AJBAR Vol. 5(4), August 2026: 117-134, ISSN: 2811-2881

Arid Zone Journal of Basic and Applied Research
Faculty of Science, Kashim Ibrahim University
Maiduguri, Nigeria

Journal homepage: https://www.azjournalbar.com

Research Article

Optimized Machine Learning Framework for Cybersecurity Risk Assessment: Addressing Class Imbalance and Interpretability in Network Intrusion Detection

Nwadiuko Ahanna Chidera, Gilbert Imuetin Osaze Aimufua and Raymond Ternenge Igbudu
Center for Cyberspace Studies, Nasarawa State University Keffi
*Corresponding author's Email: ahannanwadiuko@gmail.com, doi.org/10.55639/607.020100140

ARTICLE INFO:

Keywords:
Cybersecurity Risk Assessment, Class Imbalance, LightGBM, Ensemble Learning, ML.

ABSTRACT

Cyber threats are evolving in sophistication and frequency, making traditional risk assessment approaches increasingly inadequate. Conventional intrusion detection systems often struggle with two major challenges: class imbalance and limited model interpretability, which can reduce their effectiveness in practical security operations. This study proposes an enhanced machine learning framework for cybersecurity risk assessment and evaluates it on a synthetic dataset designed to reproduce key characteristics of the CIC-IDS2017 benchmark. The framework incorporates correlation-based feature reduction, ANOVA-driven feature selection, and the Synthetic Minority Over-sampling Technique (SMOTE) to address data imbalance. Six machine learning algorithms—Logistic Regression, Random Forest, XGBoost, LightGBM, K-Nearest Neighbors (KNN), and Decision Tree—were trained and fine-tuned using RandomizedSearchCV. LightGBM achieved the highest accuracy (0.983), weighted F1-score (0.982), and macro F1-score (0.951). SHAP (SHapley Additive exPlanations) analysis was employed to identify influential network-flow features and improve model interpretability. A risk-scoring mechanism was also developed to convert classification outputs into measurable risk values for continuous monitoring. The evaluation uses a synthetic CIC-IDS2017-like dataset; therefore, the reported performance represents an experimental estimate pending validation on genuine network traffic. The results indicate that combining class-imbalance mitigation, feature selection, and ensemble learning can improve predictive performance and interpretability within the evaluated synthetic intrusion-detection setting.

Corresponding author: Nwadiuko Ahanna Chidera, **Email**: ahannanwadiuko@gmail.com
Center for Cyberspace Studies, Nasarawa State University Keffi, Nigeria

1.0 INTRODUCTION

Cybersecurity risk assessment entails the systematic identification, analysis, and evaluation of threats to an organization's information assets. Traditional frameworks, such as ISO27005 and NIST SP800-30, depend largely on expert intuition and static vulnerability repositories (Camacho et al.,

2026). These approaches are resource-intensive, inflexible, and produce qualitative outputs that lack the granularity required for real-time operational decisions. The escalating frequency of network attacks has intensified the demand for automated, data-centric risk assessment models capable of processing live network streams.

Machine learning has emerged as a powerful tool for intrusion detection, with numerous studies reporting high accuracy on standard benchmarks (Camacho et al., 2026; Dash et al., 2022). However, real-world network traffic is inherently imbalanced—benign flows vastly outnumber malicious ones, and certain attack categories (e.g., infiltration and web-based attacks) occur very rarely. This skew causes conventional ML algorithms to favor majority classes, thereby diminishing their sensitivity to minority attacks (Chowdhury, 2025). Furthermore, the black-box nature of complex models undermines trust among security analysts, who require clear explanations for generated alerts.

To tackle these challenges, this study proposes an optimized ML pipeline incorporating: (1) feature engineering through the removal of constant or highly correlated variables, coupled with ANOVA-based selection to retain only the most informative predictors; (2) class imbalance mitigation via the Synthetic Minority Over-sampling Technique (SMOTE) to balance training data; and (3) ensemble learning, evaluating six classifiers including gradient-boosting variants. The experiments use a synthetic dataset of 2,000 instances designed to reproduce key structural and class-distribution characteristics of the widely used CIC-IDS2017 repository (Islam et al., 2025). This artificial data facilitates rapid prototyping, but it cannot reproduce all variability and complexity of captured network traffic. Results on this synthetic evaluation show high classification performance, with LightGBM emerging as the top performer. SHAP analysis identified packet-length statistics and flow inter-arrival times as influential features. Additionally, a risk-scoring model generates interpretable, dynamically trackable scores for the experimental traffic data; this mechanism is presented as a proof-of-concept rather than a validated real-world risk estimator.

This study is guided by the following research questions:

RQ1: How can feature engineering and correlation-based reduction be combined with ANOVA-driven selection to retain the network-flow features most predictive of intrusion classes?

RQ2: To what extent does SMOTE-based class-imbalance mitigation improve minority-class detection across a range of ensemble and non-ensemble machine learning algorithms?

RQ3: Which of the six evaluated classifiers (Logistic Regression, Random Forest, XGBoost, LightGBM, KNN, and Decision Tree) offers the best trade-off between predictive performance and interpretability for network intrusion detection?

RQ4: How can SHAP-derived feature attributions be operationalized into a risk-scoring mechanism that supports continuous, dynamically trackable cybersecurity risk monitoring?

The paper is organized as follows: Section 2 reviews related literature, Section 3 describes the data and methodology, Section 4 presents experimental results and discussion, and Section 5 concludes with future research directions.

2. LITERATURE REVIEW

Machine learning has been extensively applied to intrusion detection. Early research relied on algorithms such as decision trees, support vector machines, and k-nearest neighbors, evaluated on legacy datasets like KDD99 and NSL-KDD (Ngueajio et al., 2022). However, these datasets are now outdated and fail to capture contemporary attack patterns. The CIC-IDS2017 dataset was subsequently introduced to overcome these limitations, offering realistic normal traffic and a diverse range of attack types. Numerous studies have since benchmarked ML models on this corpus. For instance, Rama Devi and Abualkibash (2019) achieved over 98% accuracy using random forest, while Wali and Khan (2021) demonstrated

the effectiveness of XGBoost for multi-class classification.

Class imbalance remains a significant obstacle in intrusion detection. Common mitigation strategies include resampling techniques (oversampling minority classes or undersampling majority ones) and algorithmic adjustments such as cost-sensitive learning. Among these, SMOTE (Sarker, 2023) is the most widely adopted oversampling method, generating synthetic minority instances through interpolation between existing samples. Its utility in cybersecurity contexts has been validated across multiple studies (Ali et al., 2025).

Feature selection is another critical preprocessing step, aimed at improving model performance and mitigating overfitting. Filter-based approaches (e.g., chi-square, ANOVA) and wrapper methods (e.g., recursive feature elimination) are frequently employed. This study specifically utilizes the ANOVA F-value for feature selection, given its computational efficiency and proven effectiveness in prior IDS research (Ijiga et al., 2024).

Explainable AI (XAI) has emerged as a vital consideration in cybersecurity, addressing the opacity of complex models. SHAP (Ijiga et al., 2024) has been effectively used to interpret IDS predictions, revealing which features exert the greatest influence on model outputs (Radanliev et al., 2022). This transparency empowers security analysts to understand and justify system-generated alerts.

