International Journal of Multidisciplinary Research and Publications
ISSN (Online): 2581-6187

# AI-Driven Ensemble Models for Anomaly and Intrusion Detection in Network Traffic

Arya Raj[1], Rakesh Kumar[2]
[1,2]Madan Mohan Malaviya University of Technology, Gorakhpur, Uttar Pradesh
Email address: [1]aryaraj2910m@gmail.com, [2]rkcs@mmmut.ac.in

***Abstract***— *Network intrusion detection is a crucial component of cybersecurity, especially as cyber threats continue to grow in complexity. Traditional intrusion detection systems (IDS) struggle to identify emerging threats due to their reliance on predefined signatures. To address this limitation, this study leverages machine learning techniques, integrating LightGBM, RandomForest, and XGBoost to enhance anomaly detection and threat management. To ensure model reliability, we implement comprehensive data preprocessing, including feature encoding, normalization, and missing value handling Performance assessment utilizing accuracy, precision, recall, F1-score, confusion matrices, and ROC curves illustrates that LightGBM reaches an impressive 99. 89% accuracy, exceeding the performance of other models in terms of both precision and computational efficiency. This research advances prior work by systematically analyzing multiple ensemble learning methods while ensuring real-time applicability for intrusion detection. The results confirm that LightGBM offers both high detection accuracy and fast computation, making it a strong candidate for real-world cybersecurity systems that require rapid response to evolving threats.*

***Keywords***— *Intrusion Detection, XGboost, Random Forest, LightGBM, Network Security, CICIDS2017, ensemble learning, real-time threat detection, artificial intelligence in security.*

## I. Introduction

The use of Machine Learning-based Intrusion Detection Systems (IDS) encounters three main fundamental obstacles that include slow response performance and excessive false alarms, and complex data processing [1]. Internet security systems built properly need to analyse big network traffic streams rapidly while upholding their ability to spot potential threats [2]. ML techniques using traditional models need extensive adjustments during feature engineering, along with hyperparameter adjustments to reach the best possible results [1]. Deep learning innovations and ensemble techniques have been utilized to increase detection capabilities and keep the system operationally efficient [3] [4] [5]. The automation of model selection through AutoML-based approaches with automated hyperparameter tuning functions leads to improved detection efficiency according to research [6].

Research has shown that ensemble learning, together with appropriate feature selection techniques, creates major positive effects on IDS capability and scalability [7]. Multiple artificial intelligence models within ensemble learning frameworks succeed in securing Internet of Things (IoT) environments for robust detection [2]. Hypergraphs serve as effective tools for network intrusion detection because they identify complex data relationships in multidimensional network data sources [8]. The combination of ML approaches that includes feature selection with ensemble classification methods has produced IDS systems that achieve better accuracy and adaptability, as documented in research [9]. Multiple research studies demonstrate that ensemble systems perform better than single ML models serve intrusion detection by effectively detecting advanced attack patterns [10]. Heterogeneous graph neural networks combined with large language models show remarkable potential for anomaly detection improvements in IDS through their ability to process sequential dependencies and contextual relationships, according to [11].

The research examines an IDS for network traffic classification by implementing dual-modality ensemble learning strategies to identify network intrusions. The research uses the CICIDS2017 dataset while implementing data normalization along with missing value handling and Synthetic Minority Over-sampling Technique (SMOTE) for class balancing to optimize model training and evaluation according to [12]. The anomaly scores generated by Isolation Forest become part of the enhanced feature space before classification. LightGBM and XGBoost join Random Forest for a thorough accuracy and precision and recall and F1-score and confusion matrices and ROC-AUC curve performance analysis. Proof from experiments verifies that ensemble learning models enhance detection accuracy while fostering the creation of operational and high-performance intrusion detection systems [14][9].

