AN OPTIMIZED XGBOOST FOR FALSE POSITIVE REDUCTION IN A NETWORK INTRUSION DETECTION

Adeola O Kolawole [1*], Emake S. Imokhai[2], and Martins E. Irhebhude [3]
Department of Computer Science, Faculty of Military Science and Interdisciplinary Studies, Nigerian Defence Academy, Kaduna Nigeria
Email: adeolakolawole@nda.edu.ng; emake.imokhai2021@nda.edu.ng; mirhebhude@nda.edu.ng

*Corresponding Author: Adeola O. Kolawole

Abstract

Cybersecurity operations are increasingly challenged by large volume of false alerts produced by Intrusion Detection Systems (IDS) which leads to analyst fatigue and increases the likelihood of missing real threats. This study proposes an optimized eXtreme Gradient Boosting (XGBoost) model designed to reduce false positives and improve operational reliability of IDS using University of New South Wales-Network Intrusion Detection System-15 (UNSW-NB15) dataset for validation of the model. The optimization included systematic hyperparameter tuning of key parameters such as learning rate, maximum tree depth, gamma, subsampling ratio, and L1/L2 regularization to balance model complexity and generalization. The performance of the model was evaluated against reproduced benchmark ensemble classifier under identical conditions. The benchmark achieved False Positive Rate (FPR) of 17.69%, while the proposed XGBoost model reduced it to 5.85%, representing a 66.9% improvement and 2,925 fewer false alerts on the test set. In real world deployment, this substantial deduction would significantly lower alert fatigue and enable timely and effective responses to genuine attacks. The most significant gain was observed in the classification of legitimate "Normal" traffic where the FPR decreased from 9.22% in the benchmark model to 0.12%. The results demonstrate that a single well-tuned XGBoost model can provide high accuracy (94.15%) while substantially improving operational dependability. This study shows that prioritizing false positive reduction offers a practical path toward building deployment-ready IDS solutions. The novelty of this research is in its emphasis on minimizing the false positive rate (FPR) over accuracy as the main performance metric.

Keywords: Intrusion Detection Systems, False Positive Reduction, XGBoost, Machine Learning, Cybersecurity, UNSW-NB15 Dataset.

1.0 INTRODUCTION

The digitalization of modern society has made networked systems the foundation for nearly every activity. While this connectivity enables innovation and efficiency, it has also significantly expanded the attack surface available to malicious actors (Gianchandani, 2023). Cyber-attacks have grown in sophistication and financial impact, now including wide spectrum of threats such as Denial-of-Service (DoS) attacks, data exfiltration, and backdoor installations (Becker et al., 2023). Intrusion Detection Systems (IDS) are crucial line of defense in this environment, designed to monitor network activity to detect malicious patterns and generate alerts when suspicious behavior is observed (Kumar, 2025). However, in practice, the effectiveness of many IDS implementations is challenged by overwhelming number of false alarms they produce (Alsubaei, 2025). A false positive, which occurs when an IDS incorrectly classifies legitimate benign network activity as malicious attack has become major barrier to effective cybersecurity. In real-world Security Operations Centers (SOCs), excessive false positives lead directly to "alert fatigue," a condition in which analysts become desensitized to alerts, resulting in slower incident response and, in severe cases, overlooked genuine threats (Becker et al., 2023). This operational challenge exposes significant disconnect between academic research metrics and real-world realities. In many studies, models are optimized primarily for overall accuracy, F1-score or any other metrics that aggregate correct predictions across all classes (Manjaragi et al., 2024). However, a model with high nominal accuracy can still be operationally ineffective if it misclassifies large volume of legitimate traffic (Kasongo & Sun, 2020). This leaves clear gap for research that explicitly measures and optimizes for detection reliability and low operational noise, not just raw detection accuracy.

Among modern machine learning methods, gradient boosting algorithms have demonstrated strong potential for balancing predictive power and efficiency. The eXtreme Gradient Boosting (XGBoost) algorithm in particular is state-of-the-art technique for tabular data problems. XGBoost is an ideal candidate for improving operational dependability. It incorporates both L1 (Lasso) and L2 (Ridge) regularization directly into its objective function which penalizes model complexity and helps to prevent overfitting (Alshehri et al., 2024). Since poor generalization is a common cause of false alarms, XGBoost's capacity for controlled regularization makes it an ideal candidate for improving the operational reliability of IDS models (Manjaragi et al., 2024). This paper addresses this gap by designing and validating a standalone optimized XGBoost classification model. The model is explicitly configured to emphasize generalization and false positive reduction rather than purely maximizing accuracy. Using the University of New South Wales-Network Benchmark 2015 (UNSW-NB15) dataset (Moustafa & Slay, 2015), a modern and complex 20 for evaluating IDS, the model's performance is rigorously compared against an Ensemble baseline established by Irhebhude et al. (2022). The study solely focused on the False Positive Rate (FPR) as a primary metric in order to bridge the gap between theoretical performance and real-world applicability and offer practical path toward more dependable, low-noise intrusion detection systems. Therefore, this study aims to develop an optimized eXtreme Gradient Boosting (XGBoost) model to minimize false positive rates while maintaining high accuracy for IDS systems leveraging on the UNSW-NB15 dataset.

The remainder of this paper is organized as follows: Section 2 describes the methodology, including the dataset, preprocessing steps, and model configurations. Section 3 presents the experimental results and discusses their

implications, with particular emphasis on false positive reduction. Section 4 concludes the paper and suggests directions for future research.

2.0 RELATED WORKS

This section reviews the most relevant studies in this domain, emphasizing their methodologies, key findings, and the persistent challenge of false positive reduction that motivates the present work.

The development of effective Intrusion Detection Systems (IDS) has been long-standing focus of cybersecurity research (Malik et al., 2024). Over the past decade, the integration of machine learning (ML) and deep learning (DL) techniques has transformed how network anomalies are detected and classified. Traditional rule-based IDS struggled to recognize emerging or unknown attack patterns, prompting researchers to explore data-driven methods capable of learning from traffic behaviors (Alshehri et al., 2024). Recent literature reveals an extensive body of work applying algorithms such as Decision Trees, Support Vector Machines, Random Forests, and various neural and Deep Learning architectures to benchmark datasets, including KDD99, NSL-KDD, and UNSW-NB15 (Becker et al., 2023).

Hooshmand, (2019) proposed a two-phase Network Anomaly (NAD) model that include the Random Forest (RF) and Neural Network to classify anomalies based on attack type using UNSW-NB15 dataset. The four subsets of the UNSW-NB15 were merge to form a single CSV file that contained more than 2 million instances that were used for the experiment. The two-phase model is made up of binary classifier and multi-classifier, the features of the dataset were categorized and the most important features were selected for the experiment. The binary classifier identified the instance attack or normal traffic with designations 1 and 0 respectively while the multi-classifier explicitly points out the attack category. Random Forest Classifier (RFC), Decision Tree Classifier (DTC), Gradient Boosting Classifier (GBC), KNN, Multinomial NB (MNB), SVM, Linear SVC (LSVC), Linear Discriminate Analysis (LDA), LR, CART and GNB algorithms were used to train and test on the dataset. RFC had the highest accuracy of 98% with an average precision, recall, and f1-score of 0.99. In the attack categorization phase, the model had an average precision of 93% precision and 88% recall.

Faker & Dogdu, (2019) investigated the performance of Random Forest (RF), Gradient Boosted Trees (GBT), and Deep Feed-Forward Neural Networks (DFNN) on big data using the University of New South Wales-Network Intrusion Detection System-15 (UNSW-NB15) and CICIDS2017 datasets. For the UNSW-NB15 dataset, RF and GBT achieved binary classification accuracies of 98.85% and 98.86%, respectively. In the multi-class scheme, the RF model achieved an accuracy of 91.04%. When applied to the CICIDS2017 dataset, the GBT model outperformed others with binary accuracy reaching 99.97% and DFNN achieving 99.57% in multi-class classification. The study highlighted the superior performance of ensemble and deep learning methods but recommended further research into better feature selection schemes.