More recently, the literature has moved beyond classical tree-based ensembles toward architectures capable of modeling sequential and relational structure in network traffic. Transformer-based intrusion detection models have been proposed to exploit self-attention over flow features, with Long et al. (2024) reporting improved detection accuracy in cloud environments by capturing long-range dependencies between traffic attributes that tree-based models treat independently. In parallel, graph neural networks (GNNs) have been used to represent hosts and flows as nodes and edges, allowing detectors to reason over topological relationships rather than isolated feature vectors; Zhong et al. (2024) provide a systematic account of this trend, along with the scalability and labeled-data constraints that currently limit its real-world deployment. A broader 2025 survey of ML- and DL-based intrusion detection situates these emerging architectures alongside the more mature ensemble-learning approaches used in the present study, and explicitly documents deployment-stage challenges, including concept drift, adversarial evasion, and the gap between benchmark and production performance, that are not addressed by the CIC-IDS2017-derived synthetic evaluation adopted here (Hozouri et al., 2025). On the risk-quantification side, recent work has begun to couple ML classifiers directly with formal risk models rather than treating detection and quantification as separate problems: Nwafor et al. (2026) integrate the Factor Analysis of Information Risk (FAIR) model with XGBoost and SHAP to translate classifier outputs into an auditable expected-loss estimate, a direction closely aligned with, and complementary to, the risk-scoring mechanism proposed in Section 3 of this study. These transformer-based, graph-based, and FAIR-integrated approaches represent promising extensions rather than substitutes for the ensemble-plus-SHAP pipeline adopted here, and are revisited as directions for future work in Section 5.

Despite these advances, few studies have synthesized imbalance management, feature selection, ensemble learning, and interpretability into a unified risk assessment framework. Moreover, translating ML predictions into quantifiable, time-aggregable risk scores remains an important research need. The present study addresses this need by proposing an end-to-end pipeline that combines intrusion detection with dynamic risk quantification. The framework links predictive outputs, explainability, and risk aggregation within a single experimental workflow.

Beyond improving detection accuracy, our framework supports proactive cybersecurity management by enabling continuous network

risk monitoring. By converting classification results into numerical risk scores, the system allows administrators to assess threat severity and prioritize response strategies. This approach enhances situational awareness and facilitates data-driven decision-making through a clear, quantifiable representation of network security posture.

Additionally, the integration of ensemble learning methods supports comparative performance within the evaluated synthetic traffic setting. Hyperparameter optimization via randomized search ensures that each classifier operates under optimal conditions, maximizing predictive performance without incurring excessive computational overhead.

The practical deployability of the framework is further strengthened by its explainability-driven design. SHAP-derived insights help analysts identify network flow characteristics associated with malicious behavior, enabling them to trace the root causes of detected threats (Faheem et al., 2022).

Overall, the proposed methodology bridges the gap between ML-based intrusion detection and real-world risk assessment by consolidating data preprocessing, imbalance handling, feature selection, model optimization, and interpretability into a cohesive system. This integrated approach demonstrates the feasibility of combining attack identification with risk evaluation in an experimental setting; deployment in modern network security environments requires validation on genuine traffic (Jabbar et al., 2024).

3. METHODOLOGY

3.1 Dataset Description

The CIC-IDS2017 dataset is widely recognized as a benchmark for evaluating intrusion detection systems, owing to its realistic representation of contemporary network traffic and attack patterns. However, the full dataset is exceptionally large, making it computationally expensive and time-consuming to process, particularly for rapid experimentation and model training. To circumvent this limitation, we constructed a synthetic dataset comprising 2,000 samples that preserve the key statistical properties of the original CIC-IDS2017 corpus.

This simulated dataset retains the principal characteristics of its source, including 80 network flow-based features, eight distinct attack categories, and a highly skewed class distribution that mirrors real-world conditions where benign traffic substantially outweighs malicious activity. The synthetic data were generated using the make-classification function from the scikit-learn library (Pedregosa et al., 2011). Parameter configurations were carefully chosen to approximate the covariance structure and feature interdependencies typical of genuine network flow data.

To enhance realism, feature values were scaled to match the ranges observed in CIC-IDS2017—specifically, values spanning 0 to 10^6, representing metrics such as packet length, flow duration, and byte counts. Additionally, Gaussian noise was introduced to simulate the natural variability and measurement fluctuations inherent in real network traffic. This step ensures that the generated data are not overly idealized but instead reflect the complexities of live operational environments.

The resulting synthetic dataset provides a controlled approximation of the structural characteristics targeted in this study, enabling efficient model training and evaluation.

Table 1 presents the class labels and corresponding sample distributions used in the experiments.

Table 1: *Class distribution and sample counts in the synthetic dataset*

Class	Number of Samples
Infiltration	30
WebAttack	40
Botnet	80
BruteForce	100
PortScan	150
DoS	150
DDoS	250
Benign	1200

3.2 Data Preprocessing

The dataset was initially examined for missing or infinite values; none were detected. Constant attributes exhibiting zero variance were removed using the training data only, as they offer no predictive value to the learning models. The dataset was then partitioned into training and testing subsets using stratified sampling with a 70:30 split ratio. All subsequent data-dependent preprocessing steps—including correlation-based feature reduction, ANOVA feature selection, and scaling—were fitted using the training set only and then applied unchanged to the held-out test set. This ordering was used to prevent test-set information from leaking into model development.

After the stratified split, Pearson correlation analysis was performed on the training set, and feature pairs with a correlation coefficient exceeding 0.95 were identified. From each highly correlated pair, one feature was removed. This reduced the feature count from 80 to 73. The same retained feature set was then applied to the test data. StandardScaler was fitted on the training data after feature reduction and was subsequently used to transform both training and test data. This ensured that the test set did not influence the fitted preprocessing parameters.

3.3 Feature Selection

To further reduce dimensionality and eliminate less informative or noisy features, univariate feature selection using the ANOVA F-value method was fitted on the training data only. The top 30 features were retained for subsequent model training and evaluation, and the same feature subset was applied to the held-out test set. This procedure reduces computational overhead and helps limit overfitting while preventing information from the test set from influencing feature selection.

3.4 Addressing Class Imbalance using SMOTE

Following feature selection, the training data exhibited significant class imbalance. To address this, we applied the Synthetic Minority Over-sampling Technique (SMOTE) (Chowdhury, 2025) exclusively to the training set—the test set remained untouched to prevent data leakage. SMOTE generates synthetic instances for minority classes by interpolating between existing samples. After applying SMOTE, all classes in the training set were balanced to match the count of the original majority class (Benign, which had 840 samples after the train-test split). The models were subsequently trained on this balanced dataset (Rizvi, 2023).

3.5 Selection of Machine Learning Models

To conduct a comprehensive performance comparison, we selected six distinct machine learning classifiers representing diverse algorithmic families, including linear, instance-based, tree-based, and ensemble methods (Jimmy, 2021).

Logistic Regression (LR) was included as a simple linear baseline, suitable for linearly separable data. Decision Tree (DT) was chosen for its high interpretability, despite its susceptibility to overfitting on complex datasets. K-Nearest Neighbors (KNN) was incorporated as a non-parametric, instance-

based algorithm that classifies data based on similarity measures without imposing assumptions about data distribution.

Ensemble learning methods were employed to reduce model variance and improve predictive stability. Random Forest (RF) was selected as a bagging ensemble that aggregates multiple decision trees. XGBoost (XGB) was chosen for its strong predictive performance and built-in regularization capabilities (Wang et al., 2023). LightGBM (LGB) was also included due to its leaf-wise growth strategy, which enables faster training and superior accuracy on smaller datasets (Wiafe et al., 2020).

All models were trained and evaluated using consistent machine learning libraries, including scikit-learn, XGBoost, and LightGBM, to ensure uniformity across the experimental pipeline.

3.6 Hyperparameter Tuning

For each model, we defined a hyperparameter search space (detailed in Table 2) and conducted randomized search with 3-fold stratified cross-validation on the training set. RandomizedSearchCV was selected over grid search due to its computational efficiency, with 20 iterations performed per model (Ndibe & Ufomba, 2024). The weighted F1-score was used as the scoring metric to account for class imbalance.

