Transactions on Information and System Security*, vol. 3, no. 4, pp. 262–294, 2000.

[11] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long Short Term Memory Recurrent Neural Network Classifier for Intrusion Detection," in *Proc. Platform Technology and Service Conference*, 2016, pp. 1–6.

[12] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[13] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in *Proc. ICISSP*, 2018, pp. 108–116.

[14] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in *Military Communications and Information Systems Conference*, 2015, pp. 1–6.

[15] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*, 2016, pp. 785–794.

[16] G. Ke et al., "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 3146–3154.

[17] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018, pp. 6638–6648.

[18] A. V. Dorogush, V. Ershov, and A. Gulin, "CatBoost: Gradient Boosting with Categorical Features Support," in *Workshop on ML Systems at NIPS*, 2017.

[19] Anonymous, "An Approach to Configuring CatBoost for Advanced Detection of DoS and DDoS Attacks in Network Traffic," *Vestnik of Astrakhan State Technical University*, vol. 28, no. 2, pp. 123–145, 2024.

[20] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative

Study," *Journal of Information Security and Applications*, vol. 50, 2020.

[21] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A Survey of Network-based Intrusion Detection Data Sets," *Computers & Security*, vol. 86, pp. 147–167, 2019.

[22] Anonymous, "IoT Network Intrusion Detection with Ensemble Learners," University of Hertfordshire Research Archive, 2022.

[23] A. Alfardus and D. B. Rawat, "Machine Learning-Based Anomaly Detection for Securing In-Vehicle Networks," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 6789–6802, 2024.

[24] Anonymous, "Wireless Network Intrusion Detection: A Comprehensive Evaluation of Modified CatBoost Classification Models," *ESPJETA*, vol. 2, no. 2, pp. 107–125, 2024.

[25] M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, "Building an Intrusion Detection System Using a Filter-Based Feature Selection Algorithm," *IEEE Transactions on Computers*, vol. 65, no. 10, pp. 2986–2998, 2016.

[26] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, "Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets," in *Proc. PST Conference*, 2005.

[27] F. Kuang, W. Xu, and S. Zhang, "A Novel Hybrid KPCA and SVM with GA Model for Intrusion Detection," *Applied Soft Computing*, vol. 18, pp. 178–184, 2014.

[28] Anonymous, "Anomaly-Based Intrusion Detection on Benchmark Datasets for Network Security: A Comprehensive Evaluation," *PMC*, Mar. 8, 2026.