Journal of
Cyber Security

Tech Science Press

https://doi.org/10.32604/jcs.2026.086419

ARTICLE

Bias and False Positive Challenges in AI-Based Intrusion Detection Systems under Extreme Class Imbalance

John Ojo Ajayi* **and Grace Egenti**

Department of Cybersecurity, National Open University of Nigeria, Abuja, Nigeria

*Corresponding Author: John Ojo Ajayi. Email: ace21150008@noun.edu.ng

Received: 30 May 2026; Accepted: 26 August 2026; Published: 14 September 2026

ABSTRACT: Artificial intelligence (AI)-based intrusion detection systems (IDS) have evolved to be fundamental in detecting cyberattacks in contemporary networks. Unfortunately, the significant class imbalance present in cybersecurity datasets may introduce unfair biases during model learning, producing unpredictable predictions and inflating false alarm alerts, which eventually hampers practical deployment. This research examines the influence of class imbalance mitigation on model performance, operational fairness, explainability, and the operational integrity of an AI-based IDS subject to extreme class imbalance conditions. Three learning strategies based on Random Forest (i.e., baseline, Synthetic Minority Over-sampling Technique (SMOTE), and cost-sensitive learning) are evaluated on the CIC-BCCC-NRC TabularIoTAttack-2024 dataset and integrated into a systematic evaluation process encompassing SHAP explainability, statistical validation, external validation, and deployment analysis. The evaluation outcomes establish that the Baseline Random Forest, SMOTE Random Forest, and Cost-Sensitive Random Forest achieved classification accuracies of 99.9278%, 99.8866%, and 99.9246%, respectively, with all three models achieving superior classification accuracy of over 99.88% but displaying variations in their operational behaviour. The Baseline Random Forest demonstrated overall strong predictive capability, while the SMOTE Random Forest had the highest balanced accuracy and specificity and the lowest false-positive rate. Although SMOTE achieved the best internal false-positive performance, it is presented as a comparative imbalance-mitigation strategy rather than the recommended deployment model. The Cost-Sensitive Random Forest showed the capacity to remain competent while sustaining a pragmatic equilibrium between threat identification performance and deployment utility. SHAP analysis further revealed distinct feature importance patterns across the three models. Temporal inter-arrival features were the dominant predictors for the Baseline Random Forest, whereas the SMOTE Random Forest emphasized packet-size related features and the Cost-Sensitive Random Forest was primarily influenced by ACK Flag Count, Src Port, and Flow Bytes/s, thereby improving model interpretability. Moreover, an important increase in false positive behaviours was also found when externally validated using 2250 previously unseen legitimate network flows from the BCCC-MalNetMem-2025 dataset, where the external False Positive Rate increased to 23.51% (95% Wilson confidence interval: 21.80%–25.31%) compared with the internal benchmark evaluation, revealing an important gap between benchmark evaluation and real-world deployment. The results herein prove that combining imbalance-aware learning, explainable AI, statistical validation, and external validation lead to a more dependable and believable framework for assessing and designing AI-based IDSs for real-world cybersecurity deployments.

KEYWORDS: Artificial intelligence; class imbalance; explainable artificial intelligence; external validation; intrusion detection system; Random Forest; SHAP

1 Introduction

The use of AI in cybersecurity systems has revolutionized the detection of threats, identification of abnormalities, and management of incidents. These systems provide great advantages in terms of performance, scalability, and enabling machines to process a much larger scale of information that would be impossible for humans to handle. Yet, the widespread use of these systems has brought about serious ethical concerns in terms of fairness, accuracy, and accountability in contexts of decision-making, which may produce fatal consequences if any error should occur. Latest literature concludes that while AI enhances efficiency, it presents threats of bias, inconsistencies, and mistakes that may further compromise credibility in automated systems [1]. However, reliance on AI in the realm of security decision-making raises a series of issues related to fairness, bias, and accuracy. It is already known from the literature that AI systems may inherently hold biases from their training, architecture, and deployment stages, leading to biased decisions [2] that could harm security accuracy and lead to serious ethical implications within security decisions.

