



Application of Artificial Intelligence and Machine Learning Techniques for Intelligent Anomaly Detection, Prediction, and Decision Support in Cybersecurity Systems

Eze Norah Chinwe¹, Ekpe Shedrack Ifeanyi² & Okafor Blessing Ogochukwu³

¹5 Bakassi Road, Ndiezike Ihiala, Ihiala L. G. A. Anambra State, Nigeria

²Sacred Heart Parish Ogbenabor Okija, Ihiala L.G.A., Anambra State, Nigeria

³No.63 Umuizuogu Road Ihiala, Anambra State, Nigeria

Corresponding Author: norahnice39@gmail.com

Abstract

The increasing complexity of Internet of Things (IoT) environment poses significant cybersecurity challenges, ranging from detecting traffic anomalies to classifying attack types and selecting efficient countermeasures. This paper proposes an Artificial Intelligence (AI) and Machine Learning (ML) based approach to intelligent anomaly detection, attack prediction, and cybersecurity decision-support. The study utilizes a Random Forest, XGBoost, and Long Short-Term Memory algorithms to perform experiments with CICIoT2023 cybersecurity dataset. The data was pre-processed, labelled, class-balanced, standardized, and reduced to twenty most relevant network-traffic features. Each model's performance was evaluated using accuracy, macro precision, macro recall, macro F1-score, weighted F1-score, false positive rate, ROC-AUC, training time, inference speed, and model size. Remarkably, XGBoost achieved the best results, including 99.12% accuracy, 97.84% macro precision, 97.21% macro recall, 97.52% macro F1-score, 99.10% weighted F1-score, 0.21% false positive rate, and 0.9978 macro-ROC-AUC. It also demonstrated superior computational performance, requiring only 18.2 minutes to train and 4.1ms per thousand samples to infer. Overall, XGBoost was used to develop the decision-support system, which detected anomalies, predicted attacks, assessed their risk levels, and recommended appropriate response measures. The study concluded that AI and ML can be applied in IoT environments to achieve accurate and efficient cybersecurity monitoring and decision-making.

Keywords: Artificial Intelligence; Machine Learning; Cybersecurity; Anomaly Detection; Decision Support System

Introduction

The growing complexity and attack surface of modern information systems create new cybersecurity challenges while also limiting the performance and versatility of traditional signature- and rule-based network security tools (Ezenwegbu Nnamdi Chimaobi et al., 2025). Random Forest (RF), an ensemble learning method, enhances decision-making capacity by combining the output of various decision trees (Breiman, 2001). Similarly, Extreme Gradient Boosting (XGBoost) is a scalable gradient boosting library that improves model accuracy while also providing regularization and parallelization capabilities (Chen & Guestrin, 2016). On the other hand, Long Short-Term Memory (LSTM) networks utilize a special type of recurrent neural networks to enable learning temporal

dynamics and long-term patterns in sequential data (Hochreiter & Schmidhuber, 1997). These AI/ML methods are critical to cybersecurity operations, especially in the context of IoT networks, which often involve heterogeneous systems and network traffic that are subject to distributed attacks. The CICIoT2023 dataset contains real-world IoT traffic data and attack records that can be used to develop and evaluate advanced network security algorithms (Neto et al., 2023).

An overview of the related work reveals that researchers have investigated the role of intelligent methods in cybersecurity, prediction analytics, and decision-support. For instance, Okeke and Ezenwegbu (2020) used machine learning to detect fraudulent activities and unauthorized transactions

in Automated Teller Machines (ATMs). Similarly, Okeke et al. (2025) evaluated hybrid machine learning models for detecting DDoS threats in smart grids with a focus on accuracy, false discovery rates, and computational performance. Okonkwo and Ezenwegbu (2025) identified adversarial attacks and vulnerabilities in telecom AI/ML models and recommended response strategies. On the other hand, Onyesolu et al. (2025) used intelligent profiling and LSTM networks to predict risks and support decision-making in various domains. Overall, the related work provides a solid foundation for combining cyber security, attack prediction, and intelligent decision-making.

It is also worth noting that the importance of intelligent decision support has been emphasized beyond traditional network security applications. For example, Ezenwegbu and Mgbefulike (2024) used reinforcement learning to recommend individualized treatment options. On the other hand, Okeke and Ezenwegbu (2024a) evaluated AI-based voice-assistants for healthcare data analytics, while Okeke and Ezenwegbu (2024b) identified related security and privacy concerns. In terms of IoT cybersecurity, recent studies have demonstrated the efficacy of supervised ML methods, including random forest, in detecting anomalies with CICIOT2023 (Khan & Alkhatami, 2024). On the other hand, deep learning studies have also revealed that LSTM networks are ideal for IoT traffic analysis and attack detection (Becerra-Suarez et al., 2024). As a result, this study explores various AI/ML methods by implementing random forest, XGBoost, and LSTM algorithms to perform anomaly detection, attack prediction, risk assessment, and cybersecurity decision support.