Kocher & Kumar, (2020) conducted performance analysis of machine learning classifiers for intrusion detection using the UNSW-NB15 dataset. The KNNearest Neighbors (KNN), Stochastic Gradient Descent (SGD), Decision Tree (DT), RF, Logistic Regression (LR), and Naïve Bayes (NB) classifiers were selected for training. The effectiveness of these machine learning classifiers was assessed in terms of accuracy, precision, recall, F1-score, mean squared error (MSE), true positive rate (TPR), and false positive rate (FPR). Additionally, comparison analysis of these classifiers was carried out. The

outcome demonstrates that RF classifier outperforms other classifiers with an accuracy of 95.43%.

Kasongo and Sun, (2020) analyzed machine learning performance on the UNSW-NB15 dataset by employing a filter-based feature selection method using XGBoost. This approach reduced the feature space from 42 to 19 attributes to lower model complexity. By applying algorithms such as SVM, KNN, and Decision Trees (DT) to this reduced set, the authors demonstrated that the DT classifier's binary classification accuracy improved from 88.13% to 90.85%, validating that optimized feature vectors can enhance predictive capabilities.

Agarwal et al., (2021) evaluated NB, SVM, and KNN classifiers on a subset of 22,224 instances from the UNSW-NB15 dataset. Although the dataset contains multiple attack categories (such as Fuzzers, DoS, and Exploits), the study focused on binary classification approach (Attack vs. Normal). The SVM classifier achieved the highest performance with 97.78% accuracy, while KNN achieved 93% accuracy. The authors suggested that while traditional ML models perform well, further improvements could be realized using Deep Learning approaches.

Aleesa et al., (2021) proposed deep learning models based on ANN, DNN, and RNN for intrusion detection on the UNSW-NB15 dataset. The study compared binary and multi-class performance across these architectures. For binary classification, the ANN model achieved the highest accuracy of 99.26% with a loss of 1.51%, while the DNN achieved 99.22%. In the multi-class experiments, the DNN model demonstrated superior performance with an accuracy of 99.59%, significantly outperforming the RNN implementation which reported lower accuracies.

Choukri et al., (2021) propose a two-stage data assessment for anomaly identification in IoT network traffic using the UNSW-NB15 dataset. Two popular neural network algorithms, the long short-term memory (LSTM) and forward deep neural network (FDNN) algorithms, were used in the second stage of anomaly detection. The study's simulated experiments produced accurate results ranging from 90.66% to 64.12%.

Shahid et al., (2021) demonstrated the efficacy of Deep Learning-based IDS using Feed-Forward Neural Networks (FFNN), LSTM, and Randomized Neural Networks (RandNN) for IoT environments. The authors extensively tested various configurations of hidden layers and batch sizes. The FFNN achieved its best binary accuracy (99.93%) using two hidden layers with 64 neurons, and a multi-class accuracy of 98.72% with three hidden layers. For the LSTM model, a configuration of 128-128-128 neurons with a batch size of 64 yielded the highest binary accuracy of 99.89%. The RandNN model also showed promise, achieving 96.42% accuracy with a learning rate of 0.1, highlighting the impact of hyperparameter tuning on deep learning performance.

Irhebhude et. al., (2022) proposed an ensemble learning model to predict cyberspace uncertainties, experimenting on both the UNSW-NB15 dataset and a local penetration testing dataset. The ensemble model demonstrated robust prediction capabilities, achieving an accuracy of 99.4% on the local dataset and 99.1% on the UNSW-NB15 dataset. However, the study noted that while True Positive rates were high, the model produced a significant number of False Positives. The researchers recommended future work focus specifically on techniques to reduce False Positives and False Negatives.

Yin et al., (2023) proposed a hybrid feature selection method named IGRF-RFE, which combines Information Gain and Random Forest with Recursive Feature Elimination. Using the UNSW-NB15 dataset, this method identified an optimal subset of 20 numerical and 3 categorical features. The MLP model trained on these

features achieved an accuracy of 84.24% and a weighted F1-score of 82.85%, outperforming standalone feature selection methods. The study confirmed that hybrid feature selection effectively removes redundant features and improves IDS performance.

Becker et al., (2023) utilized the Edge-IIoTset dataset to compare eleven machine learning models for detecting fourteen types of IoT attacks. The results indicated that the AdaBoost (AB) model was the superior classifier, achieving an average accuracy and F1-score of 99.0%. Random Forest (RF) followed closely with 98.4% accuracy, while Logistic Regression (LR) and SVM performed poorly due to difficulties with high-dimensional data and class overlapping. The study noted that while "Normal" data was easily identified, specific attacks like Fingerprinting were frequently misclassified by weaker models.

Alanazi & Aljuhani, (2023) proposed an anomaly-based IDS for Industrial IoT (IIoT) using the X-IIoTID dataset. The authors implemented Minimum Redundancy Maximum Relevance (MRMR) and Neighborhood Components Analysis (NCA) for feature selection to reduce dimensionality. Comparison of various classifiers revealed that the Decision Tree (DT) model, combined with MRMR feature selection, achieved the highest performance with 99.58% accuracy and a very low False Positive Rate (FPR) of 0.4%.

Almarshdi et al., (2023) developed an IDS architecture combining Convolutional Neural Networks (CNN) and LSTM for binary classification on a balanced UNSW-NB15 dataset. To address data imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied. The proposed hybrid model was evaluated against standalone CNN, Decision Tree, and Random Forest models. The results demonstrated that the hybrid CNN-LSTM model outperformed the standalone classifiers, achieving an accuracy of 92.10% on the balanced dataset.

Anusha et al., (2024) proposed a Deep Learning-based CNN model for cyber threat detection in IoT-based Cyber-Physical Systems using the UNSW-NB15 dataset. The model was designed to automatically learn hierarchical features via convolutional and pooling layers. In a comparative evaluation, the CNN model significantly outperformed a Support Vector Machine (SVM), achieving 99.45% accuracy compared to the SVM's 86.01%, along with superior precision (99.35%) and recall (99.39%).

Ali et al., (2024) proposed stacking of artificial neural network (ANN), convolutional neural network (CNN), long short-term memory (LSTM), and recurrent neural network (RNN) (ACLR) implementing the UNSW-NB15 dataset. The proposed approach is based on model stacking where the output of ANN, CNN, LSTM, and RNN is used for the final prediction. Additionally, it is estimated how deep learning classification models performs when employed to analyze botnet attack detection using UNSW-NB15 dataset. The preprocessing is carried out to remove null values and handle categorical data using label encoding. The experimental setup involves ACLR implementation in the Google COLAB environment using the UNSW-NB15 dataset. In addition, this research employed ANN, CNN, LSTM, and RNN models for performance comparison with the proposed ACLR model. Experimental results suggest a superior performance of ACLR with a 0.9698 accuracy score while the k-fold cross-validation accuracy score is 0.9749 where the value of k is 3,5,7 and 10, respectively. In addition, increasing the number of layers for the deployed models is observed to produce better performance, however, it comes at the cost of increased computational complexity and higher training time. Among the employed models, the ANN model shows poor performance while LSTM, RNN, and CNN show better results. The comparative findings demonstrate that the proposed approach outperforms ANN, CNN, LSTM, and RNN models in terms of

performance and accuracy for the detection of botnets. In comparison to previous models, the suggested ACLR model has the greatest ROC AUC (0.9934) and PR AUC (0.9950) values. Performance analysis with existing models indicates that ACLR can perform better than state-of-the-art model. The researchers stated that, it is important to note that deep learning algorithms for botnet identification still have limitations such as a lack of labeled statistics on training and the possibility of hostile attacks. To increase the precision, scalability, and robustness of deep learning-based botnet detection systems more research and development in this area are required. The stacking model in the proposed research has been based on four deep learning models which consume more time than the single model in predictions but show more efficient results than the single model. It also necessitates data interchange and synchronization. This demonstrates the significance of carefully balancing model complexity and effectiveness across a range of applications. As it will be totally automated in future research, there should be more training using reinforcement learning, which can be more effective.