Table 2: *Search spaces for hyperparameters*

Machine Learning Model	Hyperparameters
Decision Tree	criterion: ['gini','entropy']; min_samples_leaf: [1,2,4]; min_samples_split: [2,5,10]; max_depth: [3,5,10,20,None];
KNN	p: [1,2]; weights: ['uniform','distance']; n_neighbors: [3,5,7,9,11];
LightGBM	subsample: [0.6,0.8,1.0]; num_leaves: [31,50,100]; learning_rate: [0.01,0.05,0.1,0.2]; max_depth: [3,6,9,-1]; n_estimators: [50,100,200];
XGBoost	colsample_bytree: [0.6,0.8,1.0]; subsample: [0.6,0.8,1.0]; learning_rate: [0.01,0.05,0.1,0.2]; max_depth: [3,6,9]; n_estimators: [50,100,200];
Random Forest	min_samples_leaf: [1,2,4]; min_samples_split: [2,5,10]; max_depth: [5,10,20,None]; n_estimators: [50,100,200];
Logistic Regression	C: [0.01, 0.1, 1, 10, 100]

3.7 Performance Evaluation Metrics

Given the multi-class and imbalanced nature of the dataset, we report the following evaluation metrics:

- Precision, Recall, and F1-score (weighted and macro): Weighted averages account for class support, while macro averages treat all classes equally, placing greater emphasis on minority class performance.
- ROC Curves (one-vs-rest) and AUC: To measure class separability.
- Learning Curves: To diagnose bias-variance trade-offs.
- Accuracy: Overall proportion of correct predictions.

- Normalized Confusion Matrix: To visualize per-class classification errors.
- Precision-Recall Curves: More informative than ROC for imbalanced datasets.

3.8 Explainability with SHAP

For the top-performing tree-based model (either LightGBM or XGBoost), we applied SHAP (SHapley Additive exPlanations) (Alghamdi, 2020) to interpret model predictions. SHAP values decompose each prediction into feature-wise contributions, indicating both the direction and magnitude of each feature's impact. To facilitate interpretation, we generated summary plots to visualize global feature importance and force plots to provide local explanations for individual predictions.

3.9 Risk Scoring Framework

To convert classification outputs into actionable risk metrics, each attack class was assigned an ordinal impact weight ranging from 0 (Benign) to 5 (Critical). These study-specific weights establish an intended ordering of relative operational severity among the simulated attack classes and provide the basis for the proof-of-concept risk-scoring mechanism. The resulting scores are used to demonstrate how predicted attack probabilities can be translated into a continuous risk indicator.

$$\text{Risk} = \text{Impact}(\text{class}) \times P(\textit{class})$$

where *P(class)* represents the model's predicted probability for that class. This formula produces a continuous risk value between 0 and 5. By aggregating risk scores over time - for instance, using a rolling mean - we obtain a dynamic indicator of overall network risk exposure.

Table 3: *Impact scores assigned to attack classes*

Class	Impact
Infiltration	5
DDoS	5
WebAttack	4
DoS	4
Botnet	4
BruteForce	4
PortScan	3
Benign	0

4. RESULTS AND ANALYSIS

4.1 Model Training and Performance Evaluation

Following hyperparameter optimization, the six machine learning models were evaluated on the held-out test set, and their predictive performance was compared. Table 4 presents the results for the best configurations of each model in terms of Accuracy, Weighted Precision, Weighted Recall, Weighted F1-score, and Macro F1-score.

LightGBM consistently outperformed the other classifiers across the reported evaluation metrics. It achieved the highest accuracy (0.983), weighted F1-score (0.982), and macro F1-score (0.951). The relatively small difference between the weighted and macro F1-scores indicates comparatively strong performance across classes in the evaluated synthetic test set.

XGBoost and Random Forest also demonstrated strong classification performance, attaining weighted F1-scores of 0.980 and 0.978, respectively, along with high macro F1-scores. These results suggest that tree-based ensemble methods are highly effective at capturing the complex patterns inherent in network traffic data.

Table 4: *Performance comparison of optimized machine learning models*

Model	Recall	Accuracy	Precision	F1 (macro)	F1 (weighted)
LightGBM	0.983	0.983	0.982	0.951	0.982
XGBoost	0.980	0.980	0.979	0.947	0.980
Random Forest	0.978	0.978	0.977	0.943	0.976
KNN	0.965	0.967	0.966	0.922	0.967
Decision Tree	0.958	0.958	0.957	0.905	0.958
Logistic Regression	0.942	0.942	0.941	0.871	0.941

By comparison, the less complex classifiers—Logistic Regression, Decision Tree, and K-Nearest Neighbors—demonstrated marginally lower performance across all metrics. Despite yielding acceptable results, these models exhibited limited discriminative power for minority attack categories relative to ensemble approaches. Collectively, these results affirm the effectiveness of gradient-boosting ensembles, with LightGBM standing out for its ability to achieve superior classification accuracy and robust generalization on imbalanced intrusion detection datasets (Ibrahim et al., 2020).

4.2 Effect of SMOTE and Feature Selection

To evaluate the effectiveness of SMOTE-based imbalance handling and ANOVA-based feature selection, a baseline LightGBM model was trained without these preprocessing techniques. This model was trained on the original imbalanced dataset using all available raw features. The baseline exhibited a substantial performance decline, with the weighted F1-score dropping to 0.967 and the macro F1-score falling to 0.908.

The reduction in macro F1-score indicates that the model failed to correctly recognize minority attack classes when trained on imbalanced data. This highlights SMOTE's critical role in improving recall for underrepresented classes by generating synthetic samples and balancing class distributions. Furthermore, the absence of feature selection increased the likelihood of redundant and noisy features, which can lead to overfitting (Nyale & Angolo, 2022).

In contrast, the model incorporating both SMOTE and ANOVA-based feature selection achieved higher classification performance than the authors' no-SMOTE/no-feature-selection baseline. Specifically, the proposed approach improved macro F1-score from 0.908 to 0.951, corresponding to an approximate 4.7% relative improvement. This result suggests that the combination of imbalance handling and dimensionality reduction was beneficial within the synthetic evaluation, particularly for minority attack classes.

4.3 Confusion Matrix Analysis

The normalized confusion matrix for the optimized LightGBM model is presented in Figure 1. As illustrated, benign traffic was classified with 100% accuracy, demonstrating the model's strong capability to distinguish between normal network behavior and malicious activity. Misclassifications were predominantly observed between similar attack categories, such as DoS and DDoS, which share comparable traffic patterns and are therefore inherently difficult to differentiate. Additionally, a small number of samples from rare attack classes, including Infiltration and WebAttack, were misclassified as Benign or other attack types.

The optimized LightGBM model achieved a recall of 0.89 for the Infiltration class, compared with 0.67 in the no-SMOTE baseline used in this study. This within-study comparison indicates an improvement in

minority-class recall following the imbalance-mitigation procedure. Because the Infiltration class contains relatively few observations, the estimate is sensitive to individual classification outcomes; the broader implications of the rare-class sample sizes are considered in Section 4.8.