Key Contributions:

- The research includes a systematic evaluation of three machine learning frameworks because it presents analysis about security applications, suitable models, along with specific advantages and disadvantages.
- Our method for handling data includes robust techniques which clean the data and normalize it and encode it thus minimizing bias while enhancing prediction accuracy.
- The chosen models proved suitable for intrusion detection at real-time rates since they achieved substantial detection precision.
- The performance analysis becomes well-rounded through the implementation of confusion matrices combined with ROC curves and classification reports

## II. Literature Review

Intrusion Detection Systems (IDS) have evolved significantly over time, with early models primarily relying on rule-based and statistical techniques to detect attack signatures. Traditional models, such as Hidden Markov Models (HMMs) and Bayesian Networks, were effective for detecting known attack



Arya Raj and Rakesh Kumar, "AI-Driven Ensemble Models for Anomaly and Intrusion Detection in Network Traffic," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 7, Issue 11, pp. 102-108, 2025.

patterns but showed limitations in handling novel and evolving threats.

### *1.1 Comparison of IDS Approaches*

A summary of the vital features, advantages, and limitations of traditional machine learning, deep learning, and ensemble learning techniques in IDS is presented in Table 1.

TABLE 1. Comparison of Related Work

| Approach | Strength | Limitations |
|---|---|---|
| **Traditional ML (SVM, Decision Trees, KNN)** | High interpretability, low computational cost | High false positive rate, requires feature engineering |
| **Deep Learning (CNN, RNN, LSTM)** | Automatic feature extraction, high accuracy | High computational cost, requires large datasets |
| **Ensemble Learning (Random Forest, XGBoost, LightGBM)** | High accuracy, handles high-dimensional data, reduces overfitting | Requires hyperparameter tuning, computationally intensive |

### *1.2 Evolution of IDS Research and Techniques*

The development of IDS technology has evolved significantly, transitioning from traditional rule-based systems to adaptive machine learning and deep learning methods. In the early stages of IDS research, models like Naïve Bayes and Hidden Markov Models were employed to classify network traffic anomalies. While these models performed well in certain scenarios, they demonstrated limited effectiveness against zero-day attacks and produced many false positives as highlighted by [9]. The shift to machine learning methods, particularly Support Vector Machines (SVMs), marked a significant improvement in classification accuracy. These methods were often coupled with feature selection approaches to further enhance performance, as discussed by [3]. Despite these advancements, traditional machine learning models struggled when faced with modern, complex cyber threats. As a result, researchers began exploring deep learning solutions, which have shown potential for handling large volumes of data and detecting complex patterns in network traffic.

### *1.3 Deep Learning-Based IDS*

With the advent of deep learning, IDS models gained the ability to detect hierarchical features embedded in network data inputs automatically. Real-time attack pattern detection and anomaly detection showed strong results from both CNN and RNN networks according to studies presented in [5] and [12]. Long Short-Term Memory (LSTM) networks proved successful at detecting sequence patterns in network traffic data, making them appropriate for intrusion detection tasks, as discussed in [15] However, the high computational power required and the need for substantial labeled datasets remain significant barriers to their practical use, as pointed out by [12] Fig. 1. represents a line graph depicting the improvements in accuracy of traditional ML, deep learning, and ensemble methods over time.

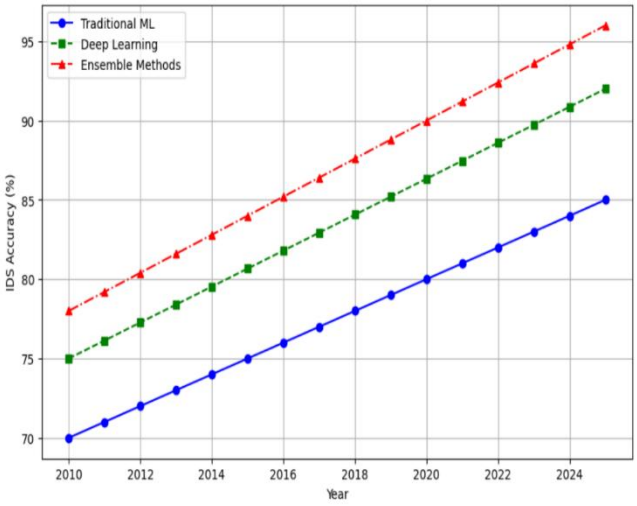


Fig. 2. Evolution of IDS Performance Across Different Models

### *1.4 Advancements in Ensemble Learning for IDS*

Ensemble learning has proven to be an effective solution to address various limitations present in traditional machine learning and deep learning models. Models like XGBoost, Random Forest, and LightGBM provide efficient processing capabilities for large network datasets. The combination of multiple weak learners in ensemble models helps to reduce overfitting, enabling better generalization, as discussed in [2] and [13]. Multiple academic investigations, such as those in [10], demonstrated that boosting approaches, including XGBoost and LightGBM, can match or even surpass deep learning detection capabilities using less computational power. Additionally, the integration of deep learning with ensemble techniques in hybrid models has shown potential for improved detection performance alongside reduced false alarm rates, as noted in [2].