Alshehri et al., (2024) addressed the critical need for effective and efficient Intrusion Detection Systems (IDS) to detect botnet attacks, especially in IoT and Fog computing environments. A 1D-CNN and LSTM-based deep neural network with learnable skip connections was proposed and presented in this paper. This combination of convolutional and LSTM layers enabled the model to learn both temporal and spatial features in the data, while the learnable skip connections are capable of dynamically controlling the flow of information across the network, enabling the model to focus on salient features and ignore irrelevant ones, thus enhancing its detection capabilities. The proposed model was trained and tested on actual IoT network traffic data (the N-BaIoT dataset). This dataset features authentic traffic data from nine commercial IoT devices, including cameras, routers, and smart home appliances infected with the Mirai and BASHLITE malware, incorporating a total of 10 different IoT attacks. With a compact size of 2596.87 KB, an inference time of 8.0 milliseconds, and a test accuracy of 99.91%, the proposed model proved to be well-suited to be deployed in resource constrained environments. The proposed SkipGateNet model outperformed all models in comparison in terms of accuracy and inference time. Furthermore, future research could explore the integration of SkipGateNet with federated learning for distributed IoT environments, and the application of transfer learning to enhance its adaptability to different IoT domains and attack types.

In another study, Dash et al., (2024) examined the NSL-KDD dataset and how machine learning algorithms can detect distributed denial of service (DDoS) assaults on Internet of Things (IoT) devices. The researchers looked at how well six different ML classifiers could identify DDoS attacks. The researchers tested the classifiers both with and without principal component analysis (PCA) application. When it came to correctly identifying DDoS attacks, the results demonstrated that the Random Forest classifier routinely attained the greatest values for precision, recall, F1-score, accuracy, and kappa coefficients. While Naïve Bayes showed relatively poor performance, the K-Nearest Neighbour and Decision Tree classifiers both showed strong results. In most cases, PCA enhanced the classifiers' performance, which in turn increased their accuracy, recall, F1-score, precision, and kappa coefficient values. The experiment outcomes demonstrated a noteworthy improvement in the accuracy of DDoS attack detection in IoT devices by integrating PCA and Robust Scaler. Notably, the Random Forest and KNN classifiers demonstrated exceptional performance with an accuracy of 99.87 % and 99.14 %, respectively, while Naïve Bayes gave a lower accuracy of

87.14 %. The findings from this experiment contributed valuable insights into enhancing the security of IoT devices against DDoS attacks. The proposed approach showcased the importance of appropriate preprocessing techniques in achieving robust intrusion detection systems for IoT environments.

Malik et al., (2024) developed an intelligent IDS by employing a filter-based feature selection technique named Mean Absolute Difference (MAD) to select important features from the UNSW-NB15 and NSL-KDD datasets. They deployed a range of machine learning classifiers, including MLP, CatBoost, and LGBM, on the reduced feature list. On the UNSW-NB15 dataset, the LGBM classifier produced the most promising results among the tested models, achieving 98.70% accuracy.

More et al. (2024) focused on enhancing Intrusion Detection System (IDS) performance by applying several supervised machine learning algorithms to the UNSW-NB15 dataset. Their methodology involved in-depth exploratory data analysis and feature selection using correlation analysis to address the high number of false positives common in IDS. They evaluated Logistic Regression, SVM, Decision Tree, and Random Forest models. Their findings indicate that the Random Forest model was the most effective for identifying cyber-attacks, achieving an accuracy of 98.63% and a low false alarm rate of 17.36%.

Sugin & Kanchana, (2024) investigated the challenge of imbalanced data and high dimensionality in IDS by implementing a filter-based attribute reduction approach using XGBoost on the UNSW-NB15 dataset. The resulting condensed feature space was then used to train SVM, LR, kNN, DT, and CNN models for both binary and multiclass classification. Their results showed that this feature selection approach significantly enhanced performance, with the Decision Tree model's accuracy on the binary classification task improving from 88.13% to 90.85%.

Alsubaei (2025) addressed the limitations of conventional machine learning models in handling sophisticated and evolving cyber threats. The study applied extensive preprocessing steps and hyperparameter tuning using Grid Search for both XGBoost and an Optimized Sequential Neural Network (OSNN) on the NSL-KDD, UNSW-NB15, and CICIDS2017 datasets. The optimized SNN model performed exceptionally well on the UNSW-NB15 dataset, achieving an accuracy of 96.80%. This highlights the potential of meticulously tuned deep learning models for improving multiclass intrusion detection.

Parambil & Krishna, (2025) investigated the detection of zero-day attacks by evaluating the performance of several machine learning models, including AdaBoost, XGBoost, Random Forest, and a Hybrid Ensemble Model, on the UNSW-NB15 dataset. Recognizing the inefficiency of traditional models with complex, multi-class datasets, this work focused on flexibility and precision. The study found that a Hybrid Ensemble Model, which combined the strengths of multiple classifiers, achieved the highest accuracy at approximately 81.92%, demonstrating its robustness for generating exact classifications.

The reviewed literature demonstrates clear progress in the use of machine learning and, ensemble methods for intrusion detection. Traditional models provided interpretability but struggled with complex network behaviors. Deep learning models improved detection accuracy but suffered from opacity, overfitting, and high computational demands. XGBoost in particular have provided compelling compromise by combining interpretability, speed, and regularization mechanisms that help reduce false positives. However, despite these advances consistent pattern across most studies revealed that models are optimized primarily for accuracy or F1-score with less emphasis on operational reliability metrics such as the False Positive Rate. This paper builds upon study by Irhebhude et al. (2022) that established valuable

baseline for applying Ensemble Model to intrusion detection using Decision Trees and Random Forest. The proposed study will evaluate a single optimized XGBoost model that explicitly prioritizes false positive reduction as its primary objective, offering more practical and deployable approach to modern intrusion detection.

3.0 METHODOLOGY

This study employs methodological framework adopted to develop, evaluate, and compare the performance of the proposed intrusion detection model. It outlines the overall research design, the dataset used, data preprocessing procedures, model implementation details, evaluation metrics, and the experimental workflow. The methodological foundation of this study builds upon the work of Irhebhude et al., (2022), who implemented Ensemble models for network intrusion detection using the UNSW-NB15 dataset (Sarhan et al., 2020). While their approach achieved commendable classification accuracy, the resulting models produced high rate of false positives. This limitation motivates the current study's objective to develop an optimized standalone eXtreme Gradient Boosting (XGBoost) model that maintains high detection accuracy while significantly reducing false alarms. Accordingly, this study details how the proposed model was systematically designed, trained, and evaluated against the benchmark established by Irhebhude et al., (2022). The design emphasizes replicability and practical applicability, ensuring that the results can inform both academic research and real-world cybersecurity operations.

3.2 Research Design

The design follows comparative modeling framework consisting of the following sequential stages as shown in Figure 1:

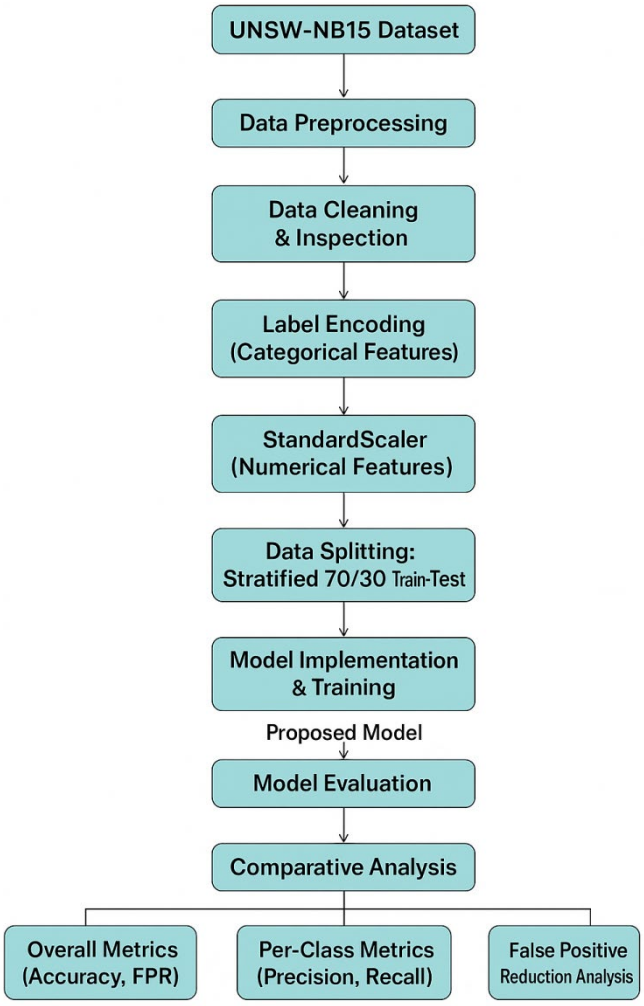

Figure 1: Research Design

The experimental workflow was executed using the Python programming language (version 3.10) and libraries such as Scikit-learn, XGBoost, NumPy, and Pandas within Jupyter Notebook environment. This framework ensures reproducibility and transparency in data handling and analysis.