In many operational networks, benign traffic is the majority, whereas in the CIC-BCCC-NRC TabularIoTAttack-2024 benchmark used in this study, attack traffic constitutes approximately 99% of the samples after preprocessing, and benign traffic forms the minority class. Consequently, SMOTE synthesizes benign network flows in this benchmark. Class imbalance is an issue that many real-world models are facing, but it is one of the most prominent sources of biases for AI models. In furtherance of this, extreme class imbalance causes models to become biased toward the majority class, which may lead to inaccurate and misleading performance evaluation measures, which further negatively impacts models and accuracy [3,4]. The result can contribute to models that are very accurate yet do not correctly detect the instances that are typical of the minority class or incorrectly estimate the minority class, which will produce excessive false positives. Several methods have been proposed to address the class imbalance problem, such as data-level methods like SMOTE and algorithm-level methods like cost-sensitive learning. The main goal of the strategies is to improve classification performance by balancing class distribution or altering the learning priorities. But it is worth noting that the bias mitigation methods in recent research are considered to involve trade-offs, i.e., the improvement in some indicator would bring degradation in another (especially when dealing with real-world decision problems) [5]. For instance, oversampling methods lead to an increase in recall as well as false alarms, while cost-sensitive methods reduce false positives but result in an increase in lost detections. This trade-off is of great importance in security systems, where the trade-off between detection, accuracy, and reliability of operation must be carefully managed. Despite these advancements, however, many existing works solely report accuracy and recall; few studies address false-positive behaviour, model interpretability, statistical validation, computational efficiency during deployment, or external validation with independent traffic data. Thus, the reported performance is likely an overestimation of the true reliability of intrusion detection models during operation.

In operational networks, benign traffic generally constitutes the overwhelming majority, whereas the benchmark dataset used in this study exhibits the opposite class distribution after preprocessing. However, this distribution differs from that observed in many real-world datasets. In this project, the CIC-BCCC-NRC TabularIoTAttack-2024 dataset was used, which, after preprocessing, will primarily consist of attacking traffic, providing a valuable opportunity to experiment with new imbalance solutions in a unique class imbalance context. Beyond predictive performance, the deployment implications are often left out of the IDS literature. Studies claiming excellent detection results fail to account for computational efficiency, inference latency, model size, or processing throughput, all factors that will critically influence the usability of AI-based IDS. A model that performs very well on the benchmark, but requires extremely large computational resources,

or leads to significant detection delays, may not be deployed. As such, deployment-aware evaluation has emerged as essential for real-world deployment of ML techniques in cybersecurity applications.

Another major constraint in the current studies is the evaluation based on benchmarks, where models are evaluated on data from the same distribution as the training data. Such evaluations may report extremely high-performance figures, but they can be inaccurate when compared to real-world conditions due to differing data distributions. AI exhibits performance decrease in unexpected contexts, necessitating enhanced resilience [6,7]. Generalisation is a fundamental requirement within the security domain, as models are anticipated to operate with evolving data streams. The crucial problem of AI-based security systems arises because a system that performs well in the controlled environment might not succeed in the real world with high numbers of false positives. Secondly, fair machine learning is becoming one of the critical challenges in decision-making systems. Fairness is the extent to which algorithms' outputs do not discriminate against some groups or behaviours more than others. However, fairness cannot be easily obtained due to conflicting definitions and trade-offs between accuracy and equality. In the cybersecurity landscape, unfairness might take the form of too many false positives on benign traffic, or detection patterns that are biased against certain network behaviours. In this study, operational fairness is defined as ensuring that IDS mitigate biased classification behaviours that arise from high class imbalance. Operational fairness means minimizing the excessive misclassification of minority-class network traffic while sustaining robust attack-detection performance. This study attempts not only to detect several types of cyberattacks but also to create balanced decisions, avoiding bias in minority classes while the overall majority is correctly classified. This means we ensure unbiased decision-making in a high-class imbalance setting, thus creating a trustworthy and operationally stable AI intrusion detection system.

This study aims to demonstrate the effects of class imbalance mitigation methods on predicted performance, operational fairness, false positive behaviour, explainability, statistical validity, and readiness for deployment of a machine learning-based intrusion detection system. Two optimized Random Forest variations (SMOTE-based oversampling and cost-sensitive learning), along with a basic classifier, are evaluated with regard to a subset of the CIC-BCCC-NRC TabularIoTAttack-2024 dataset. In addition to traditional measures of predicted performance (e.g., accuracy, MCC, ROC-AUC, and precision-recall AUC), analysis is extended with consideration of 95% confidence interval estimations for all performance indicators, external testing on unknown benign traffic, significance testing using McNemar's test, explainability (via SHAP), and internal robustness.