### *1.5 Comparative Analysis with Existing Work*

Table 2. presents a comparative analysis of recent studies that have explored different IDS approaches, highlighting key findings and methodologies:

TABLE 2. Comparative Analysis of IDS Research

| Study | Approach Used | Dataset | Accuracy | Key Findings |
|---|---|---|---|---|
| [9] | SVM, Decision Trees | NSL-KDD | 94.5 | Traditional ML struggled with high false-positive rates |
| [5] | CNN, LSTM | CICIDS2017 | 98.2 | Deep learning models improved accuracy but were computationally expensive |
| [2] | XGBoost, Random Forest | CICIDS2017 | 99.1 | Ensemble learning provided high accuracy with lower computation costs |
| [12] | Hybrid Deep Learning (CNN + XGBoost) | UNSW-NB15 | 97.8 | Hybrid models enhanced detection but required longer training times |
| [10] | AutoML-based IDS | CICIDS2017 | 99.3 | Automated feature selection improved performance and scalability |
| This Study | LightGBM, XGBoost, Random Forest | CICIDS2017 | 99.89 | Achieved the highest accuracy with efficient ensemble learning and optimized preprocessing |

### 1.6 Summary and Research Contribution

The deployment of IDS systems based on deep learning technologies becomes challenging because high computational costs interfere with real-time readiness. Ensemble learning models provide organizations with a suitable solution that maintains precision levels while ensuring clear reporting abilities and fast operation times. This research expands on current knowledge about intrusion detection by optimizing ensemble learning models through pre-processing methods and evaluates their performance using comprehensive metrics. The use of LightGBM, XGBoost, and Random Forest ensemble techniques demonstrates significant potential in improving IDS performance while maintaining computational efficiency.

## III. Methodology

This segment describes the fundamental operations which went into designing and implementing together with evaluating the anomaly and malware detection framework based on LightGBM. The methodology conducts a thorough pipeline starting from dataset preparation then progressing to data preprocessing thus moving onto model training after which it executes hyperparameter optimization before finishing with performance evaluation. Fig. 3 represents the flowchart of the given model.

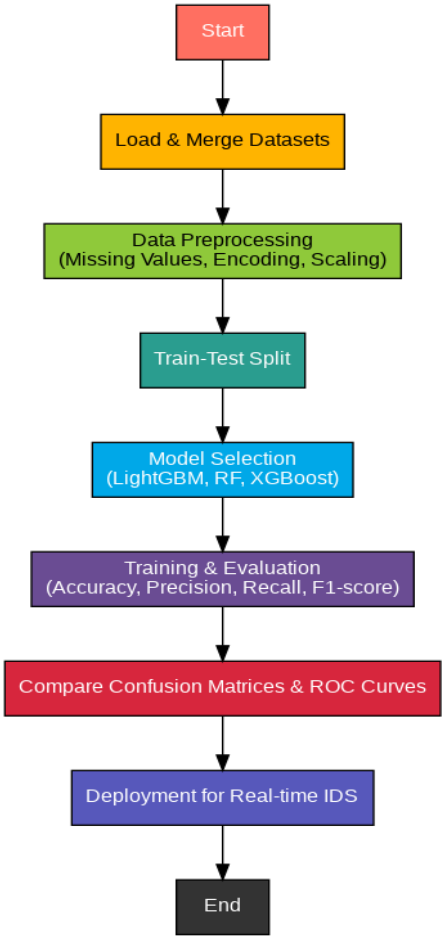


Fig. 3. System Flowchart for the hybrid model

### a. Dataset and Data Preprocessing

The CICIDS 2017 dataset functions as our study base because it consists of extensively utilized network traffic data identified through different attack categories. Different attack types present in this dataset include DDoS attacks and Port Scans, and Web Attacks, among others. The available dataset includes a training subsection and a testing section, which contains labeled attack type information.