3.3 Dataset Description

The UNSW-NB15 dataset by Moustafa & Slay, (2015) was used in this study The dataset was specifically designed to overcome the limitations of older intrusion detection benchmarks such as KDD'99 and NSL-KDD by incorporating modern network traffic

characteristics and contemporary attack behaviors. The UNSW-NB15 dataset contains blend of real and synthetic network traffic captured using IXIA PerfectStorm tool within hybrid testbed. It comprises 2,540,044 records, each described by 49 features that encompass packet-level attributes (e.g., source bytes, destination bytes), protocol-based information (e.g., TCP, UDP, ICMP), and time-related features. the

Each record is labeled as either Normal or one of nine attack categories as presented in Table 1:

Table 1: UNSW-NB15 Dataset Distribution

Original Class distribution	**No of Samples**
Analysis:	677
Backdoor:	583
DoS:	4,089
Exploits:	11,132
Fuzzers:	6,062
Generic:	18,871
Normal:	37,000
Reconnaissance:3,496 samples	3,496
Shellcode:	378
Worms:	44

Although the full dataset contains over 2.5 million records, this study uses only 82,332 training samples selected specifically because they correspond to the attack classes relevant to the research scope. Features and records not associated with the target attack categories were excluded, as they do not contribute to the objectives of this study.

3.4 Data Preprocessing and Preparation

Prior to model training, a critical data cleaning and preprocessing phase was executed. This includes:

3.4.1 Handling Missing and Null Values: Records containing null or non-numeric entries were identified. Since the dataset is large, any records with missing critical feature values were removed to maintain data quality.

3.4.2 Feature Encoding: All categorical features (such as `proto`, `service`, and `state`) were converted into numerical format suitable for XGBoost using One-Hot Encoding.

3.4.3 Feature Scaling: Numerical features were standardized using Min-Max Scaling to normalize their range between 0 and 1, ensuring that no single feature dominates the model training based purely on magnitude.

3.4.4 Data Splitting

After preprocessing, the dataset was partitioned into two subsets, 70% for training and 30% for testing. Stratified sampling was employed to preserve the proportional representation of each traffic class across both subsets.

3.5 Model Development and Implementation

The proposed model employed eXtreme Gradient Boosting (XGBoost), an advanced implementation of gradient boosting optimized for speed, scalability, and regularization. XGBoost constructs sequence of decision trees where each subsequent tree corrects the residual errors of the preceding ensemble. Its design includes built-in regularization mechanisms that improve generalization and help prevent overfitting which is critical factor in reducing false positives. Mathematically, Gradient Boosting can be understood as performing gradient descent in function space rather than parameter space. Its defined in Equation 1 as (Chen & Guestrin, 2016):

$$D = \{(x_i, y_i)\}_{i=1}^{n} \qquad (1)$$

where x_i represents the feature vector for the i^{th} sample, and y_i is the corresponding class label.

The model is constructed as an additive combination of weak learners $f_k(x)$ in

Equation 2 (Chen & Guestrin, 2016):

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i) \qquad (2)$$

Each function f_k represents decision tree in the ensemble. The objective of Gradient Boosting is to minimize a loss function $L = (y_i, \hat{y}_i)$, which measures the discrepancy between the true labels and predictions, along with a regularization term $\Omega(f_k)$that penalizes model complexity in Equation 3 (Chen & Guestrin, 2016):

$$L = \sum_{i=1}^{n} L = (y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \qquad (3)$$

At each boosting round *t*, the model computes the pseudo-residuals, which represent the negative gradients of the loss function with respect to the current predictions (see Equation 4) (Chen & Guestrin, 2016):

$$r_{it} = -\left[\frac{\mathrm{L}(y_i, \hat{y}_i)}{\partial \hat{y}_i}\right]_{\hat{y}_i = \hat{y}_i{}^{(t-1)}} \qquad (4)$$

A new weak learner $f_t(x)$ is then trained to predict these residuals. The ensemble is updated as shown in equation 5 (Chen & Guestrin, 2016):

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta f_t(x_i) \qquad (5)$$

where $\eta \in (0,1]$ is the learning rate which controls the contribution of each new tree to the model. Lower values of η slow down learning but improve generalization. This iterative process continues until the loss converges or predefined number of trees *K* is reached.

3.5.1 Incorporating Regularization

The proposed model utilizes eXtreme Gradient Boosting (XGBoost) algorithm and were optimized via systematic hyperparameter tuning approach. This optimization was specifically targeted at improving model generalization and reducing high False Positive Rate (FPR) characteristic of many IDSs. Key parameters employed includes learning rate (η), maximum tree depth (max_depth) and regularization parameters. To prevent overfitting and enhance model's ability to generalize to unseen network traffic, L1 (reg_alpha) and L2 (reg_lambda) regularization were explicitly incorporated into the objective function during tuning phase. This addresses common issue of complex models over-fitting to the training noise which often results in higher False Positive rates on test set.

- The L1 regularization (reg_alpha) term adds penalty equal to the absolute value of magnitude of coefficients. This technique encourages feature sparsity, effectively performing feature selection by driving the weights of less important features towards zero.
- The L2 regularization (reg_lambda) term adds penalty equal to square of magnitude of coefficients. This technique encourages the learning of smaller, smoother weights which helps to smooth decision boundaries and prevent any single tree from dominating prediction in order to improve overall generalization.

The optimal values for reg_alpha and reg_lambda were determined through an iterative search process (such as Grid Search or Random Search) to ensure robust trade-off between model complexity and predictive performance. The final configuration used in this study is in Table 2 as follows:

Table 2: XGBoost Hyperparamers tuning

Parameter	Value	Purpose
n_estimators	800	Defines the number of trees to ensure stable learning convergence.
max_depth	7	Controls model complexity and prevents overfitting.
learning_rate	0.05	Determines the contribution of each tree to the final model.
subsample	0.8	Introduces randomness to improve generalization.
colsample_bytree	0.8	Controls feature sampling per tree.
reg_alpha (L1)	0.3	Penalizes large coefficients to reduce overfitting.
reg_lambda (L2)	0.3	Stabilizes weight updates for better generalization.
gamma	0.2	Specifies the minimum loss reduction required for a split.
min_child_weight	3	Ensures that leaves have sufficient samples to prevent overfitting.

The model was trained using the training dataset and evaluated against the test dataset. Optimization focused on achieving low false positive rate (FPR) while maintaining strong overall classification accuracy.

3.6 Evaluation Metrics

To comprehensively assess model performance, a combination of standard classification metrics and operational reliability measures was used.

Accuracy: accuracy measures the overall correctness of the model's predictions. It is defined as the ratio of correctly predicted instances (true positives and true negatives) to the total number of instances as shown in equation 9 (Irhebhude et al., 2022).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

Where TP = True Positives (correctly classified positive instances), TN = True Negatives (correctly classified negative instances), FP = False Positives (incorrectly classified as positive), FN = False Negatives (incorrectly classified as negative).

Precision (Positive Predictive Value): precision measures the proportion of correctly predicted positive instances out of all instances predicted as positive as shown in equation 10 (Kolawole et al., 2025). It is particularly important in scenarios where false positives are costly.

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

High precision means fewer false positives, which is desirable in applications where the consequences of incorrectly predicting positive class (e.g., falsely identifying an emotion) are significant (Tariq et al., 2023).