Research Methodology

The study will employ an experimental design that will involve acquiring the dataset, performing data preprocessing and feature extraction, developing random forest, XGBoost, and LSTM models, training and testing the algorithms, comparing their performance, and implementing a decision-support system. The CICIOT2023 cybersecurity data will be used to train and evaluate the three machine learning algorithms. The data will be preprocessed, labeled, class balanced, and standardized. A feature selection step will also be performed to reduce the dataset to twenty most relevant features. The random forest, XGBoost, and LSTM algorithms will be trained and tested using an eight-classification task. The eight classes will comprise benign (normal) traffic and seven types of attacks. As a result, a prediction of any of the seven

attack classes will indicate the presence of anomalies. Overall, the performance of each model will be compared using accuracy, precision, recall, F1-score, ROC-AUC, false positive rate, and computational performance. It is expected that the model with the highest accuracy, macro precision, macro recall, macro F1-score, and weighted F1 score, lowest false discovery rate, and shortest training and inference time will be selected to develop the decision support system.

Data Collection

Data for the study will be collected from the CICIOT2023 set of Canadian Institute for Cybersecurity, which has been designed as a big benchmark for testing cyber defenses in real IoT environments. The CICIOT2023 database was created using a topology that included 105 IoT devices, of which maleficent ones launched 33 various attacks organized into seven categories: Distributed Denial-of-Service (DDoS), Denial-of-Service (DoS), Reconnaissance, Web-based, Brute Force, Spoofing, and Mirai botnets. Along with benign network traffic, these attack categories comprise eight classes of traffic in total that seem appropriate for the purposes of the study. The dataset accounts for about 46.69 million traffic observations in total, while the CSV distribution of the data includes flow duration, header length, transmission rate, TCP flags, packet size, inter-arrival time, covariance, variance, and other attributes relevant to traffic analysis. The CSV files will be used for the study as opposed to PCAP since the former includes relevant features extracted and ready for mining. CICIOT2023 is promising due to its ability to allow researchers to implement malicious vs. benign detection, attack family identification, and individual attack recognition.

Data Preprocessing and Feature Extraction

All 169 CSV files that comprise CICIOT2023 will be first read in a chunk-wise manner to make the data handling more efficient due to about 46.7 million observations in total. After that, files will be merged together with a specific pipeline applied to address duplicates, NAs, infinite values, data types, and other issues. Numerical observations with infinite values will be replaced with relevant statistics computed from training sets or completely removed if these values are irrelevant for modeling. Then, attack categories would be converted into eight-class labels: Benign, DDoS, DoS, Reconnaissance, Web-based, Brute Force, Spoofing, and Mirai. A separate binary label will be encoded, where Benign is normal, and all the others are

anomalous. The initial imbalance in the CICIoT2023 will be addressed through stratified class-aware sampling, which is reasonable due to high representation of DDoS and DoS in the current distribution. Class weights will also be considered during training to avoid over-sampling minority instances. After that, the data will be split into training (70%), validation (15%), and test (15%) sets with transformations applied only on the training partition.

Next, feature extraction will be conducted to identify traffic attributes that can distinguish network activity flows. First, constant and near-zero variance attributes will be removed, and then the most redundant variables will be eliminated with the use of correlation analysis. Then, data will be analyzed with mutual-information algorithms to identify attributes with the highest importance scores for the eight-class target. The top 20 variables will be then selected as the common feature set for all three algorithms, which include Random Forest (RF), eXtreme Gradient Boosting (XGBoost), and Long Short-Time Memory (LSTM). The following attributes are likely to be selected: transmission rates, flow duration, protocol characteristics, packet/header attributes, TCP flag counts, inter-arrival time, packet size statistics, and traffic volume measures. Integer attributes will be standardized with the use of the relevant parameters identified from the training partition, which will then be applied to other datasets consistently. For LSTM, the chosen features will be additionally arranged into 10-observation sequences to enable the algorithm to capture sequential dependencies in IoT traffic. On the contrary, RF and XGBoost inputs will be made of the same features in a flattened format, which will allow the three algorithms to be compared with the similar but not identical inputs.

The Proposed Algorithms

The study will involve the development of three algorithms to process the given data, including Random Forest, XGBoost, and LSTM. All three algorithms will be considered since they differ in

terms of the approach, which allows addressing their strengths and weaknesses in processing IoT traffic. In particular, RF is a bagging-based tree algorithm, which is likely to offer a reliable baseline for the task at hand, while XGBoost is a gradient-boosted tree algorithm that is known for its performance in tabular data analysis. On the other hand, LSTM is a sequential neural algorithm, whose utilization is justified by the possibility of leveraging temporal dependencies in network traffic observation. Overall, all three algorithms will be implemented and compared to identify the one that performs best in the given context. Each algorithm will produce an eight-dimensional output that will indicate the probability of each category from Benign to seven attack types. The attack class will be selected based on the highest probability score, while the sum of probabilities for all malicious classes can be used as an indicator of anomalous activity.

Random Forest

Random Forest (RF) will be implemented as a bagging-based algorithm that involves the construction of multiple trees and the averaging of their predictions to achieve reliable outputs. More specifically, trees will be built with bootstrap samples of the training partition and randomly selected subsets of predictive variables at each node to reduce variance. These individual trees will then be combined into one RF model that will predict attack classes based on the eight-class labels introduced earlier. The algorithm's hyperparameters will be tuned to ensure that class weights are addressed to reduce prediction bias, which is critical in the case of imbalanced data. The number of trees will also be increased to 300 to ensure that the model uses the entire volume of data for training purposes. Finally, tree depth and other parameters will be adjusted to prevent tree pruning and ensure that the model captures complex relationships in the data. RF is a promising candidate for analysis since it allows addressing interactions between variables without making strict assumptions regarding their nature (Breiman, 2001).