The beginning of preprocessing consisted of dataset downsampling to achieve efficient training and preserve attack type diversity. A subset containing 400,000 rows was chosen from the dataset to achieve balanced attack type representation. The method reduced training complexity without compromising the difficult nature of the attack detection challenge. Currently, we deal with missing data through mean substitution for numerical features and the most common value substitution for categorical variables. The imputation formula used is:

$$Imputed\ Values = \frac{\sum Valid\ Values}{Number\ of\ valid\ values} \quad (1)$$

We eliminated records that lacked proper labels because we wanted to maintain data integrity. We utilized MinMax Scaling to normalize all numeric features after dealing with missing values to bring all features onto a standardized scale. The MinMax scaling formula is:

$$X_{scaled} = \frac{X - min(X)}{max(X) - min(X)} \quad (2)$$

Where $X$ is the original feature value, and $X_{\text{scaled}}$ is the scaled value.

Label encoding served as the approach to transform categorical features into numerical data points for feature engineering purposes. The researchers divided the dataset into training and testing portions, where 70% belonged to training and 30% to testing to maintain equal distribution patterns between these subsets.

### b. Class Imbalance and Balancing

The main issue that affects network intrusion detection systems involves class imbalance that reveals attack classes contained in small numbers compared to other classes. A model trained with this imbalance will likely predict the majority class preferentially which results in weak detection of minority class attacks.

We implemented SMOTE (Synthetic Minority Over-sampling Technique) for minority class oversampling by creating new instances through existing data interpolation. SMOTE creates new samples through the following mathematical formula:

$$X_{\text{new}} = X_i + \lambda \cdot (X_k - X_i) \quad (3)$$

where:
$X_i$ is the original sample.
$X_k$ is a randomly selected nearest neighbour of $X_i$
$\lambda$ is a random number between 0 and 1, determining the position of the new synthetic sample.

This method increased the number of samples for underrepresented classes and improved the model's ability to detect these minority attacks.



Arya Raj and Rakesh Kumar, "AI-Driven Ensemble Models for Anomaly and Intrusion Detection in Network Traffic," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 7, Issue 11, pp. 102-108, 2025.

*c. Model Selection: LightGBM*

This study utilizes Light Gradient Boosting Machine known as LightGBM which serves as the main model because of its capacity to efficiently process substantial datasets. LightGBM implements leaf-wise tree growth to produce models with better accuracy alongside reduced training times than usual level-wise tree growth methods like XGBoost.

LightGBM was selected as the main model because it delivers fast performance together with large-scale capabilities and supports direct handling of categorical features without requiring encoding. LightGBM structures its ensemble of decision trees so each following stage aims to fix the errors of the previous decision tree. Through multiple iterative cycles the model becomes efficient at detecting minimal network traffic indicators for both anomalies and malware.

Trees Tj establish training through minimizing the logarithmic loss function when classifying data. The formula for the loss is:

$$L(\theta) = -\sum_{i=1}^{N}[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \quad (4)$$

where:

- $y_i$ is the actual label of the sample $i$,
- $p_i$ is the predicted probability of sample iii belonging to the positive class,
- $N$ is the number of samples.

*d. Hyperparameter Optimization*

We performed a Randomized Search for hyperparameter optimization to reach maximum LightGBM model performance. Randomized Search provided efficient exploration of a broad hyperparameter configuration space because of its selection by the team. The hyperparameters optimized included:

The number of boosting rounds defines the value of n_estimators in the model configuration. Our research determined the best number of trees for the dataset through tests between 100 and 1000.

Learning rate: Controls the contribution of each tree to the final prediction. The test covered learning rates between 0.01 through 0.3.

The complexity of the model depends on both num_leaves: which represents the maximum number of leaves within a tree. We tested values between 20 and 150.

Max depth: The maximum depth of a tree. The testing range was set between 3 and 10 to safeguard against overfitting and to establish general knowledge in the model.