A further weakness of several existing intrusion detection studies is the poor interpretability of machine learning models. Although ensemble learning algorithms generally have high predictive accuracy, many are treated as black-box models where security experts struggle to understand why a given traffic session is classified as benign or malicious. Explainable AI (XAI) approaches, such as SHAP (SHapley Additive exPlanations), enable robust feature contributions and provide the security expert with understandable justifications for individual model predictions. The resultant increased trust in the model allows for efficient auditing and the detection of potential biases in security decision-making. The augmented confidence leads to more meaningful security evaluations and allows researchers to reveal biases in security decisions. Most prior works for anomaly detection rely primarily on a single train-test division, which may provide limited evidence of model stability or statistical reliability [8]. The application of confidence interval estimation, the McNemar test of significance, and stratified cross-validation on the various splits of the dataset more conclusively supports the fact that the reported improvements were not due to chance.

In conclusion, this work provides valuable contributions to building reliable AI-based intrusion detection systems through an organised comparative analysis of both data- and algorithm-level balancing techniques for extremely imbalanced data. Aside from a thorough set of performance metrics (accuracy,

precision, recall, F1-score, AUC-ROC, MCC), the work also encompasses explainable AI via SHAP, statistical testing via confidence intervals, McNemar's test, stratified cross-validation, validation against previously unknown benign traffic (external validation), and model performance assessment from the perspective of its computational cost, size, and inference speed and capacity (deployment analysis). These elements together allow a realistic appraisal of an IDS in real-world cybersecurity deployments. The models were assessed not only based on accuracy but also on efficiency and the capability for real-time deployment into enterprise/IoT intrusion detection environments, differentiating them from several existing works that only focus on benchmark accuracy. In addition to comparing predictive performance, this study investigates whether algorithm-level cost-sensitive learning offers a more practical balance between detection effectiveness and deployment efficiency than conventional data-level oversampling techniques. The presented results are expected to help cybersecurity engineers and scientists in identifying an intrusion detection model that has excellent predictive power, as well as a low false alarm rate, strong interpretability, statistical robustness, and efficient deployment for online execution.

2 Related Works

The emergence of AI has significantly influenced IDSs by enhancing early threat detection, anomaly identification, and automated response functionalities. In current IDS research, machine learning and deep learning methods are predominantly adopted owing to their capabilities to handle vast amounts of network traffic, discover novel attack scenarios, and facilitate automatic threat mitigation [9,10]. However, even with such advancements in IDS, several limitations, such as class imbalance, operational fairness, interpretability, and false positive behaviour are yet to be adequately addressed in many of these AI-based security frameworks.

Machine learning approaches for intrusion detection have been applied in several pieces of research with various benchmark datasets namely, NSL-KDD, CICIDS2017, UNSW-NB15, ToN_IoT, etc. Kocher and Kumar [11] discussed contemporary applications of machine and deep learning in IDS and found that ensemble and deep learning methods achieve superior accuracy on attack detection when compared with other established methods. Similar to this study, Maseer et al. [12] evaluated different machine learning algorithms using the CICIDS2017 data and concluded that Random Forest and ensemble methods achieve the best performance on anomaly detection. In addition, Gu and Lu [13] introduced an intrusion detection model using SVM-based feature embedding for enhanced attack detection performance.

Lately, studies on IDS use ensemble learning and hybrid architectures for improved performance in imbalanced class distributions. For example, Abbas et al. [14] proposed an IoT IDS in the form of ensemble learning based on various classifiers to obtain high attack detection. Abirami et al. [15] presented ensemble-based IDS research and concluded that ensemble learning results in more robustness and detection consistency compared to a single classifier. Similarly, Naz et al. [16] compared a set of ensemble learning methods for intrusion detection and concluded that ensembles based on Random Forests have robust performance for most attack classes. Ajayi et al. [17] described an IDS using the Super Learner ensemble approach; the results of the study showed excellent detection accuracy and that the use of ensembles led to better detection rates and fewer false alarms. However, the study's limitations included a lack of discussion about fairness, imbalance, and external validation of false alarms in real scenarios.

Although SMOTE is widely used as a data-level technique for addressing class imbalance, cost-sensitive learning methods at the algorithm level have gained popularity because they maintain the original data distribution while assigning higher penalties to the misclassification of minority-class samples. Cost-sensitive learning has obtained promising performance in detection with low false alarms and reduced computational load compared to heavy oversampling. The combination of cost-sensitive learning and

ensemble improves the detection of minority attack classes with acceptable costs for practical IDS, according to some studies [18,19].