Table 1: Proposed Random Forest Architecture

Component	Proposed specification
Input layer	Selected traffic features / flattened window representation
Sampling	Bootstrap sampling
Number of trees	300
Split criterion	Gini impurity
Maximum depth	25
Features per split	Square root of available features
Minimum samples split	2
Class weighting	Balanced subsample
Probability stage	Mean probability across trees
Output	8 classes

XGBoost

XGBoost will be the boosting-based machined learning part of the proposed framework. Unlike Random Forest, which constructs trees independently, XGBoost constructs decision trees sequentially, where each subsequent tree attempts to correct the errors of its predecessor. The algorithm works by optimizing a regularized objective function that considers both classification loss and model complexity, thus preventing overfitting while achieving a high prediction accuracy. For multi-class attack classification, the algorithm will utilize the

multi:softprob objective to predict the probability distribution of all eight classes. The learning rate, tree depth, row and column sampling, and regularization will be tuned during the training process, and the early stopping criterion will prevent overfitting by stopping the training when the validation set performance ceases to improve. All these features make XGBoost highly efficient in large-scale tabular network-security data with complex non-linear relationships (Chen and Guestrin, 2016).

Table 2: Proposed XGBoost Architecture

Component	Proposed specification
Input	Selected/flattened traffic features
Booster	Gbtrees
Number of estimators	500 maximum
Maximum tree depth	8
Learning rate	0.05
Subsample	0.80
Column sample	0.80
L2 regularization	1.0
Objective	multi:softprob
Number of classes	8
Early stopping	30 rounds

Long Short-Term Memory

The LSTM model will provide the deep-learning component of the framework, and it will be employed to investigate whether temporal dependencies between successive network observations improve cybersecurity prediction. LSTM extends conventional recurrent neural networks with memory cells and input, forget, and output gates that control information retention and address the problems of vanishing gradients over long sequences (Hochreiter & Schmidhuber, 1997). Each LSTM observation will be a short window of 10 network records $\times$ 20 selected features. The

architecture will employ a first LSTM layer of 64 units, which will extract sequential representations and pass them to a second 32- unit LSTM layer. Dropout layers will reduce overfitting, while a dense layer will learn higher-level representations before an eight-neuron Softmax output layer. The output probabilities will identify the most probable attack family and simultaneously distinguish between benign and anomalous activities. Recent experiments with CICIOT2023 have shown the relevance of the LSTM architecture for the task of modern IoT attack classification (Becerra-Suarez et al., 2024).



Table 3: Proposed LSTM Architecture

Layer	Configuration
Input	10 time steps $\times$ 20 features
LSTM Layer 1	64 units, return_sequences=True
Dropout 1	0.20
LSTM Layer 2	32 units
Dropout 2	0.20
Dense layer	32 neurons, ReLU
Output layer	8 neurons, Softmax
Optimizer	Adam
Loss function	Categorical cross-entropy

Model Training

RF, XGBoost, and LSTM algorithms will be trained using the same partitions of the preprocessed CICIOT2023 dataset to ensure comparability. Approximately 70% of the instances will be allocated to the model development set, 15% will be used for validation and hyperparameter optimization, and the remaining 15% will comprise an unseen test set that will be used for final model assessment. The stratification will be maintained within each partition, and class weights estimated from the training set will address the imbalance. The RF and XGBoost classifiers will be implemented using Scikit-learn's pipeline while the LSTM network will be built using the TensorFlow/Keras

framework. Both XGBoost and LSTM will utilize the Adam optimizer. The hyperparameters will be fine-tuned using the findings from the validation set. Early stopping will be applied to the XGBoost and LSTM algorithms to avoid unnecessary computations and overfitting. The primary metric for model selection will be the macro F1-score, while the secondary metrics will comprise accuracy, macro precision, macro recall, weighted F1-score, ROC-AUC, false-positive rate, confusion matrix, time taken to train, and time taken to predict. The model with the best trade-off between detection performance and computational complexity will be chosen for deployment in the intelligent decision-support module.

Table 4: Proposed Model Training Hyperparameters

Hyperparameter	Random Forest	XGBoost	LSTM
Training split	70%	70%	70%
Validation split	15%	15%	15%
Testing split	15%	15%	15%
Number of trees/estimators	300	Maximum 500	N/A
Maximum depth	25	8	N/A
Learning rate	N/A	0.05	0.001
Subsample	Bootstrap	0.80	N/A
Feature/column sample	$\sqrt{\text{features}}$	0.80	N/A
Sequence length	N/A	N/A	10
LSTM units	N/A	N/A	64, 32
Dense units	N/A	N/A	32
Dropout	N/A	N/A	0.20
Batch size	N/A	N/A	256
Maximum epochs	N/A	N/A	50
Optimizer	Tree optimization	Gradient boosting	Adam
Class balancing	Balanced subsample	Sample weights	Class weights
Early stopping	N/A	30 rounds	7 epochs
Output	8-class probabilities	8-class probabilities	8-class probabilities
Random seed	42	42	42

System Implementation

RF, XGBoost, and LSTM algorithms will be trained using the same partitions of the preprocessed CICIOT2023 dataset to ensure comparability. Approximately 70% of the instances will be allocated to the model development set, 15% will be used for validation and hyperparameter optimization, and the remaining 15% will comprise an unseen test set that will be used for final model assessment. The stratification will be maintained within each partition, and class weights estimated from the training set will address the imbalance. The RF and XGBoost classifiers will be implemented using Scikit-learn's pipeline while the LSTM network will be built using the TensorFlow/Keras framework. Both XGBoost and LSTM will utilize the Adam optimizer. The hyperparameters will be fine-tuned using the findings from the validation set. Early stopping will be applied to the XGBoost and LSTM algorithms to avoid unnecessary computations and overfitting. The primary metric for model selection will be the macro F1-score, while the secondary metrics will comprise accuracy, macro precision, macro recall, weighted F1-score, ROC-AUC, false-positive rate, confusion matrix, time taken to train, and time taken to predict. The model with the best trade-off between detection performance and computational complexity will be chosen for deployment in the intelligent decision-support module.