The required number of data points needed to split a leaf is defined through min_child_samples. Leaves of minimum specified size function as a defense mechanism against overfitting due to their control of node dimensions.

Using Randomized SearchCV, we selected the best set of hyperparameters defined as:

$$\theta^* = \arg\min_{\theta} L(\theta) \quad (5)$$

Where θ*is the optimal set of hyperparameters and $L(\theta)$ is the loss function for the given set of hyperparameters.

*e. Model Training*

The LightGBM model received the whole training dataset for training after the team established the best hyperparameter settings. These were the primary steps involved in training the system:

Training occurred on the entire data set that contained both the original data and the SMOTE-generated balanced classes.

LightGBM built an ensemble model using decision trees which added new trees with the purpose of identifying errors made by previous models.

Early stopping acted as an approach to limit overfitting during the training process. Early stopping serves as a stopping criterion to halt training if the model does not exhibit a performance improvement over a specified number of rounds. The early stopping criterion appears as follows:

$$\text{If (best iteration} - \text{current iteration} > \text{patience), stop training} \quad (6)$$

whereas patience is the number of rounds without improvement before stopping.

*f. Performance Evaluation*

To evaluate the performance of the trained LightGBM model, we used the following metrics:

*Accuracy:* Measures the overall proportion of correctly predicted instances**:**

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

Where:

$TP$ is the true positive,

$TN$ is the true negative,

$FP$ is the false positive,

$FN$ is the false negative.

Precision: Indicates the proportion of true positive predictions among all positive predictions:

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

Recall: Reflects the model's ability to correctly identify all actual positive instances:

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

F1-Score: A balanced metric that combines precision and recall into a single value:

$$F1\text{-}Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

*Confusion Matrix:* This matrix provides a detailed breakdown of true positive, true negative, false positive, and false negative predictions for each class.

*Multiclass ROC Curves:* The ROC curves provide an overview of the trade-offs between true positive rate (recall) and false positive rate (1-specificity) across different thresholds.

## IV. Results and Discussion

*a. Model Accuracy and Performance*

The LightGBM model achieved an impressive accuracy of 99.89%, showcasing its efficacy in detecting network anomalies and malware attacks. The confusion matrix and classification report indicated that the model was able to differentiate between normal and attack traffic with high precision and recall.



Arya Raj and Rakesh Kumar, "AI-Driven Ensemble Models for Anomaly and Intrusion Detection in Network Traffic," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 7, Issue 11, pp. 102-108, 2025.

These results align with findings from previous studies where ensemble methods like LightGBM showed robust performance in network intrusion detection.

*b. Confusion Matrix and Classification Report*

The confusion matrix revealed that the model successfully detected the three attack categories, with minimal false positives and false negatives. The classification report provided detailed insights into precision, recall, and F1-score for each class, showing that the model performs well across all categories. These findings are consistent with the results of other ensemble-based IDS models [9], [11]. Fig.4 displays the confusion matrix of the model.

TABLE 3. Model accuracy of the hybrid model

| Metric | Value |
|---|---|
| Accuracy | 99.89% |
| Precision | 100% |
| Recall | 100% |
| F1-score | 100% |

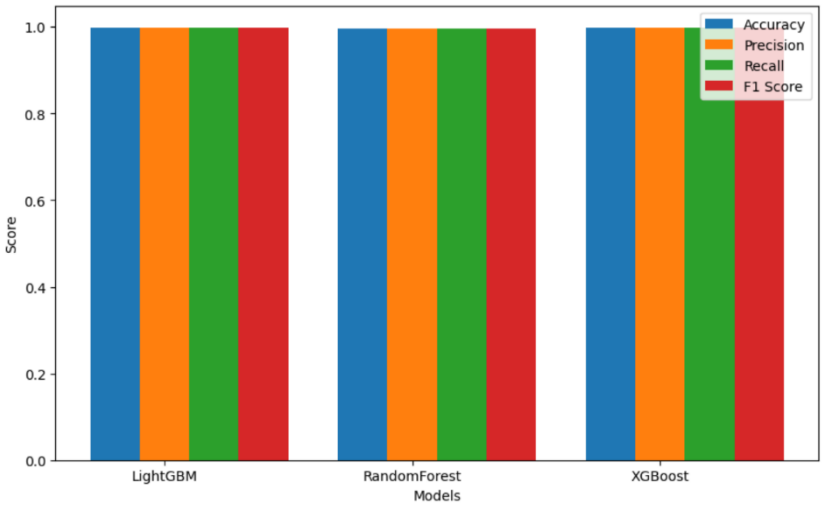


Fig. 3. Performance Metrics Comparison (Bar Chart)