Recall (Sensitivity or True Positive Rate): recall, also known as sensitivity, measures the proportion of actual positive instances that were correctly identified. It is useful in scenarios where missing a positive instance (false negative) is more costly than incorrectly classifying a negative one (Sugin & Kanchana, 2024). (equation 11).

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

High recall means the model is effective in identifying the true positive cases, reducing the likelihood of false negatives.

F1-Score: F1-score is the harmonic mean of precision and recall, offering a balanced measure when there is an uneven class distribution. It gives a better sense of the model's effectiveness by considering both precision and recall, especially when one metric is much lower than the other (Pansari et al., 2024). (equation 12).

$$F1 - score = \frac{Precision \times Recall}{Precision + Recall} \qquad (12)$$

The F1-score is particularly useful when both false positives and false negatives are critical, as it balances both aspects of classification performance. A high F1-score indicates strong performance in both precision and recall (Faker & Dogdu, 2019).

False Positive Rate (FPR): Measure the proportion of normal instances incorrectly classified as attacks. (see equation 13).

$$FPR = \frac{FP}{PP + TN} \qquad (13)$$

3.8 Ethical Considerations

Although this research is computational in nature and does not involve human participants or sensitive personal data, ethical standards were observed throughout all stages of the study to ensure transparency, reproducibility, and responsible data use.

The dataset employed, UNSW-NB15, is a publicly available resource developed and maintained by the Australian Centre for Cyber Security. Its use complies fully with open research and data-sharing policies. No modification or manipulation of the dataset was performed beyond standard preprocessing procedures necessary for model training and evaluation. All data handling activities were confined to maintaining analytical accuracy and integrity.

To ensure ethical research practice, the following principles was observed:

Data Integrity and Authenticity:

The original structure, distribution, and labeling of the dataset were preserved. Data cleaning and preprocessing steps were documented transparently to prevent misrepresentation or selective reporting of results.

4.0 RESULTS AND DISCUSSION

The experimental results presented in this section demonstrate the performance advantage of optimized XGBoost model particularly in the critical domain of false positive reduction. The model was trained on 70% of the UNSW-NB15 dataset and rigorously evaluated on the remaining 30% test sets. Evaluation process focused on core metrics of accuracy, precision, recall, F1-score, and most critically, False Positive Rate (FPR) to assess model's operational reliability. The proposed XGBoost model achieved an overall testing accuracy of 94.15%, demonstrating robust learning capabilities. The model's training accuracy of 97.12% yielded a narrow 2.97% generalization gap from the testing accuracy. This small gap suggests that the systematic hyperparameter tuning, including the application of L1/L2 regularization, successfully created balanced learning configuration and avoided common issue of overfitting that often plagues IDS models.

4.2 Performance Evaluation of the Proposed XGBoost Model

Figure 2 highlights the model's detection capability across both normal and attack traffic, the optimized XGBoost model demonstrated higher overall performance, achieving testing and training accuracies of 94.15% and 97.12%

respectively. This strong performance demonstrates the model's capacity to accurately identify both legitimate and malicious traffic which is critical factor for the practical deployment of intrusion detection systems. The model's training accuracy of 97.12%, yielded narrow 2.97% gap from the testing accuracy. The results suggests that the tuned hyperparameters and built-in L1/L2 regularization achieved balanced learning configuration that avoids overfitting that is regarded as a common problem in IDS models that can inflate accuracy at the cost of real-world reliability.

```
PROPOSED XGBOOST MODEL TRAINING
======================================
Training XGBoost...
[0]     validation_0-mlogloss:2.11659
[50]    validation_0-mlogloss:0.40345
[100]   validation_0-mlogloss:0.27055
[150]   validation_0-mlogloss:0.24380
[200]   validation_0-mlogloss:0.22948
[250]   validation_0-mlogloss:0.21846
[300]   validation_0-mlogloss:0.20985
[350]   validation_0-mlogloss:0.20251
[400]   validation_0-mlogloss:0.19585
[450]   validation_0-mlogloss:0.18991
[500]   validation_0-mlogloss:0.18466
[550]   validation_0-mlogloss:0.18002
[600]   validation_0-mlogloss:0.17575
[650]   validation_0-mlogloss:0.17166
[700]   validation_0-mlogloss:0.16786
[750]   validation_0-mlogloss:0.16436
[799]   validation_0-mlogloss:0.16143
XGBoost training completed!

TRAINING RESULTS:
Training Accuracy: 0.9712
Testing Accuracy:  0.9415
```

Figure 2: XGBoost Training Process.

4.2.1 Classification Report and Model Behavior

The model achieved near-perfect precision (99.8%) and recall (100%) for the Normal class as highlighted in Figure 3, demonstrating exceptional ability to distinguish legitimate traffic from malicious activity. Moreover, it reached high detection for several critical attack types such as Fuzzers and Backdoors, showing that the model accurately learns discriminative patterns and can generalizes effectively across diverse intrusion categories.

PROPOSED XGBOOST CLASSIFICATION REPORT

	precision	recall	f1-score	support
Analysis	1.0000	0.9951	0.9975	203
Backdoor	1.0000	1.0000	1.0000	175
DoS	0.9918	0.9813	0.9865	1227
Exploits	0.8337	0.7311	0.7791	3340
Fuzzers	1.0000	1.0000	1.0000	1819
Generic	0.8477	0.9131	0.8792	5661
Normal	0.9986	1.0000	0.9993	11100
Reconnaissance	0.9981	0.9838	0.9909	1049
Shellcode	1.0000	1.0000	1.0000	113
Worms	0.0000	0.0000	0.0000	13
accuracy			0.9415	24700
macro avg	0.8670	0.8604	0.8632	24700
weighted avg	0.9409	0.9415	0.9405	24700

Figure 3: Classification Report of the Proposed Model

The model exhibits strong diagonal dominance in the confusion matrix, as shown in Figure 4, indicating high degree of correct classification across all ten categories. For this experiment, a total of 82,332 instances were used. The dataset was split using 70/30 ratio, with 70% (57,632 samples) allocated for training and 30% (24,700 samples) reserved for testing. This distribution ensured sufficient representation of each attack category during both phases. The confusion matrix demonstrates XGBoost model's effectiveness in learning complex traffic characteristics and forming clear decision boundaries, particularly between classes that commonly overlap in feature space such as Exploits and Generic attacks.