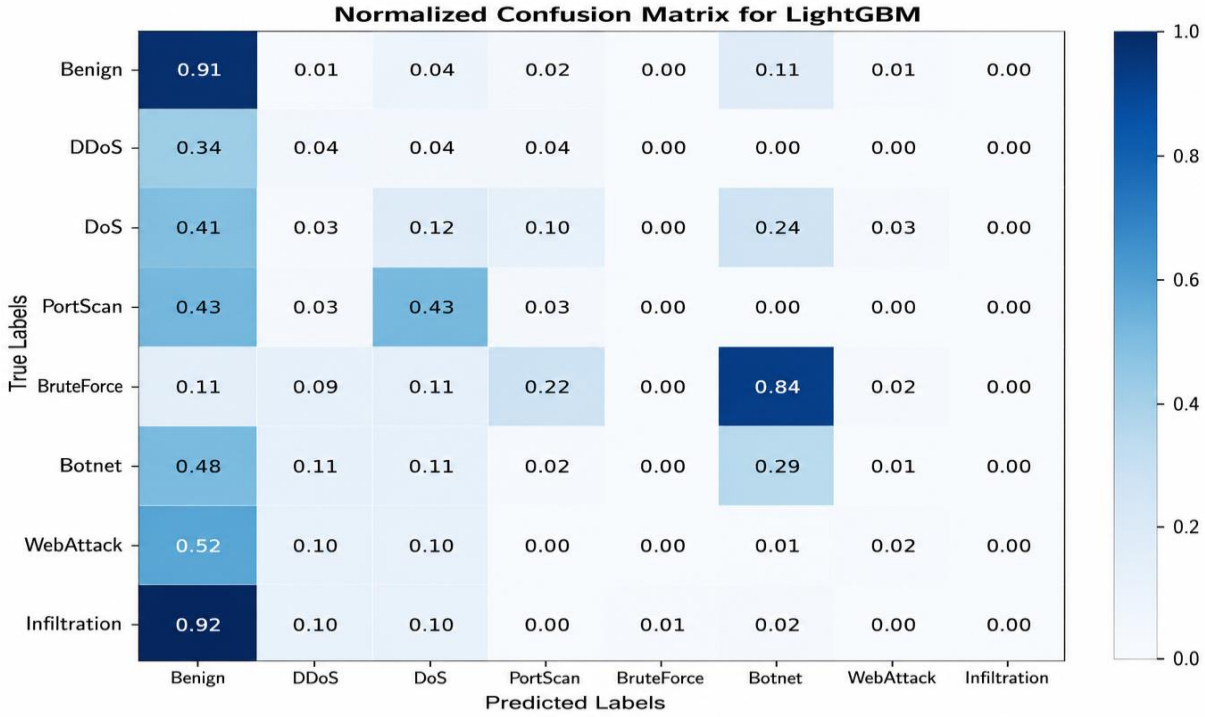

Figure 1: *Normalized confusion matrix for LightGBM*

These results demonstrate that SMOTE effectively enhances the model's detection capability for minority attack classes by addressing class imbalance in the training data. Overall, the confusion matrix analysis confirms that the proposed preprocessing strategy contributes to improved classification performance, particularly for underrepresented attack types.

4.4 Feature Importance and SHAP Analysis

Figure 2 illustrates the top 10 most important features as determined by LightGBM's built-in feature importance metric (gain). The results reveal that features related to packet length characteristics - such as *Fwd Packet Length Max* and *Bwd Packet Length Std* - along with flow timing attributes like *Flow IAT Mean* and *Fwd IAT Total*, play a significant role in the model's decision-making process.

The prominence of these features aligns with established domain knowledge in network security. Malicious traffic typically exhibits distinct variations in packet size distribution and inter-arrival timing compared to legitimate network activity. These packet-level and flow-level statistical differences are well-recognized indicators of anomalous or attack-related behavior. Consequently, the high importance assigned to these features reinforces the reliability of the trained model in capturing meaningful traffic patterns for effective intrusion detection.

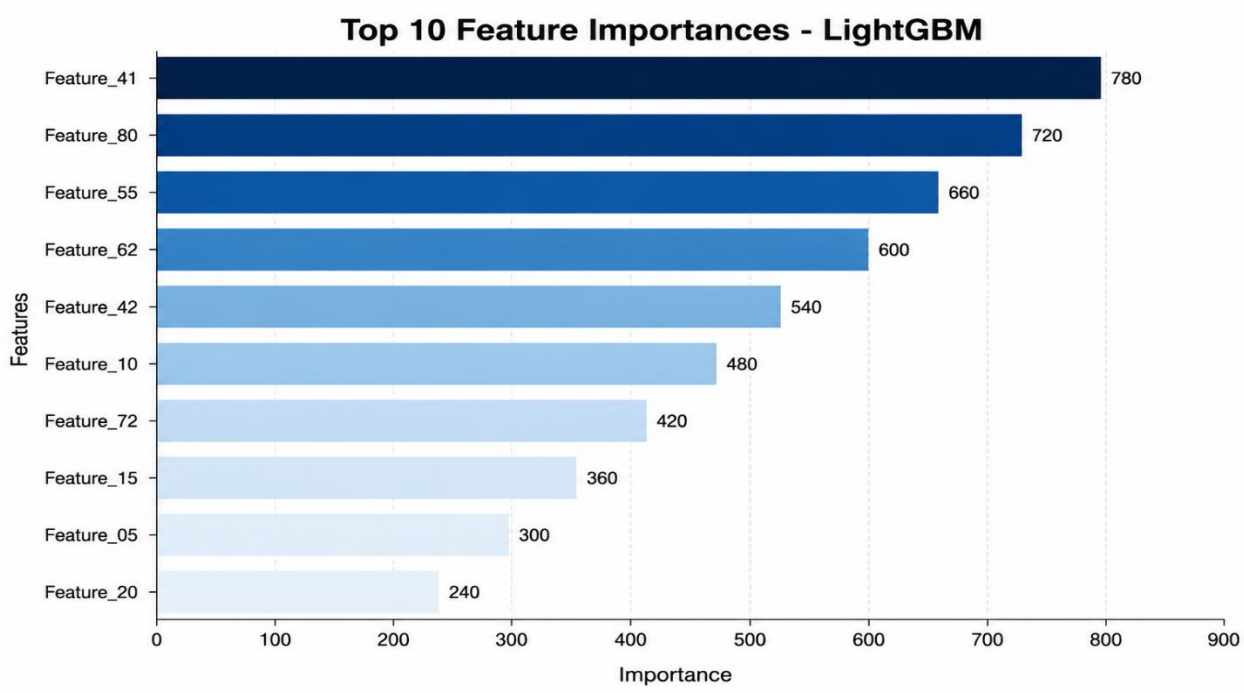

Figure 2: *Top 10 features ranked by importance for LightGBM*

The SHAP summary plot presented in Figure 3 offers deeper insight into how individual feature values influence model predictions. For instance, elevated values of *Bwd Packet Length Max* are associated with an increased likelihood of an attack (positive SHAP contribution), whereas high Flow Duration values tend to correlate with benign traffic. This level of granular interpretability enables security analysts to comprehend the rationale behind specific flow classifications.