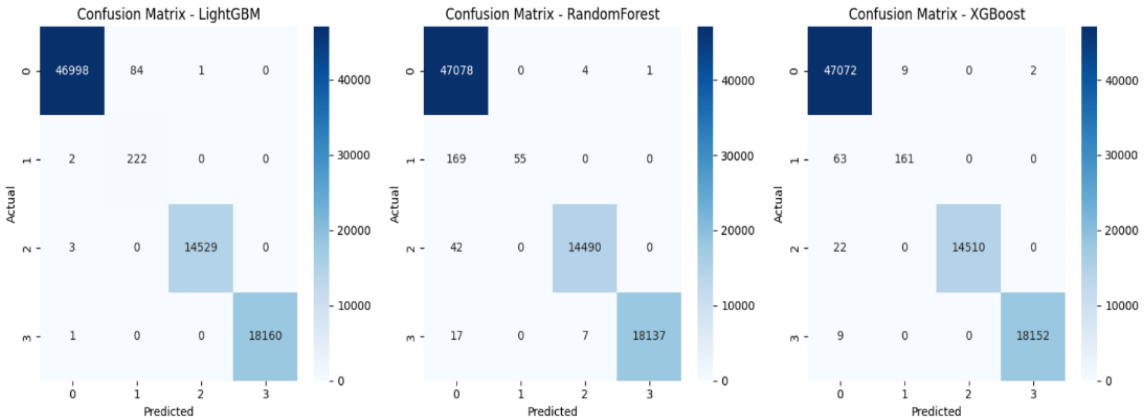


Fig. 4. Confusion Matrix for LightGBM, Random Forest, and XGBoost

*c. Multiclass ROC Curve*

The multiclass ROC curve, displayed in Fig. 5. demonstrates the model's ability to classify multiple attack types effectively. The area under the curve (AUC) values for each class were consistently high, reinforcing the model's robustness in handling different attack scenarios, similar to the findings of previous ensemble and hybrid approaches [16], [18].

*d. Feature Importance*

The top 10 most important features identified by LightGBM are plotted in Fig. 6. These features provide valuable insights into the behavior of network traffic and the characteristics that are most indicative of various types of attacks.

*e. Precision-Recall Curve*

In Fig. 7 plot illustrates the trade-off between precision and recall for each class predicted by the three ensemble models. Higher area under the curve (AUC) values indicates better performance in distinguishing between classes, particularly in imbalanced data scenarios.



Arya Raj and Rakesh Kumar, "AI-Driven Ensemble Models for Anomaly and Intrusion Detection in Network Traffic," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 7, Issue 11, pp. 102-108, 2025.

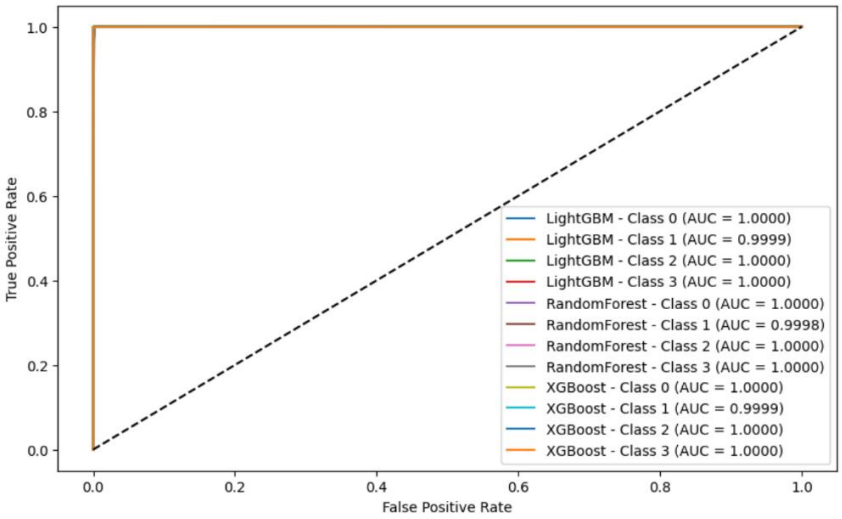


Fig. 5. Multiclass ROC Curve All Models (One-vs-Rest)