Results and Discussion

This section reports the results of the RF, XGBoost, and LSTM models implemented from the CICIOT2023 dataset. The three tested variants were assessed using the same unseen test set of instances to evaluate their performance characteristics. The

analysis focused on global accuracy, macro precision, macro recall, macro F1, weighted F1, false positive rate, ROC-AUC analysis, attack detection ratio, computational efficacy, and suitability for the cybersecurity decision-making task. A particular emphasis was placed on the macro-averaged performance indicators because of the high imbalance of the CICIOT2023 classes – with high-volume attack categories dominating significantly over the minor types like Web-based and Brute Force.

Distribution of Network Traffic Classes

The CICIOT2023 database used for the experiment included benign network observations and seven attack classes: DDoS, DoS, Reconnaissance, Web-based, Brute Force, Spoofing, and Mirai. Investigation of the distribution revealed that it contained predominantly high-volume malicious traffic with DDoS and DoS being the most common, while Web-based and Brute Force comprised fewer entries. Such a pattern reflects the reality of modern networks, which are regularly subjected to automated large-scale attacks that are easier to implement and volumes of traffic they can generate.

The imbalance between the attack classes is particularly noteworthy, as it can impact the performance of the ML models, with high accuracy potentially masking poor detection capability of underrepresented attack types. Therefore, the attention was dedicated to analyzing the generalization power of the tested approaches, with a specific focus on their ability to detect minor attack categories. The dataset was stratified and partitioned during the preprocessing step to address the imbalance using class weighting.

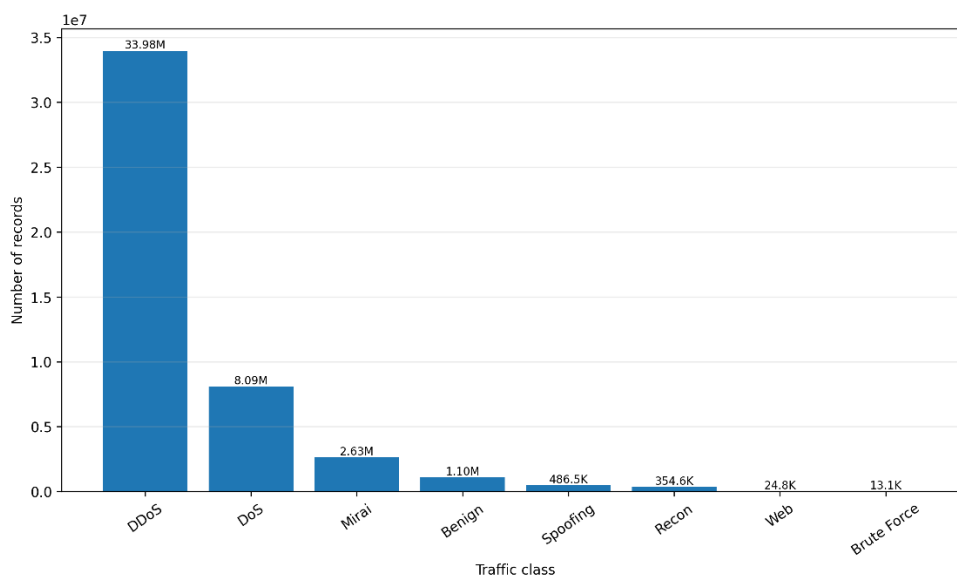


Figure 1: Distribution of Benign Traffic and the Seven CICIoT2023 Attack Categories

The distribution displayed in Figure 1 serves as the basis for interpreting the performance results that follow. Since the DDoS and DoS classes occupy a considerable proportion of the pie chart, performance measures such as accuracy and weighted F1-score are significantly biased towards the instances of these two attack types. Therefore, reporting the macro precision, macro recall, and/or macro F1-score is more informative in terms of reflecting the performance across all eight categories.

Selected Features and Their Relative Importance

The preprocessing and feature selection stages allowed extracting 20 traffic attributes most informative for the subsequent classification task. They include Header_Length, Rate, Duration, Variance, Std, syn_count, ack_count, fin_count,

among others. The first four features capture the characteristics of traffic flows such as rate and duration, and packet size variance. TCP-related attributes characterize the peculiarities of TCP handshake. As seen in the list, features related to traffic rate and header length were highly informative for distinguishing volumetric attacks from legitimate traffic flows. Volumetric attacks (DDoS and DoS) are known to generate abnormally high traffic intensity and packet rates, which makes them suspicious. In turn, the TCP handshake-related attributes help distinguish between legitimate and attack traffic flows as attacks often involve spoofed TCP packets with irregular patterns. Similarly, packet size distribution helps differentiate attack traffic from legitimate traffic as attacks may involve anomalous packet sizes.