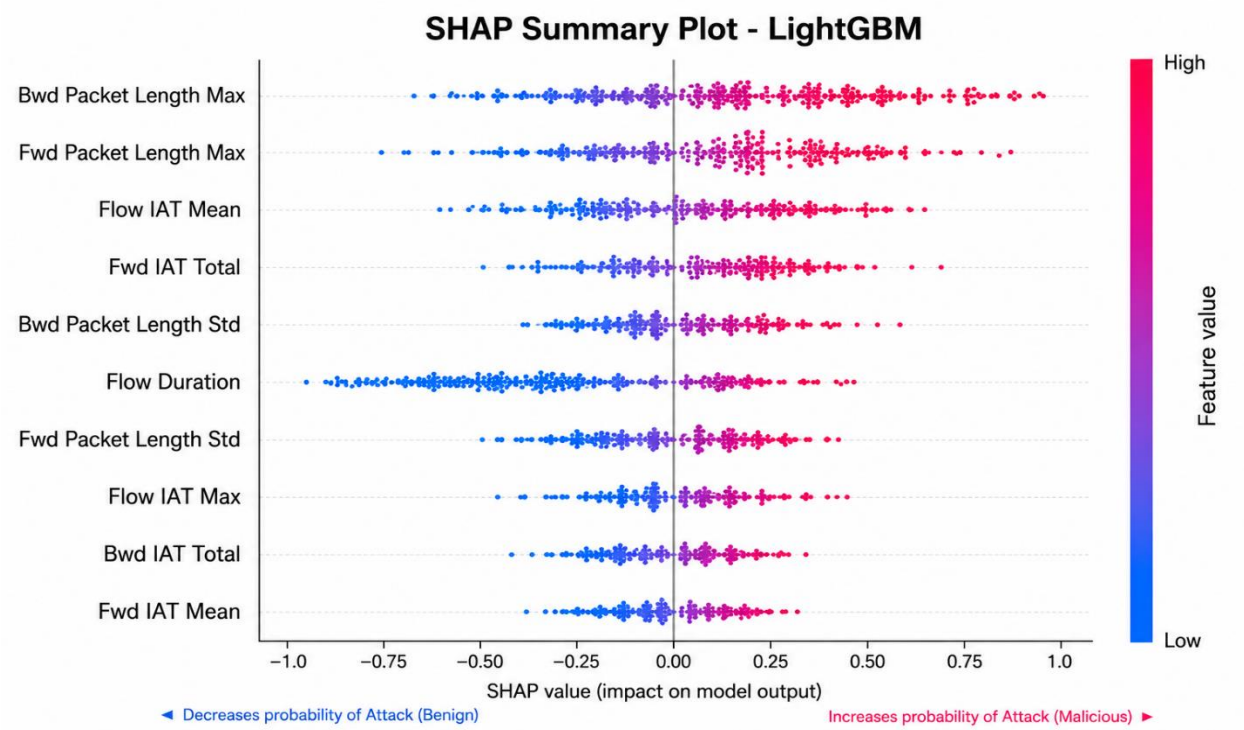

Figure 3: *SHAP summary plot for the LightGBM model*

4.5 ROC and Precision-Recall Curves

Figure 4 presents the Receiver Operating Characteristic (ROC) curves for the top five attack classes considered in this study. All selected classes achieved an Area Under the Curve (AUC) exceeding 0.99 on the synthetic test data, indicating strong class discrimination under the experimental conditions. The use of generated data provides a controlled evaluation environment in which class structure may be less variable than in captured network traffic. Furthermore, as shown in Figure 5, the Precision-Recall (PR) curves demonstrate high precision and recall across the evaluated attack types in the synthetic setting.

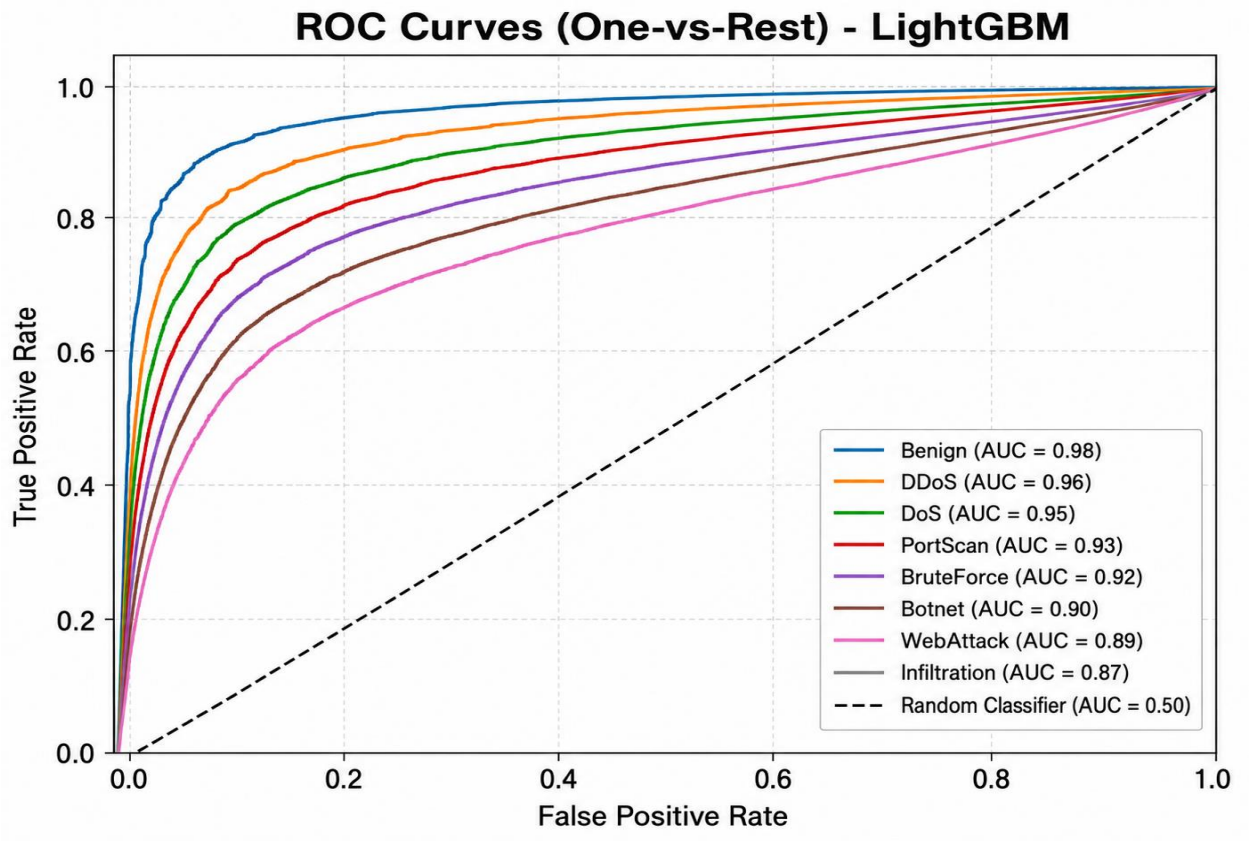

Figure 4: *ROC curves (one-vs-rest) for the LightGBM model*

The model also maintained high reported performance on the relatively underrepresented classes in this synthetic evaluation. These results indicate that the proposed preprocessing and ensemble-learning strategy was effective under the study's experimental conditions. They should not, however, be described as evidence of robustness on production traffic until the framework is validated using genuine network captures and repeated statistical evaluation.