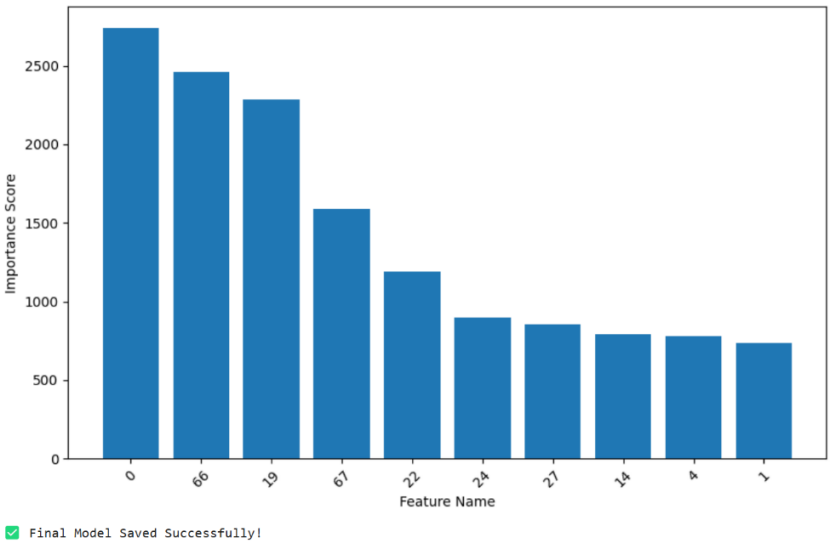


Fig. 6. Feature Importance for all models

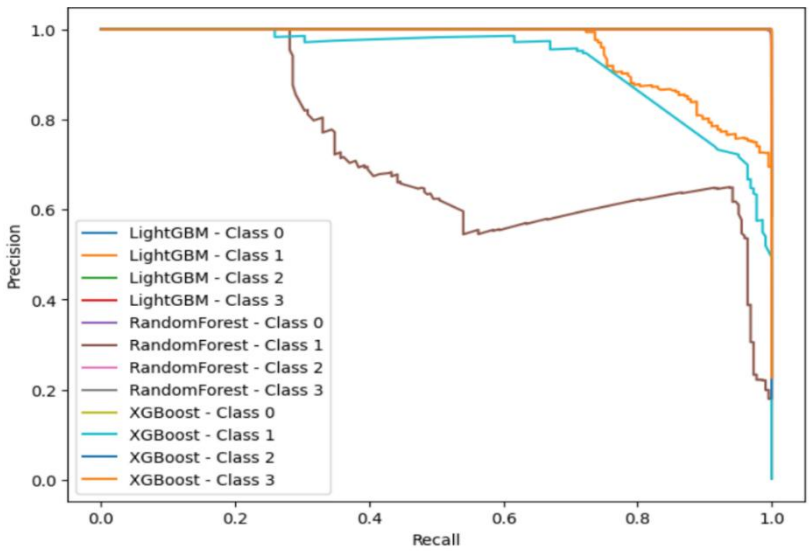


Fig. 7. Precision-Recall Curves for All Models



Arya Raj and Rakesh Kumar, "AI-Driven Ensemble Models for Anomaly and Intrusion Detection in Network Traffic," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 7, Issue 11, pp. 102-108, 2025.

## V. Conclusion

A complete methodology for network intrusion detection uses machine learning methods as presented in this study. Advanced ensemble learning models demonstrate their effectiveness for network anomaly classification when tested with LightGBM, RandomForest and XGBoost. A LightGBM model demonstrated the best performance regarding accuracy combined with swift execution speed which makes it suitable for actual cybersecurity implementations.