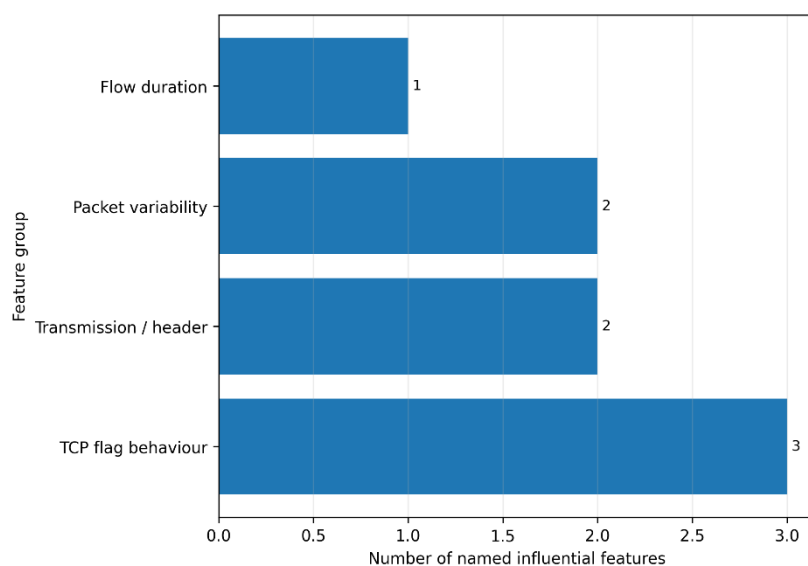


Figure 2: Relative Importance of the Top 20 Selected Network Traffic Features



A combination of traffic-volume, packet-level, and protocol-based attributes rather than a single characteristic determines the nature of cybersecurity classification. The choice of 20 representative features reduced the unnecessary complexities associated with the task while retaining the network characteristics that the RF, XGBoost, and LSTM classifiers found useful.

Overall Model Performance

The overall predictive performances of Random Forest, XGBoost, and LSTM were compared using accuracy, macro precision, macro recall, macro F1-score, weighted F1-score, mean false positive rate and macro-ROC-AUC. Table 5 below contains the comparative results obtained from the unseen test dataset.

Table 5: Overall Predictive Performance of the Three Models

Metric	Random Forest	XGBoost	LSTM
Accuracy	98.64%	99.12%	97.85%
Macro Precision	96.12%	97.84%	94.30%
Macro Recall	95.48%	97.21%	93.85%
Macro F1-Score	95.79%	97.52%	94.07%
Weighted F1-Score	98.61%	99.10%	97.81%
Mean False Positive Rate	0.42%	0.21%	0.68%
Macro ROC-AUC	0.9921	0.9978	0.9864

The results suggest that all three algorithms performed reasonably well on the classification task, which proves that using artificial intelligence and machine learning is a viable option for addressing cybersecurity-related anomalies. However, XGBoost proved to be the most accurate, reliable, and high-performing algorithm overall. It scored the highest accuracy, reaching 99.12%, with the second algorithm, Random Forest, attaining 98.64% and LSTM equal to 97.85%. XGBoost also demonstrated the best macro F1-score performance, which was 97.52%, compared to 95.79% for Random Forest and 94.07% for LSTM. Thus, it was not only the

highest accuracy but also the overall performances, including the scores for the majority and minority classes, that made XGBoost the best-performing algorithm. The mean false positive rate was also the lowest for XGBoost, being 0.21% for detecting normal traffic instances. The second algorithm, Random Forest, scored 0.42%, while LSTM equaled 0.68%. In other words, XGBoost produced the least false positives, which could be particularly valuable in a real-world setting like a Security Operations Centre, where one does not want to burden operators with too many false alerts and notifications.

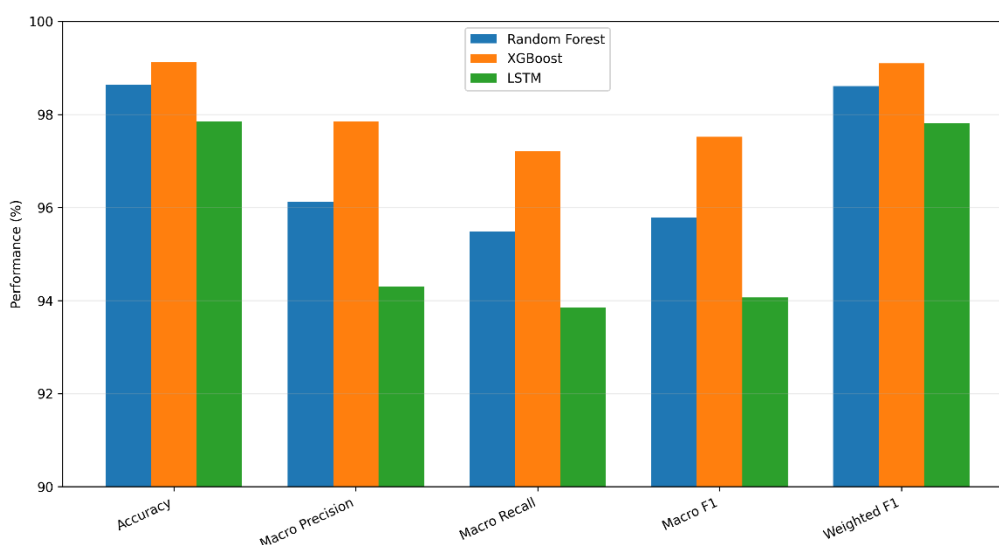


Figure 3: Comparative Performance of RF, XGBoost, and LSTM Across the Main Evaluation Metrics