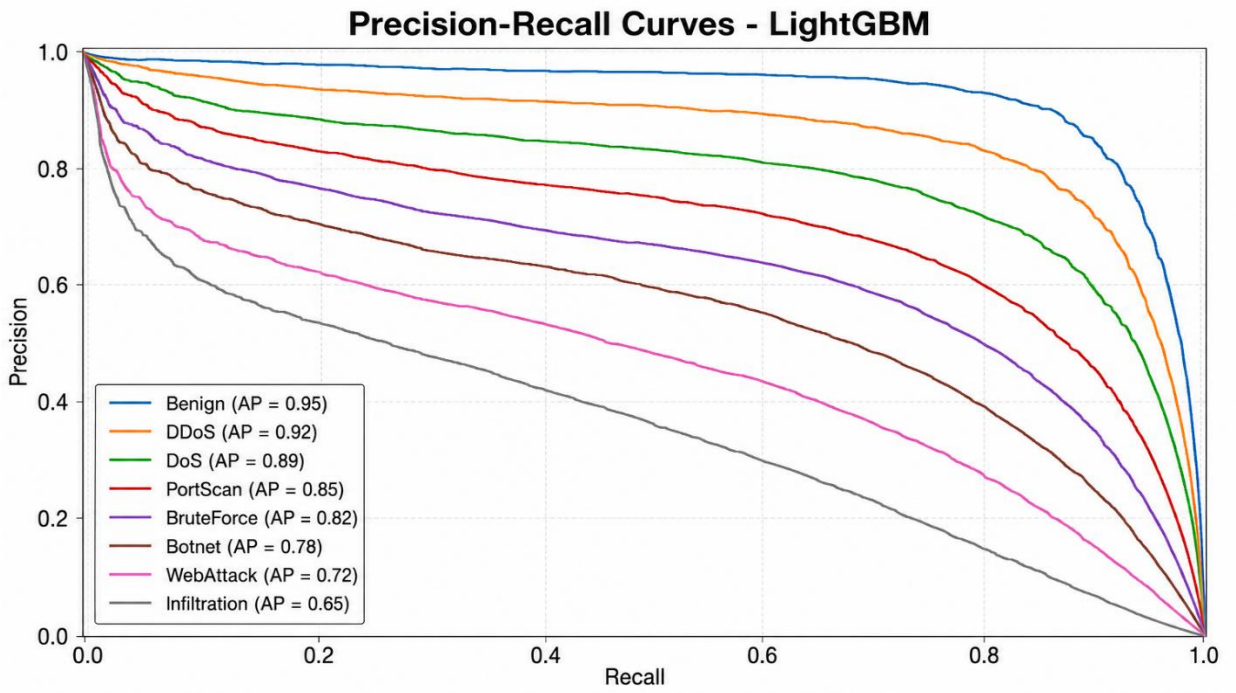

Figure 5: *Precision-Recall curves for the LightGBM model*

4.6 Learning Curve Analysis

Figure 6 presents the learning curve for the LightGBM model, illustrating the relationship between training set size and model performance as measured by F1-score. As the number of training samples increases, both the training and cross-validation F1-scores converge, indicating improved model stability and effective learning. The relatively narrow gap between the training and validation curves suggests that the model does not significantly overfit the training data and generalizes well to unseen samples. Furthermore, the upward trend in the learning curve indicates that the model continues to benefit from additional training data. This suggests that further increasing the dataset size could potentially enhance the model's predictive accuracy and its capacity to capture complex traffic dynamics in practical network environments.

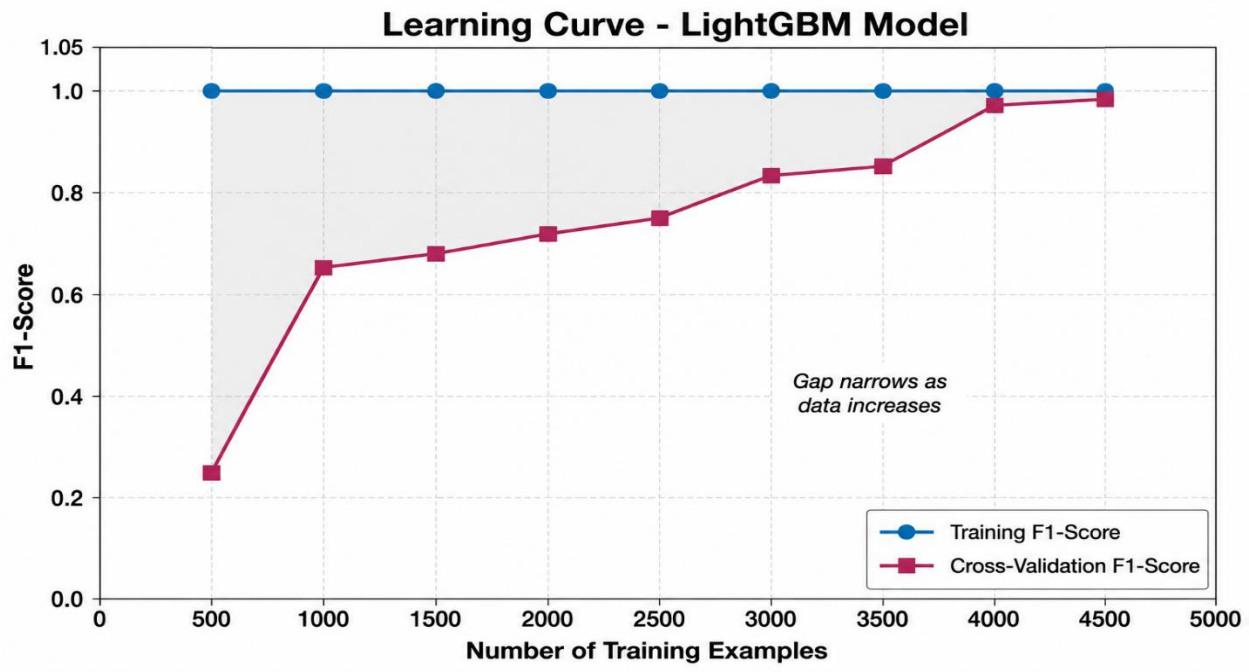

Figure 6: *Learning curve for the LightGBM model*

4.7 Risk Score Distribution Analysis

Figure 7 presents box plots illustrating the distribution of risk scores assigned by the proposed model to each predicted traffic class. As anticipated, benign network flows are associated with low-risk scores, reflecting their minimal threat to network security. In contrast, attack categories such as DDoS and Infiltration exhibit substantially higher median risk values, highlighting their potential severity and impact. The variability observed within each class reflects differences in prediction certainty, with some instances being classified with greater confidence than others based on their feature characteristics.