The research outcomes indicate that optimally detecting intrusions depends on how data is prepared alongside selection of appropriate features and comparative modeling methods. A complete evaluation of each model's efficiency can be acquired through using confusion matrices and classification reports together with ROC curves. These models demonstrate promising effectiveness for real-time implementation in intrusion detection systems because of their robust scalability features.

### *a. Future Work & Enhancements*

The research findings present promising results yet further developments would improve its effect through the following actions:

Implement SHAP values for detecting which features lead to the most impactful intrusion detection process.
Hyperparameter Optimization: Applying GridSearchCV or Bayesian Optimization to fine-tune model performance.
Deployment Considerations: Exploring the integration of these models into real-time IDS frameworks.
Future exploration may include using GANs to generate synthetic intrusion patterns for training, potentially improving minority class representation.

The proposed method receives testing against both traditional rule-based and signature-based intrusion detection systems for evaluation purposes.
Final Thoughts:

Research collaboration between machine learning systems produces automation techniques to maintain and improve high-security accuracy for intrusion detection. The developed method enhances threat detection capabilities while simultaneously with building an infrastructure for upcoming AI-based cybersecurity solutions.

## References

[1] "Analysis of Intrusion Detection Systems: Techniques, Datasets and Research Opportunity." [Online]. Available: https://ssrn.com/abstract=4749820
[2] Y. Alotaibi and M. Ilyas, "Ensemble-Learning Framework for Intrusion Detection to Enhance Internet of Things' Devices Security," Sensors, vol. 23, no. 12, Jun. 2023, doi: 10.3390/s23125568.
[3] 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, Oct. 2017, doi: 10.1109/ACCESS.2017.2762418.
[4] Q. Niyaz, W. Sun, A. Y. Javaid, and M. Alam, "A deep learning approach for network intrusion detection system," in EAI International Conference on Bio-inspired Information and Communications Technologies (BICT), 2015. doi: 10.4108/eai.3-12-2015.2262516.
[5] N. Shone, T. Nguyen Ngoc, V. Dinh Phai, and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," 2017.
[6] N. K. Gyimah et al., "An AutoML-based approach for Network Intrusion Detection," Nov. 2024, [Online]. Available: http://arxiv.org/abs/2411.15920
[7] Z. K. Maseer, R. Yusof, B. Al-Bander, A. Saif, and Q. K. Kadhim, "Systematic Review for Anomaly Network Intrusion Detection Systems: Detection Methods, Dataset, Validation Methodology, and Challenges."
[8] Z.-Z. Lin, T. D. Pike, M. M. Bailey, and N. D. Bastian, "A Hypergraph-Based Machine Learning Ensemble Network Intrusion Detection System," Nov. 2022, doi: 10.1109/TSMC.2024.3446635.
[9] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an Efficient Intrusion Detection System Based on Feature Selection and Ensemble Classifier," Apr. 2019, doi: 10.1016/j.comnet.2020.107247.
[10] I. Bibers, O. Arreche, and M. Abdallah, "A Comprehensive Comparative Study of Individual ML Models and Ensemble Strategies for Network Intrusion Detection Systems," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2410.15597
[11] Y. A. Farrukh, S. Wali, I. Khan, and N. D. Bastian, "XG-NID: Dual-Modality Network Intrusion Detection using a Heterogeneous Graph Neural Network and Large Language Model," Aug. 2024, [Online]. Available: http://arxiv.org/abs/2408.16021
[12] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection systems (UNSW-NB15 Network Data Set)." [Online]. Available: https://cve.mitre.org/
[13] X. Zhao, K. W. Fok, and V. L. L. Thing, "Enhancing Network Intrusion Detection Performance using Generative Adversarial Networks," Apr. 2024, [Online]. Available: http://arxiv.org/abs/2404.07464
[14] Md. A. Talukder et al., "A Dependable Hybrid Machine Learning Model for Network Intrusion Detection," Dec. 2022, doi: 10.1016/j.jisa.2022.103405.
[15] M. A. Hossain and M. S. Islam, "Ensuring network security with a robust intrusion detection system using ensemble-based machine learning," Array, vol. 19, Sep. 2023, doi: 10.1016/j.array.2023.100306.