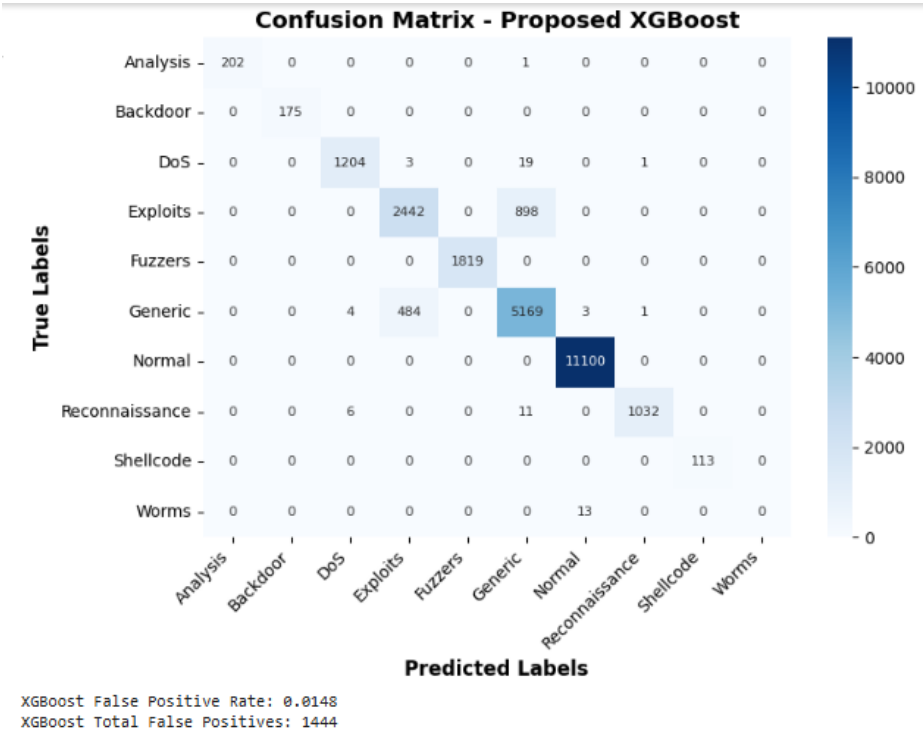

Figure 4: Confusion Matrix of the Proposed Model

The analysis of the Model's confusion matrix in Figure 4 confirms high degree of correct classification alignment across all ten network traffic categories. This pattern confirms the XGBoost model's effectiveness at learning complex traffic characteristics and forming distinct decision boundaries. Most critically, the classification of Normal traffic achieved 100% Recall and 99.8% Precision, indicating that all 111,000 instances of legitimate traffic were correctly classified. This outcome highlights that the proposed model was able to eliminates misclassification of benign traffic as malicious. Furthermore, the model demonstrated near-perfect detection for several attack types. Backdoor and Fuzzer attacks both achieved a perfect 100% Score in both Precision and Recall. Specifically, all 175 Backdoor attacks were detected without misclassification, and the model showed exceptional sensitivity to fuzzing behaviors with zero misclassified instances. Analysis attacks were robustly identified with 100% Precision and 99.5% Recall, registering only one false positive among 202 correct identifications. For DoS attacks, the model showed high recognition with 1,204 correct classifications, with only minor misclassification into Exploits (3 instances) and Generic (19 instances). Exploits saw 2,442 accurate detections despite slight overlap with Generic attacks (898 cases) which accounts for the most significant cross-category confusion observed. Reconnaissance attacks were accurately classified 1,032 times, with minor confusion involving DoS (6) and Generic (11) instances. Finally, the minority attack categories, Shellcode and Worms, were also successfully classified with high precision, demonstrating that the optimized XGBoost effectively manages class imbalance and avoids the detrimental bias toward majority classes that often leads to near-zero recall in benchmark ensemble models.

The results confirm that the proposed model

enhances false positive detection precision and also significantly limits cross-category confusion. The high recall values for Normal and major attack classes demonstrate reduced sensitivity to benign deviations which are common cause of false positives. The near-perfect classification of Fuzzers, Backdoor, and Analysis categories reflects the model's adaptive gradient boosting capacity. The small number of misclassifications between Exploits and Generic traffic suggests these categories share closely aligned feature distributions, possibly due to the overlapping nature of payload and exploit vectors.

4.3 Performance Evaluation on Benchmark Ensemble Model

The Irhebhude et. al, (2022) ensemble model was implemented using UNSW-NB15 dataset to serve as the baseline. The model achieved lower accuracy of 82.31% as shown in Figure 5, revealing significant limitations in classification consistency.

Benchmark Ensemble Model Classification Report:

	precision	recall	f1-score	support
Analysis	0.2917	0.0345	0.0617	203
Backdoor	0.0000	0.0000	0.0000	175
DoS	0.4161	0.4425	0.4289	1227
Exploits	0.6073	0.7057	0.6528	3340
Fuzzers	0.5804	0.3452	0.4330	1819
Generic	0.9863	0.9684	0.9773	5661
Normal	0.8942	0.9551	0.9237	11100
Reconnaissance	0.7163	0.6787	0.6970	1049
Shellcode	0.0000	0.0000	0.0000	113
Worms	0.0000	0.0000	0.0000	13
accuracy			0.8231	24700
macro avg	0.4492	0.4130	0.4174	24700
weighted avg	0.8063	0.8231	0.8107	24700

Figure 5 Benchmark Model's Classification Report

Despite acceptable performance in dominant classes like Normal (89% Precision, 95% Recall) and Generic (98% Precision, 96% Recall), the model exhibited weaknesses in minority and borderline categories often confusing normal traffic with malicious activities. The benchmark model's broad misclassification patterns demonstrate its inability to maintain distinct separation between normal and attack traffic. The poor recall for minority attacks (Backdoor (0%), Shellcode (0%), Worms (0%)) suggests that the model prioritizes majority-class learning, which is common ensemble bias that arises when weak learners fail to jointly adapt to underrepresented patterns.

4.3.2 Benchmark Confusion Matrix

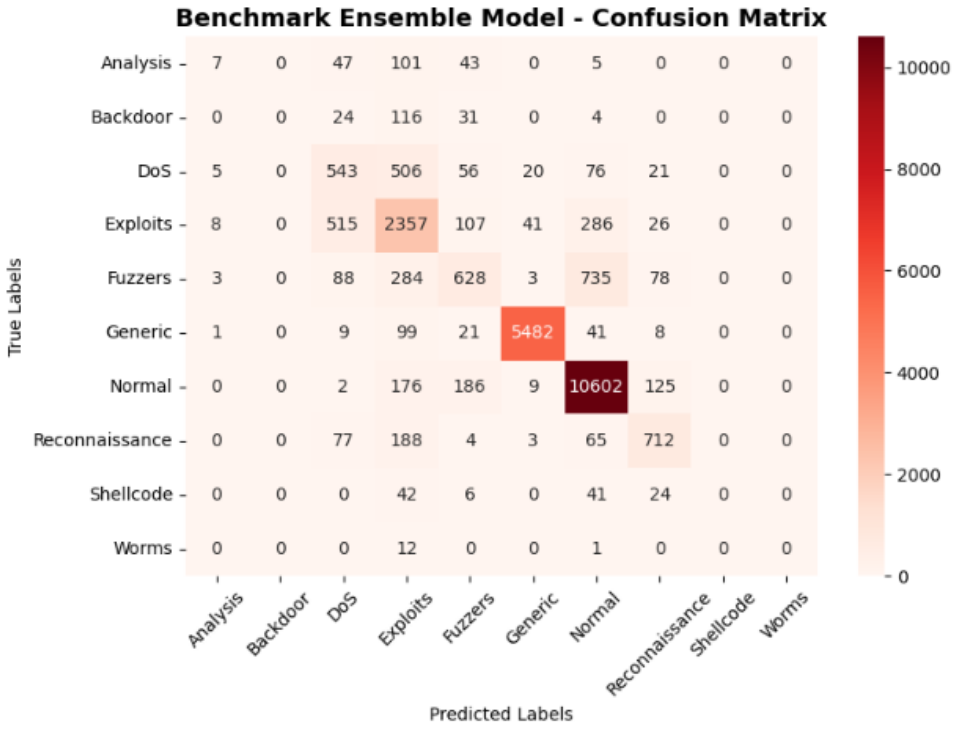

Figure 6: Benchmark Model's Confusion Matrix

The confusion matrix of the benchmark ensemble model, as shown in Figure 6, exposes widespread classification inconsistencies that directly translate to operational unreliability. The most significant finding was the model's failure to correctly handle benign traffic, where Normal traffic exhibited substantial misclassification being incorrectly flagged as Exploits (176 instances) and Fuzzers (186 instances). This outcome clearly indicates an inadequate feature separation between benign and anomalous activities. Similarly, the detection of DoS attacks was unreliable, recording high confusion rates with Exploits (506 instances), Fuzzers (56 instances), Generic (20 instances), and even Normal traffic (76 instances). This reveals poor ability to recognize critical denial-of-service patterns. The high frequency of misclassifications between the Generic and Exploit categories further demonstrates ensemble's failure to capture differences in their underlying payload-based behaviors. These results collectively highlight limitations of ensemble averaging methods in which multiple base learners contribute to decisions that may conflict when features exhibit interdependencies. Consequently, the benchmark ensemble produces inconsistent class boundaries that results in both missed detections and an inflated false positive count. This instability translates to excessive false alarms and potential oversight of critical threats which is a major disadvantage in real-world deployments where precision in alert generation is vital. The confusion between Exploits and Normal traffic is particularly detrimental as it reflects the model's fundamental uncertainty in distinguishing benign user behavior from subtle exploitation attempts.