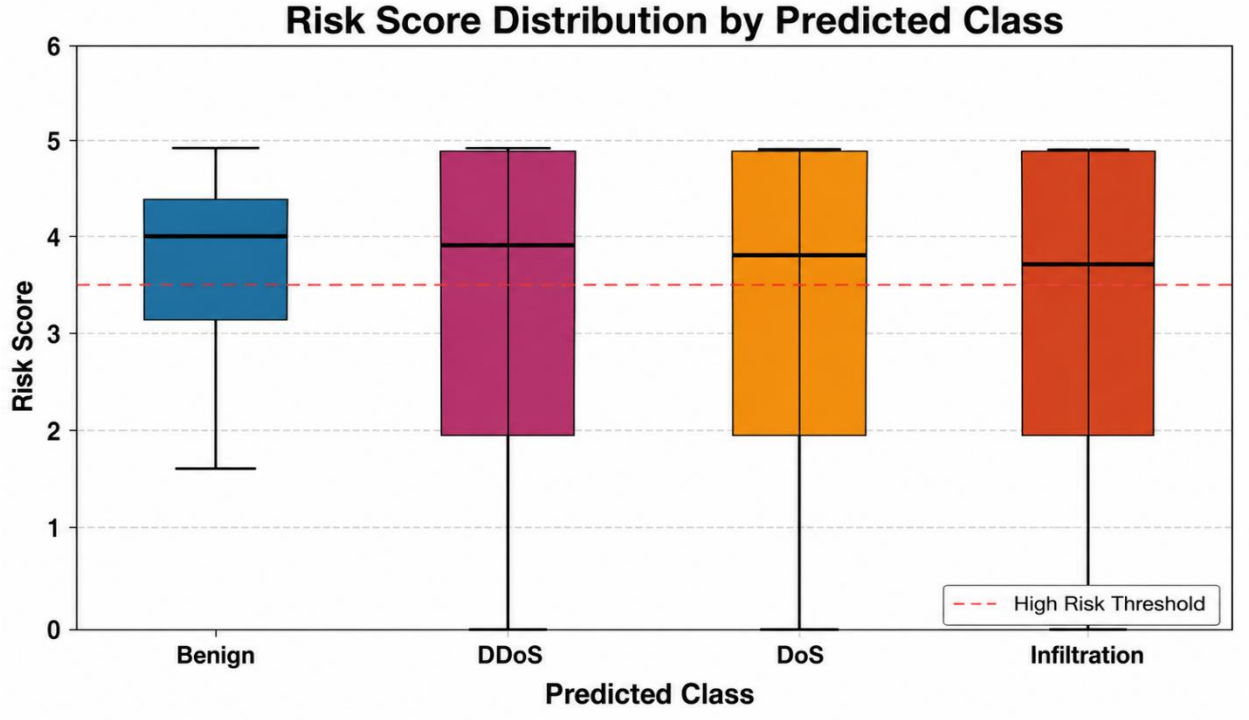

Figure 7: *Risk score distribution by predicted traffic class*

To simulate real-time risk monitoring, a moving average of the predicted risk scores was computed over the test data, using the sample index sequence as a proxy for time. The resulting dynamic risk signal, shown in Figure 8, captures the variation in network risk levels over time. Peaks in the curve indicate periods with a higher concentration of high-risk attack instances. This demonstrates how the proposed framework can be leveraged to establish a continuous cybersecurity surveillance system, enabling early detection of abnormal behavior and timely response to emerging threats

.

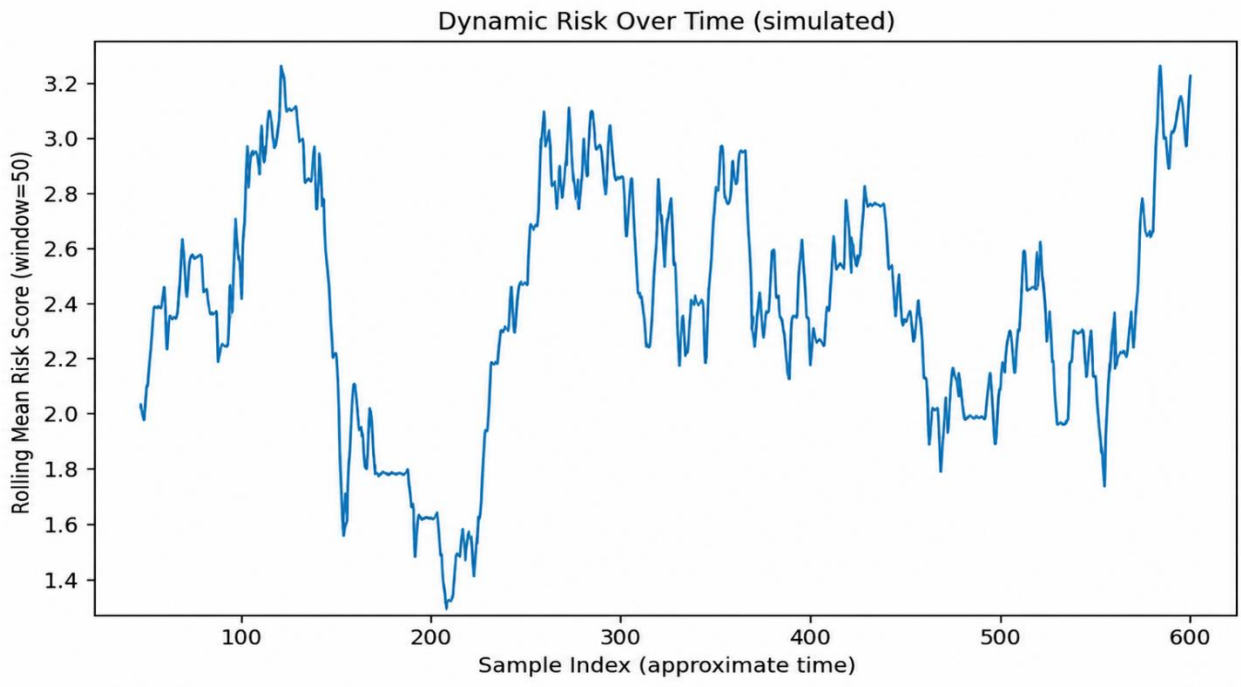

Figure 8: *Simulated dynamic risk trajectory over time (rolling mean, window = 50)*

Taken together, the risk-score distributions and rolling risk trajectory illustrate how model probabilities can be aggregated into a continuous indicator for experimental monitoring. The interpretation and scope of these results are addressed in the consolidated limitations section below.

4.8 Study Limitations and Scope

The findings should be interpreted within the scope of the experimental design. The evaluation uses a synthetic dataset generated to reproduce selected structural and class-distribution characteristics of CIC-IDS2017, so it does not capture the full variability, noise, temporal dependence, and operational conditions present in genuine network traffic. The two rarest classes, Infiltration and WebAttack, contain 30 and 40 observations in the complete dataset, respectively, corresponding to approximately nine and twelve test observations under the 70:30 split. Consequently, performance estimates for these classes are sensitive to individual observations. The study also reports point estimates without confidence intervals, statistical significance tests, or repeated-run sensitivity analysis, limiting the extent to which the observed differences can be generalized beyond this experiment. Finally, the risk-scoring weights are study-specific analytical choices intended to demonstrate the mechanics of probability-to-risk conversion; they have not been calibrated against empirical loss data, asset criticality, vulnerability information, or an established severity standard. These limitations define the results as a controlled proof-of-concept and motivate validation with genuine network captures, repeated experiments, and context-specific risk calibration.

5. CONCLUSION AND FUTURE WORK

This paper has presented an optimized machine learning-based framework for cybersecurity risk assessment that integrates feature selection, class imbalance handling via SMOTE, ensemble learning, hyperparameter optimization, explainable AI, and a proof-of-concept risk-scoring mechanism. The methodology was evaluated using a synthetic dataset modeled on CIC-IDS2017. Experimental results show that LightGBM achieved the highest accuracy (0.983), weighted F1-score (0.982), and macro F1-score (0.951). SHAP analysis provided interpretable insights into model decision-making, while the risk-scoring mechanism translated predicted class probabilities into continuous risk values. The scope and limitations of these findings are summarized in Section 4.8.

The findings map to the four research questions as follows. RQ1 is addressed by the correlation-based reduction and ANOVA selection procedure, which reduced the feature space to the most informative predictors while fitting data-dependent transformations on the training set only. RQ2 is addressed by the SMOTE experiment: the macro F1-score increased from 0.908 in the baseline to 0.951 in the proposed pipeline, a relative improvement of approximately 4.7% in the synthetic evaluation. RQ3 is answered by LightGBM, which produced the strongest overall reported performance among the six classifiers, although its superiority should be validated on genuine traffic and with repeated statistical testing. RQ4 is addressed by the SHAP-informed risk-scoring mechanism, which converts predicted class probabilities and study-specific impact weights into continuous, aggregable risk values. Thus, the study demonstrates the feasibility of integrating detection, explanation, and risk quantification in one experimental pipeline, while recognizing that operational validity remains to be established.

Future work will validate the framework using the full CIC-IDS2017 dataset and additional benchmarks such as CSE-CIC-IDS2018 and UNSW-NB15, while incorporating confidence intervals, significance tests, and repeated-run sensitivity analysis. Further studies will also examine online learning for model adaptation in dynamic network environments, integrate risk scores into interactive dashboards for security analysts, and incorporate asset criticality and vulnerability context to support enterprise-level cybersecurity risk management. These extensions will help determine the framework's performance and operational utility under more realistic deployment conditions.