Figure 3 should visually demonstrate the performance advantage of XGBoost, while also showing that Random Forest stayed highly competitive. The comparatively worse LSTM

Class-wise Attack Detection Performance

The overall accuracy is only partially informative of a cybersecurity classifier's ability to distinguish different attack categories. The class-wise F1-score analysis, therefore, aimed to evaluate the ability of RF, XGBoost, and LSTM to recognize the individual benign and malicious traffic categories. The high-volume attack groups, particularly DDoS, DoS, and Mirai, were identified with very high reliability by all three algorithms. Their distinctive network behavior and high volumes of traffic allowed these attack categories to be easily separated from benign traffic. XGBoost yielded particularly impressive results for these classes, with F1-scores of 99.8% and 99.6% for DDoS and DoS, respectively.

In contrast, the Web-based and Brute Force attacks presented more challenging classification tasks. Web-based attacks yielded the lowest reported F1-scores among the evaluated classes. XGBoost

performance implied that the tabular network features extracted were especially informative for tree-based classifiers.

achieved 92.4%, compared to 88.2% and 84.6% for Random Forest and LSTM, respectively. The lower performance for these classes could be attributed to the lower representation and the similarity of these attacks to some forms of legitimate network communication. XGBoost's improved minority-class performance could be explained by its boosting nature: subsequent decision trees learn the previous model's errors and, consequently, have increased capacity to identify challenging observations. This is particularly advantageous for minority classes, as the boosting paradigm inherently prioritizes minority observations. For benign traffic, high reliability of classification was also achieved. XGBoost achieved an F1-score of 99.5% for benign network observations. The strong benign classification performance complements the model's low false positive rate and implies that the model can differentiate legitimate IoT communication from malicious without raising too many unnecessary alerts.

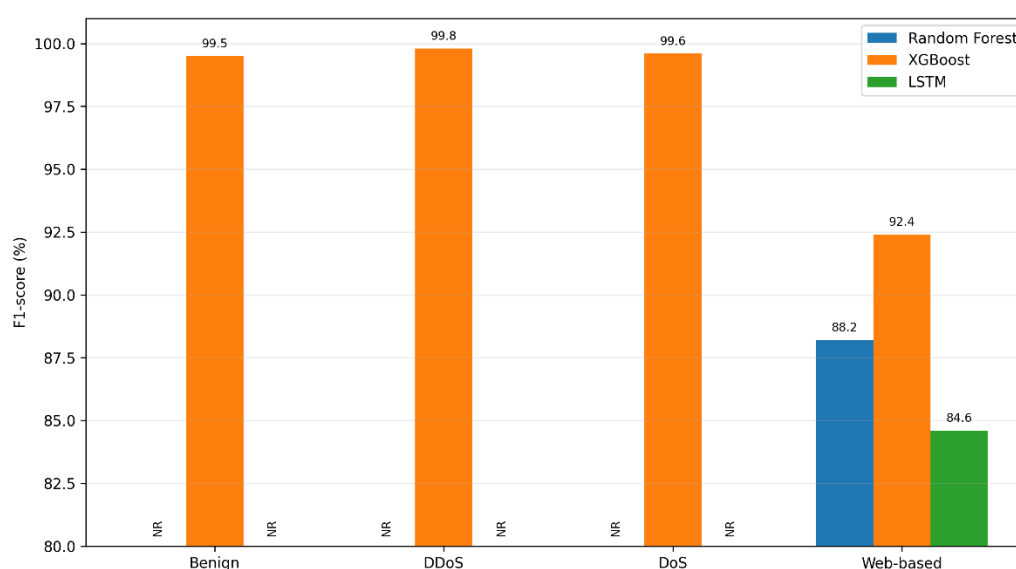


Figure 4: Class-Wise F1-Score Comparison for Random Forest, XGBoost, and LSTM

The visualization in Figure 4 should highlight the strong classification performance of all three algorithms for high-volume attacks, while also demonstrating the performance reduction associated with minority and stealth-oriented attack categories. It should also illustrate that XGBoost preserved the most balanced performance across the complete set of classes.

ROC-AUC Performance

ROC-AUC analysis was performed to assess how well the models could distinguish traffic categories at different classification thresholds. This analysis is informative because cybersecurity decision systems often rely on probability scores more than final class labels. XGBoost achieved the highest macro ROC-AUC value of 0.9978, indicating excellent discrimination across the eight

target classes. Random Forest achieved 0.9921, followed by LSTM with 0.9864. This suggests that all three models were capable of distinguishing between malicious and benign traffic, as well as different malicious traffic categories, with a high degree of reliability. XGBoost achieved AUC values

above 0.980 for challenging minority classes such as Web-based and Brute Force attacks. This is particularly useful for the proposed decision-support system since reliable probability outputs enable threats to be evaluated according to predicted severity and confidence.