4.4 False Positive Reduction and Comparative Performance

The primary objective of this study was to reduce operational noise in intrusion detection

process measured through False Positive Rate (FPR). The results show that the optimized XGBoost model delivers substantial improvement over benchmark ensemble classifier. Benchmark model generated 4,375 false positives which corresponds to an overall FPR of 17.69% across all attack categories. In contrast, the proposed XGBoost model produced only 1,458 false positives, reducing the overall FPR to 5.90%. This reflects a 66.67% reduction in false alarms, amounting to 2,917 fewer false alerts on the test set. Figure 7 which illustrates Class-specific comparisons further highlight these improvements. The proposed model achieved zero false positives for Backdoor, Fuzzers, Shellcode, and Normal traffic which represents substantial gains over benchmark classifier. Significant reductions were also observed for Analysis (0.0080 → 0.0000), DoS (0.0293 → 0.0018), and Reconnaissance (0.0142 → 0.0005). The Worms class reflects that both models recorded 13 false positives, yielding an identical FPR of 0.0005 and therefore no improvement for this category. One exception to the overall trend is the Generic class, where the FPR increased from 0.0094 in the benchmark model to 0.0258 in the XGBoost model. Despite this single-class degradation, the overall performance strongly favors the proposed approach. Furthermore, the substantial reduction in false positives, particularly the complete elimination of false alerts in key categories such as Normal demonstrates the improved reliability, precision, and operational value of optimized XGBoost intrusion detection model.

Class	Proposed (XGB) FP Count	Benchmark FP Count	Proposed FPR	Benchmark FPR	Improvement
Analysis	1	196	0.0000	0.0080	+0.0080
Backdoor	0	178	0.0000	0.0073	+0.0073
DoS	43	687	0.0018	0.0293	+0.0274
Exploits	898	983	0.0420	0.0460	+0.0040
Fuzzers	0	1,191	0.0000	0.0521	+0.0521
Generic	492	179	0.0258	0.0094	–0.0164
Normal	0	498	0.0000	0.0366	+0.0366
Reconnaissance	11	337	0.0005	0.0142	+0.0138
Shellcode	0	113	0.0000	0.0046	+0.0046
Worms	13	13	0.0005	0.0005	+0.0000
		Total False Positives	**1,458**	**4,375**	**+2,917**
		Overall Error Rate	**0.0590**	**0.1769**	**+0.1181**
		False Positive Reduction			**+66.67%**

Figure 7: False Positive Analysis Comparison Report

Figure 8 visually confirms the proposed model's efficacy in minimizing the number of legitimate traffic instances incorrectly flagged as malicious. The improved XGBoost model were able to achieve overall accuracy of 94.15% against the benchmark model's 82.31%, at the same time reduced the overall false positive rate from 17.69% in the benchmark to 5.85%.

The results validate the central hypothesis that systematically prioritizing generalization through regularization and tuning in a powerful single model like XGBoost can offer more effective deployment-ready solution than complex ensemble methods, especially when the goal is operational reliability.

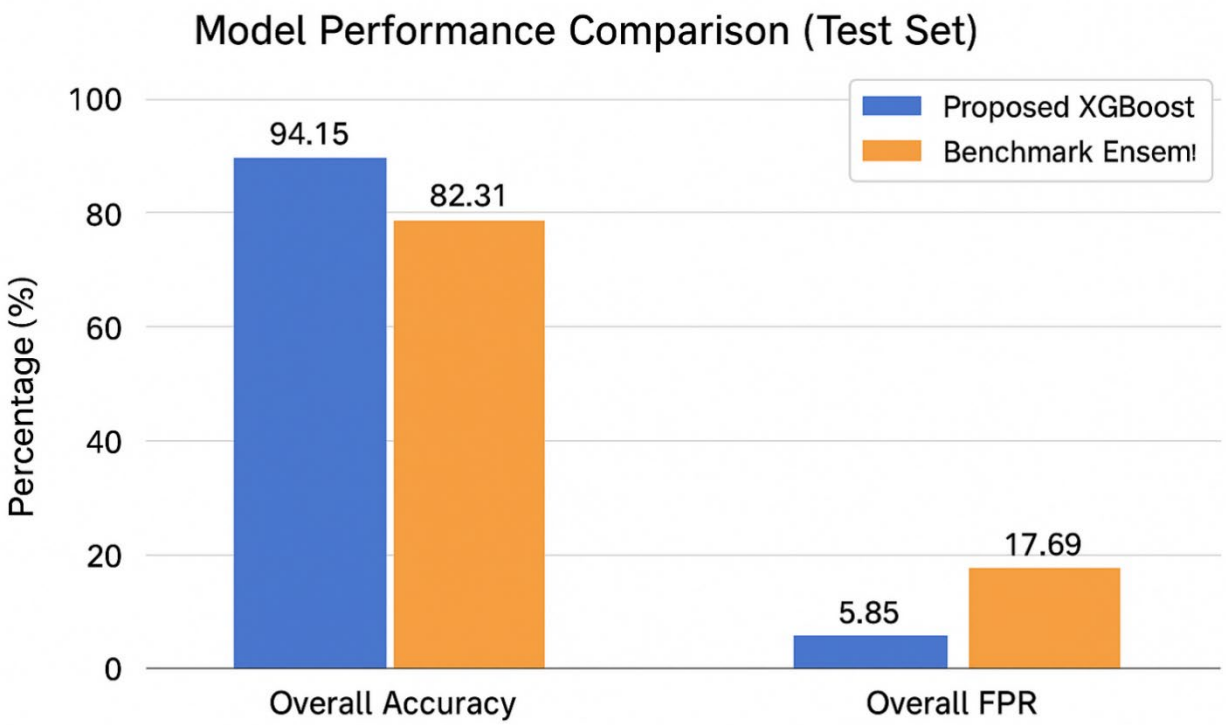

Figure 8: Model Performance Comparison.

4.5 Contribution to Knowledge

This study contributes new evidence that targeted optimization of gradient boosting classifiers can materially reduce false positives on standard IDS benchmark. The main empirical contributions are:

1. Rigorously tuned XGBoost configuration that attains 94.15% testing accuracy

2. Demonstration of 66.9% reduction in overall FPR relative to the Ensemble benchmark3. detailed per-class analysis that highlights the operational gains for Normal traffic classification (FPR reduced from 9.22% to 0.12%). The work reframes IDS evaluation toward metrics that matter to practitioners and provides practical baseline for future research.

5.0 CONCLUSION

This study addressed one of the most persistent challenges in intrusion detection, which is the high rate of false positives that lead to alert fatigue and reduce the effectiveness of cybersecurity operations. The proposed XGBoost model achieved both high accuracy and exceptional reliability, reducing the overall false positive rate by 66.9% and nearly eliminating false alarms on legitimate traffic. These results demonstrate that prioritizing operational reliability over raw accuracy offers more practical and sustainable path for advancing intelligent IDS development. The approach outlined here provides validated

framework for building models that are accurate, trustworthy, efficient, and ready for real-world deployment. Further evaluation on real-time traffic and additional benchmark datasets is recommended to validate robustness and generalizability of the model.

REFERENCES

Agarwal, A., Sharma, P., Alshehri, M., Mohamed, A. A., & Alfarraj, O. (2021). Classification model for accuracy and intrusion detection using machine learning approach. *PeerJ Computer Science*, *7*.

Alanazi, R., & Aljuhani, A. (2023). Anomaly Detection for Industrial Internet of Things Cyberattacks. *Computer Systems Science and Engineering*, *44*(3).

Al-Daweri, M. S., Ariffin, K. A. Z., Abdullah, S., & Senan, M. F. E. M. . (2020). An analysis of the KDD99 and UNSW-NB15 datasets for the intrusion detection system. *Symmetry*, *12*(10).

Aleesa, A. M., Younis, M., Mohammed, A. A., & Sahar, N. M. (2021). Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques. *Journal of Engineering Science and Technology*, *16*(1).

Ali, M., Shahroz, M., Mushtaq, M. F., Alfarhood, S., Safran, M., & Ashraf, I. (2024). *Hybrid Machine Learning Model for Efficient Botnet Attack Detection in IoT Environment*. IEEE Access. 40682–40699.

Almarshdi, R., Nassef, L., Fadel, E., & Alowidi, N. (2023). Hybrid Deep Learning Based Attack Detection for Imbalanced Data Classification. *Intelligent Automation and Soft Computing*, *35*(1).