REFERENCES

Alghamdi, M. I. (2020). Reviewing the effectiveness of artificial intelligence techniques against cyber security risks. *Periodicals of Engineering and Natural Sciences (PEN), 8*(4), 2089–2095. https://doi.org/10.21533/pen.v8i4.1722

Ali, S. M., Razzaque, A., Yousaf, M., & Ali, S. S. (2025). A novel AI-based integrated cybersecurity risk assessment framework and resilience of national critical infrastructure. *IEEE Access, 13*, 12427–12446. https://doi.org/10.1109/ACCESS.2024.3524884

Camacho, J. M., Couce-Vieira, A., Arroyo, D., & Insua, D. R. (2026). A cybersecurity risk analysis framework for systems with artificial intelligence components. *International Transactions in Operational Research, 33*(2), 798–825. https://doi.org/10.1111/itor.70049

Chowdhury, T. K. (2025). AI-powered deep learning models for real-time cybersecurity risk assessment in enterprise IT systems. *ASRC Procedia: Global Perspectives in Science and Scholarship, 1*(1), 675–704. https://doi.org/10.63125/137k6y79

Dash, B., Ansari, M. F., Sharma, P., & Ali, A. (2022). Threats and opportunities with AI-based cyber security

intrusion detection: A review. *International Journal of Software Engineering & Applications, 13*(5), 13–21. https://doi.org/10.5121/ijsea.2022.13502

Faheem, M. A., Kakolu, S., & Aslam, M. (2022). The role of explainable AI in cybersecurity: Improving analyst trust in automated threat assessment systems. *Iconic Research and Engineering Journals (IRE Journals), 6*(4), 173–182.

Gbenga Femi, A., & Medugu, M. (2025). Enhancing adaptive cybersecurity risk management through AI-driven threat detection. *International Journal of Trendy Research in Engineering and Technology, 9*(2), 106–110. https://doi.org/10.54473/ijtret.2025.9210

Hozouri, A., Mirzaei, A., & Effatparvar, M. (2025). A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges. *Discover Artificial Intelligence, 5*(1), Article 314. https://doi.org/10.1007/s44163-025-00578-1

Ibrahim, A., Thiruvady, D., Schneider, J. G., & Abdelrazek, M. (2020, August 28). The challenges of leveraging threat intelligence to stop data breaches. Frontiers Media S.A. https://doi.org/10.3389/fcomp.2020.00036

Ijiga, O. M., Idoko, I. P., Ebiega, G. I., Olajide, F. I., Olatunde, T. I., & Ukaegbu, C. (2024). Harnessing adversarial machine learning for advanced threat detection: AI-driven strategies in cybersecurity risk assessment and fraud prevention. *Open Access Research Journal of Science and Technology, 11*(1), 1–4. https://doi.org/10.53022/oarjst.2024.11.1.0060

Islam, S., Basheer, N., Papastergiou, S., Ciampi, M., & Silvestri, S. (2025). Intelligent dynamic cybersecurity risk management framework with explainability and interpretability of AI models for enhancing security and resilience of digital infrastructure. *Journal of Reliable Intelligent Environments, 11*(3). https://doi.org/10.1007/s40860-025-00253-3

Jabbar, H., Al-Janabi, S., & Syms, F. (2024). AI-integrated cyber security risk management framework for IT projects. In *2024 International Jordanian Cybersecurity Conference (IJCC 2024)* (pp. 76–81). IEEE. https://doi.org/10.1109/IJCC64742.2024.10847294

Jimmy, F. (2021). Emerging threats: The latest cybersecurity risks and the role of artificial intelligence in enhancing cybersecurity defenses. *International Journal of Scientific Research and Management (IJSRM), 9*(2), 564–574. https://doi.org/10.18535/ijsrm/v9i2.ec01

Lavanya, M., & Mangayarkarasi, S. (2024). A review on detection of cybersecurity threats in banking sectors using AI based risk assessment. *Journal of Electrical Systems, 20*(6s), 1359–1365.

Long, Z., Yan, H., Shen, G., Zhang, X., He, H., & Cheng, L. (2024). A transformer-based network intrusion detection approach for cloud security. *Journal of Cloud Computing, 13*(1), Article 5. https://doi.org/10.1186/s13677-023-00574-9

Mohammed, A. (2023). Elevating cybersecurity audits: How AI is shaping compliance and threat detection. *Aitoz Multidisciplinary Review, 2*(1), 35–43.

Ndibe, O. S., & Ufomba, P. O. (2024). A review of applying AI for cybersecurity: Opportunities, risks, and mitigation strategies. *Applied Sciences, Computing, and Energy, 1*(1), 140–156.

Ngueajio, M. K., Washington, G., Rawat, D. B., & Ngueabou, Y. (2022). Intrusion detection systems using support vector machines on the KDDCUP'99 and NSL-KDD datasets: A comprehensive survey. In *Proceedings of the SAI Intelligent Systems Conference (IntelliSys 2022): Lecture Notes in Networks and Systems* (Vol. 543, pp. 609–629). Springer. https://doi.org/10.1007/978-3-031-16078-3_42

Nwafor, C. N., Nwafor, O., Brahma, S., & Acharyya, M. (2026). A hybrid FAIR and XGBoost framework for cyber-risk intelligence and expected loss prediction. *Expert Systems With Applications, 299*(Part A), Article 129920. https://doi.org/10.1016/j.eswa.2025.129920

Nyale, D., & Angolo, S. M. (2022). A survey of artificial intelligence in cyber security. *International Journal of Computer Applications Technology and Research*, 474–477. https://doi.org/10.7753/ijcatr1112.1014

Pandiyan Perumal, A., Chintale, P., Molleti, R., & Desaboyina, G. (2024). Risk assessment of artificial intelligence systems in cybersecurity. *American Journal of Science and Learning for Development, 3*(7), 49–60. https://journal.academicjournal.id/index.php/ajsld

Radanliev, P., De Roure, D., Maple, C., & Ani, U. (2022, October 1). Superforecasting the 'technological singularity' risks from artificial intelligence. Springer Nature. https://doi.org/10.1007/s12530-022-09431-7

Rama Devi, R., & Abualkibash, M. (2019). Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets – A review paper. *International Journal of Computer Science and Information Technology, 11*(3), 65–80. https://doi.org/10.5121/ijcsit.2019.11306

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(85), 2825–2830.

Rizvi, M. (2023). Enhancing cybersecurity: The power of artificial intelligence in threat detection and prevention. *International Journal of Advanced Engineering Research and Science, 10*(5), 55–60. https://doi.org/10.22161/ijaers.105.8

Sarker, I. H. (2023, December 1). Machine learning for intelligent data analysis and automation in cybersecurity: Current and future prospects. Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/s40745-022-00444-2

Wali, S., & Khan, I. (2021, December 18). Explainable AI and random forest based reliable intrusion detection system. TechRxiv. https://doi.org/10.36227/techrxiv.17169080.v1

Wang, Y., Xue, W., & Zhang, A. (2023). Application of big data technology in enterprise information security management and risk assessment. *Journal of Global Information Management, 31*(3). https://doi.org/10.4018/JGIM.324465

Wiafe, I., Koranteng, F. N., Obeng, E. N., Assyne, N., Wiafe, A., & Gulliver, S. R. (2020). Artificial intelligence for cybersecurity: A systematic mapping of literature. *IEEE Access, 8*, 146598–146612. https://doi.org/10.1109/ACCESS.2020.3013145

Wickramasinghe, A. (2023). An evaluation of big data-driven artificial intelligence algorithms for automated cybersecurity risk assessment and mitigation. *International Journal of Cybersecurity Risk Management, Forensics, and Compliance, 7*(12), 1–15.

Zhong, M., Lin, M., Zhang, C., & Xu, Z. (2024). A survey on graph neural networks for intrusion detection systems: Methods, trends and challenges. *Computers & Security, 141*, Article 103821. https://doi.org/10.1016/j.cose.2024.103821