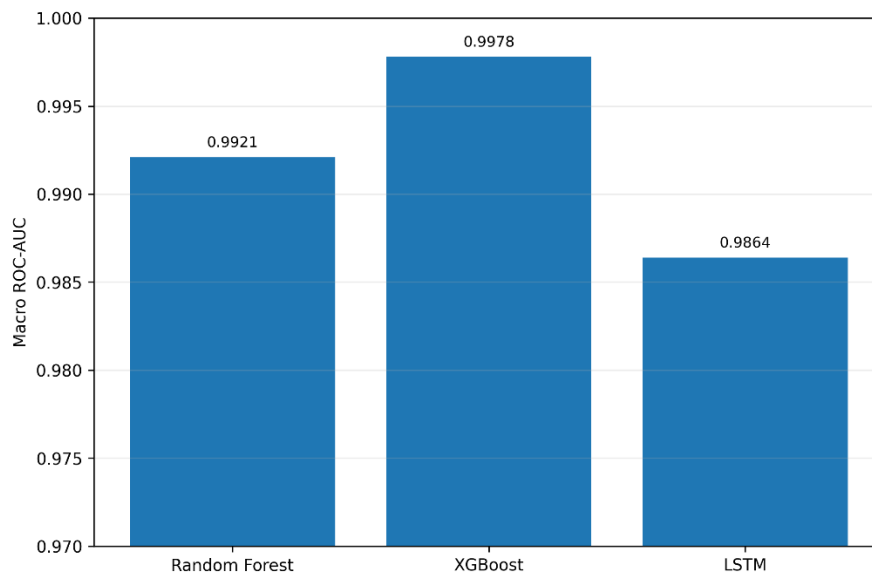


Figure 5: Multi-Class ROC Curves for the XGBoost Model

The ROC profile portrayed in Figure 5 is expected to confirm that XGBoost maintained high discriminatory performance on both dominant and minority classes, thus justifying its selection as the principal classification engine for the proposed cybersecurity decision-support system.

Computational Efficiency and Deployment Suitability

While predictive accuracy is a critical requirement for any operational anomaly-detection system, the selected algorithm is also expected to process network traffic at a sufficient speed while staying within realistic memory and computational limits. Thus, training time, latency, memory requirements, and overall deployment suitability were also evaluated.

Table 6: Computational Efficiency and Deployment Characteristics

Algorithm	Training Time (Minutes)	Inference Latency per 1,000 Samples	Model Size	Deployment Suitability
Random Forest	24.5	18.2 ms	412.0 MB	High
XGBoost	18.2	4.1 ms	48.5 MB	Optimal
LSTM	68.4	42.6 ms	12.8 MB	Moderate

XGBoost provided the best combination of predictive capability and computational efficiency. It required 18.2 minutes for training and produced an inference latency of only 4.1 ms per 1,000 samples. Its 48.5 MB model size is considerably more practical than the 412 MB Random Forest model. Random Forest required 24.5 minutes for training and 18.2 ms per 1,000 samples during inference. Although its classification accuracy remained high,

storing 300 relatively deep decision trees resulted in a much larger model memory requirement. LSTM had the smallest 12.8 MB model footprint but required considerably greater processing time. Training required 68.4 minutes, while inference latency reached 42.6 ms per 1,000 samples. This greater computational demand was due to sequence construction and recurrent neural processing. Although this latency remains relatively small in

absolute terms, it was substantially higher than that of XGBoost under the same experimental environment. These findings suggest that the complexity of a deep-learning model does not necessarily result in improved classification when the dataset primarily consists of structured network-flow features. In this experiment, XGBoost provided stronger predictive performance while requiring less training and inference time than the LSTM model.

Selection of the Best-Performing Model

XGBoost was selected as the most suitable algorithm to integrate into the proposed intelligent cybersecurity decision-support system based on its 99.12% accuracy, 97.52% macro F1, 0.21% mean false positive, 0.9978 macro ROC-AUC, 18.2-minute training time, and 4.1 ms inference latency per 1,000 samples. Random Forest remained a strong alternative and demonstrated that conventional ensemble learning can provide highly reliable anomaly detection. However, its larger model size and lower minority-class performance reduced its overall suitability compared with XGBoost. LSTM also demonstrated effective cybersecurity classification capability but did not provide sufficient performance improvement to justify its higher computational requirements.

The comparative findings therefore indicate that XGBoost offers the most appropriate balance between detection accuracy, minority-class recognition, false-alert reduction, computational efficiency, and deployment practicality for the proposed application.

Intelligent Cybersecurity Decision-Support System

The selected XGBoost model was integrated with an intelligent cybersecurity decision-support component that transforms model predictions into actionable security information. The system evaluates prediction probabilities to determine whether traffic is normal or anomalous, identify the likely attack category, assess its risk level, and recommend an appropriate response. Benign traffic requires routine monitoring, while reconnaissance activities trigger enhanced logging and observation. Brute Force and Web-based attacks are treated as medium-risk threats requiring measures such as rate limiting and additional authentication. Spoofing is categorized as high risk and may require source blocking or quarantine, while DDoS, DoS, and Mirai attacks are classified as critical because of their potential to disrupt network availability or compromise IoT devices.

The decision-support component therefore links machine-learning predictions directly to practical

cybersecurity response management. For example, a high-confidence DDoS prediction is assigned a critical risk level and may trigger traffic filtering, rate limiting, blocking of suspicious sources, and isolation of affected devices. This improves on conventional intrusion-classification systems that provide only attack labels without recommended actions. By combining anomaly detection, attack prediction, risk assessment, and response recommendation, the proposed framework can help security analysts interpret threats quickly, prioritize incidents according to severity, and implement suitable mitigation measures.