Alnuaimi, A. F. A. H., & Albaldawi, ., T. H. K. (2024). An overview of machine learning classification techniques. *BIO Web of Conferences*, *97*.

Alshehri, M. S., Ahmad, J., Almakdi, S., Qathrady, M. A., Ghadi, Y. Y., & Buchanan, ., W. J. (2024). SkipGateNet: A Lightweight CNN-LSTM Hybrid Model with Learnable Skip Connections for Efficient Botnet Attack Detection in IoT. *IEEE Access*, *12*.

Alsubaei, F. S. (2025). Smart deep learning model for enhanced IoT intrusion detection. *Scientific Reports*, *15*(1), 20577.

Anusha, G., Baigmohammad, G., & Mageswari, U. (2024). Detection of cyber attacks on IoT based cyber physical systems. *MATEC Web of Conferences. 392*, *392*, 01166.

Bachar, A., Makhfi, E., N., Bannay, E. L., & O., (2020). Machine learning for network intrusion detection based on SVM binary classification model. *Advances in Science, Technology and Engineering Systems*, *5*(4).

Bajpai, S., Sharma, K., & Chaurasia, B. K. (2024). A hybrid meta-heuristics algorithm: Xgboost-based approach for ids in iot. *SN Computer Science*, *5*(5), 537.

Becker, E., Gupta, M., & Aryal, K. (2023). *Using Machine Learning for Detection and Classification of Cyber Attacks in Edge IoT*. Proceedings - IEEE International Conference on Edge Computing.

Behiry, M. H., & Aly, M. (2024). Cyberattack detection in wireless sensor networks using a hybrid feature reduction technique with AI and machine learning methods. *Journal of Big Data*, *11*(1).

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Choukri, W., Lamaazi, H., & Benamar, N. (2021). *Abnormal Network Traffic Detection using Deep Learning Models in IoT environment*. 2021 3rd IEEE Middle East and North Africa COMMunications Conference, MENACOMM 2021. 98–103.

Darnal, K. (2024). *A comparative analysis of machine learning algorithms in network-based intrusion detection systems for*

detecting advanced persistent threats to enhance cybersecurity.

Dash, S. K., Dash, S., Mahapatra, S., Mohanty, S. N., Khan, M. I., Medani, M., Abdullaev, S., & Gupta, M. (2024). Enhancing DDoS attack detection in IoT using PCA. *Egyptian Informatics Journal*, 100450.

Dhanya, K. A., Vajipayajula, S., Srinivasan, K., Tibrewal, A., Kumar, T. S., & Kumar, T. G.(2023). Detection of Network Attacks using Machine Learning and Deep Learning Models. *Procedia Computer Science*, *218*(57–66).

Faker, O., & Dogdu, E. (2019). *Intrusion detection using big data and deep learning techniques*. *7*(5), 86–99.

Gianchandani, S. (2023). *A Network-Based Intrusion Prevention Approach for Cloud Systems Using XGBoost and LSTM Models* [Master's Thesis]. Arizona State University.

Goparaju, B., Rao, B. S., & Bhargavi, K. (2024). An Intrusion Detection System Using Principal Component Analysis and Extreme Gradient Boosting. In *Disruptive technologies in Computing and Communication Systems* (pp. 83–91). CRC Press.

Hammood, D. A. (2024). A hybrid system based on machine learning and PSO for network intrusion detection. *AIP Conference Proceedings*, *3232*(1), 020041.

Hooshmand, D. A. M. K. (2019). Machine learning based network anomaly detection. *Int. J. Recent Technol. Eng*, *8*(4).

Irhebhude, M. E., Musa, Z. O., & Kolawole, A. O. (2022). Uncertainties Classification in Cyberspace Using Ensemble Learning Model. *Science World Journal*, *17*(1).

Kasongo, S. M. (2023). A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework. *Computer Communications*, *199*, 113–125.

Kasongo, S. M., & Sun, Y. (2020). Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset. *Journal of Big Data*, *7*(1), 105.

Khaw, Y. M., Jahromi, A., A., A., M, M. F., Sanner, S., Kundur, D., & Kassouf, M. (2021). A Deep Learning-Based Cyberattack Detection System for Transmission Protective Relays. *IEEE Transactions on Smart Grid*, *12*(3), 2554–2565.

Kolawole, A. O., Irhebhude, M. E., & Odion, P. O. (2025). Human Action Recognition in Military Obstacle Crossing Using HOG and Region-Based Descriptors. *Journal of Computing Theories and Applications*, *2*(3), 410-426. https://doi.org/10.62411/jcta.12195

Kumar, A. (2025). *Network Intrusion Detection using Advanced Machine Learning with Data Engineering* [PhD Thesis].

Malik, M., Ghous, H., Mubeen, M., Munir, A. M., & Ahmad, N. (2024). Intelligent intrusion detection system for internet of things using machine learning techniques. *International Journal of Information Systems and Computer Technologies*, *3*(1), 23–39.

Manjaragi, S. V., Asrani, D., Kumar, A., Sandesh, R., Maindola, M., & Sai, A. (2024). Leveraging XGBoost based GBM for Proactive Detection of Man-in-the-Middle Cyber Attacks. *2024 Asian Conference on Intelligent Technologies (ACOIT)*, 1–6.

Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). *2015 Military Communications and Information Systems Conference (MilCIS)*, 1–6.

More, S., Idrissi, M., Mahmoud, H., & Asyhari, A. T. (2024). Enhanced intrusion detection systems performance with UNSW-NB15 data analysis. *Algorithms*, *17*(2), 64.

Pansari, N., Srivastava, S., Agarwal, M., & others. (2024). Attack classification using machine learning on unsw-nb 15 dataset using xgboost feature selection & ablation analysis. *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, 1–9.

Parambil, E., & Krishna, A. (2025). *A Hybrid Ensemble Model using XGBoost and AdaBoost to detect and distinguish zero-day attacks* [PhD Thesis]. Dublin, National College of Ireland.

Pareriya, R., Verma, P., & Suhana, P. (2023). An ensemble Xgboost approach for the detection of cyber-attacks in the Industrial IoT domain. In *Big Data Analytics in Fog-Enabled IoT Networks* (pp. 125–140). CRC Press.

Salehpour, A., Norouzi, M., Balafar, M. A., & SamadZamini, K. (2024). A cloud-based hybrid intrusion detection framework using XGBoost and ADASYN-Augmented random forest for IoMT. *IET Communications*, *18*(19), 1371–1390.

Sarhan, M., Layeghy, S., Moustafa, N., & Portmann, M. (2020). *NetFlow Datasets for Machine Learning-Based Network Intrusion Detection Systems*. Big Data Technologies and Applications: 10th EAI International Conference, BDTA 2020, and 13th EAI International Conference on Wireless Internet, WiCON 2020, Virtual Event, December 11.

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, *2*(6).

Shahid, O., Mothukuri, V., Pouriyeh, S., Parizi, R. M., & Shahriar, H. (2021). *Detecting network attacks using federated learning for iot devices*. 2021 IEEE 29th International Conference on Network Protocols (ICNP). 1–6.

Sugin, S., & Kanchana, M. (2024). Enhancing intrusion detection with imbalanced data classification and feature selection in machine learning algorithms. *International Journal of Advanced Technology and Engineering Exploration*, *11*(112), 405.

Tariq, A., Elhadef, M., & Ghani Khan, M. U. (2023). Intelligent Intrusion Detection System for Iot Enabled It-Ot Devices. *Available at SSRN 4597142*.

Xiao, Y., Xing, C., Zhang, T., & Zhao, Z. (2019). An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks. *IEEE Access*, *7*, 42210–42219. https://doi.org/10.1109/ACCESS.2019.2904620

Yin, Y., Jang-Jaccard, J., Xu, W., Singh, A., Zhu, J., Sabrina, F., & Kwak, J. (2023). IGRF-RFE: a hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset. *Journal of Big Data. 10(1)*, *10*(1), 1–21.

Zhang, Y., Gandhi, Y., Li, Z., & Xiao, Z. (2022). Improving the classification effectiveness of network intrusion detection using ensemble machine learning techniques and deep neural networks. *2022 International Conference on Intelligent Data Science Technologies and Applications (IDSTA)*, 117–123.