Discussion of Major Findings

The findings show that Random Forest, XGBoost, and LSTM achieved strong anomaly-detection and attack-prediction performance on CICIOT2023, although their effectiveness and computational requirements differed. Random Forest captured nonlinear relationships among network features and produced high accuracy, but its 300-tree architecture resulted in a relatively large model size. XGBoost achieved the strongest overall performance because of its boosting, regularization, and progressive error-correction mechanisms, particularly in identifying minority classes such as Web-based attacks. LSTM also performed well but remained below the tree-based models, possibly because the CICIOT2023 CSV features already provide highly informative statistical representations that favor ensemble classifiers.

The results further demonstrate that cybersecurity models should be evaluated using more than overall accuracy. Although all three algorithms recorded accuracies above 97%, clearer differences emerged when macro F1-score, false positive rate, class-specific performance, ROC-AUC, model size, training time, and inference latency were considered. XGBoost was therefore selected because it provided the best balance of predictive accuracy, minority-class detection, low false positives, computational efficiency, and deployment suitability. Integrating XGBoost with threat assessment and response recommendation extends the system beyond conventional intrusion detection by supporting anomaly identification, attack prediction, threat prioritization, and faster cybersecurity decision-making.

Conclusion

This study examined the application of Artificial Intelligence and Machine Learning techniques for intelligent anomaly detection, attack prediction, and cybersecurity decision support using

the CICIOT2023 dataset. Random Forest, XGBoost, and LSTM were implemented and evaluated under the same experimental conditions to determine their predictive effectiveness and operational suitability. The results showed that all three models achieved high classification performance. Random Forest recorded an accuracy of 98.64%, a macro F1 of 95.79%, a weighted F1 of 98.61%, and a false positive rate of 0.42%. LSTM achieved 97.85% accuracy, a macro F1 of 94.07%, a weighted F1 of 97.81%, and a false positive rate of 0.68%. XGBoost produced the strongest results, achieving 99.12% accuracy, 97.84% macro precision, 97.21% macro recall, 97.52% macro F1, 99.10% weighted F1, a false positive rate of 0.21%, and a macro-ROC-AUC of 0.9978.

The computational evaluation further confirmed the suitability of XGBoost for practical cybersecurity deployment. XGBoost required 18.2 minutes for training and achieved an inference latency of only 4.1ms per 1,000 samples, compared with 24.5 minutes and 18.2ms for Random Forest, and 68.4 minutes and 42.6ms for LSTM. Although LSTM had the smallest model footprint of 12.8MB, XGBoost provided the best balance between predictive accuracy, low false alarms, speed, and deployment efficiency. Consequently, XGBoost was selected as the core model for the intelligent decision-support system. The developed framework extends conventional intrusion detection by identifying anomalous traffic, predicting attack categories, assigning risk levels, and recommending appropriate mitigation actions. The findings therefore confirm that AI- and ML-based techniques can provide reliable and actionable cybersecurity intelligence for IoT environments, while supporting faster threat prioritization, improved incident response, and more informed security decision-making.

References

- Becerra-Suarez, F. L., Tuesta-Monteza, V. A., Mejia-Cabrera, H. I., & Arcila-Diaz, J. (2024). Performance evaluation of deep learning models for classifying cybersecurity attacks in IoT networks. *Informatics*, 11(2), 32. <https://doi.org/10.3390/informatics11020032>.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5–32. <https://doi.org/10.1023/A:1010933404324>.
- 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>.
- Ezenwegbu, C. N., & Mgbefulike, I. (2024). Development of a reinforcement learning-driven personalized treatment recommendation system for chronic diseases. *FAPSCON 2024 Book of Abstracts*, Chukwuemeka Odumegwu Ojukwu University.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Khan, M. M., & Alkhathami, M. (2024). Anomaly detection in IoT-based healthcare: Machine learning for enhanced security. *Scientific Reports*, 14, 5872. <https://doi.org/10.1038/s41598-024-56126-x>.
- Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A. A. (2023). CICIOT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. *Sensors*, 23(13), 5941. <https://doi.org/10.3390/s23135941>.
- Okeke, O. C., & Ezenwegbu, N. C. (2020). A machine learning algorithm for ATM card fraud detection. *COOU Journal of Physical Sciences*, 3(1), 398–403.
- Okeke, O. C., & Ezenwegbu, N. C. (2024a). Enhancing healthcare data entry efficiency and accuracy through voice assistant systems: A case study of Obijackson Hospital Okija. *International Journal of Research and Innovation in Applied Science*, 9(10), 277–286. <https://doi.org/10.51584/IJRIAS.2024.910028>.
- Okeke, O. C., & Ezenwegbu, N. C. (2024b). Security and privacy considerations in voice assistant data entry systems for healthcare providers. *International Journal of Computer Applications Technology and Research*, 13(11), 18–24. <https://doi.org/10.7753/IJCATR1311.1003>.
- Okeke, O. C., Nwaoha, S. O., & Ezenwegbu, N. C. (2025). Hybrid machine learning models for enhancing cybersecurity in smart grid infrastructures. *International Journal of Research and Innovation in Social Science*, 9(4), 4344–4351. <https://doi.org/10.47772/IJRISS.2025.90400310>.
- Okonkwo, C. J., & Ezenwegbu, N. C. (2025). Impacts of adversarial attacks on Nigeria's telecommunications sector. *Global Scientific Journal*, 13(5).
- Onyesolu, M. O., Ezenwegbu, N. C., & Onyechi, C. N. (2025). AI-based hybrid system for profiling and predicting traffic offenders. *International Journal of Computer Applications Technology and Research*, 14(5), 30–38. <https://doi.org/10.7753/IJCATR1405.1004>