Explainability is another trend in AI security systems; for instance, Larriva-Novo et al. [20] use explainable AI techniques in real-time cyberattack detection and discovered that explainability increases trustworthiness and interpretability of an IDS. Arreche et al. [21] introduce the XAI-IDS framework to increase the transparency of IDS.

Fairness and bias in AI systems are critical issues, and many application domains were affected; among them is the cybersecurity domain. Ferrara [22] identified data, algorithm, and human decision as sources for bias in AI systems. Gallegos et al. [2] argue that skewed data distributions and under-represented minority classes appear to be one of the causes of unfair AI decisions. Pagano et al. [23] present a systematic literature review on bias mitigation strategies in machine learning and show that no single metric and mitigation technique fits all contexts. Again, Zhou et al. [24] introduce lossless debiasing approaches without losing accuracy in classification.

The AI safety literature has recently addressed concerns about deployment reliability; for example, the International AI Safety Report [1] points out that benchmark-based evaluations do not reflect the operational behaviour of the systems. Yang et al. [25] state that the performance of IDS models degenerates in noisy or unknown traffic distributions. Islam [26] observe that while AI-based IDS achieve very high accuracy on benchmark datasets, they do not yield trustworthy alerts for real-world enterprise security environments, and their heterogeneous traffic patterns and extreme imbalance issues are still unsolved. Another area of significant research for IDS has been based on their deployability characteristics, not just their prediction capabilities. Model size, computational complexity, processing rate, and inference delay are the most important considerations for deciding whether a machine learning algorithm can actually achieve performance at the level required for real-time deployments of an enterprise or for Internet of Things security contexts [27]. The growing reliance on external validation (e.g., datasets that were not included in the original training data or network traffic data) and using unseen data have provided better evaluation for model generalization capability. Models that are solely evaluated using datasets that were on benchmarks tend to show poor performance in real-time environments because of a difference in distribution between the traffic at training time and in production. As such, external validation is often considered a reliable method to verify the usability of AI-based IDS [28].

The problem of false positives is still critical in AI-driven intrusion detection systems. Most papers prioritise accuracy and recall, neglecting the importance of reliability and minimising false alarms. Najafimehr et al. [29] noted that although machine learning has significantly improved DDoS detection, achieving high detection rates while maintaining low false positive rates remains a major challenge because of the diversity of attack patterns and the heterogeneity of network traffic. Mohammad et al. [30] found that data augmentation improves detection rates, but it also increased false positive rates in IDS, which reveals that improving detection performance is not sufficient for real-world application. More recently published work on intrusion detection approaches suggests the need to supplement these traditional performance metrics with additional statistical validation to strengthen comparisons of the metrics. Confidence intervals, stratification, and statistical hypothesis tests offer significantly greater assurance that observed improvements are genuine, repeatable enhancements rather than just artefacts of the random division of the train-test data. The use of this type of testing should become increasingly important given the severity of class imbalance present in security data [31].

In many survey papers published between 2025 and 2026, new challenges in AI-driven intrusion detection systems are reported. To substantiate this claim, Hozouri et al. [32] discuss the problem of adversarial robustness, explainability, and deployability in IDS. Boateng et al. [33] review a variety of AI-based

cyberattack detection methods and conclude that trustworthy AI frameworks are essential in cybersecurity. Several special issues on AI-based IDS emphasise the lack of datasets and evaluation frameworks that handle imbalanced data and operational reliability challenges. Despite substantial recent advances in AI-based IDS, several gaps still exist in prior research. First, several of the extant efforts primarily focus on maximizing classification accuracy without significant evaluation of other relevant metrics, including false positive behaviour, MCC, and balance under highly imbalanced conditions. Second, compared with data-level techniques like oversampling, the use of algorithm-level cost-sensitive learning has been less explored, yet it can provide significant computational benefits. Third, the explanation of models is typically analyzed in isolation without simultaneous evaluation in conjunction with imbalance mitigation and fairness analysis. Fourth, most related work uses dataset internal validation, and external validation with truly new benign samples is rarely performed, leading to a loss of confidence in model generalization. Fifth, no thorough and statistically validated system evaluation including factors related to deployment (e.g., computational cost, inference speed, model size, and throughput) is reported.

This study addresses these research gaps by offering a holistic and comparative assessment of data-level and algorithm-level methods for reducing class imbalance in the context of AI-based intrusion detection. The framework proposed in this paper goes beyond that of typical papers by evaluating, in a combined manner, predictive performance, false positive behaviour, interpretability, statistical robustness, external validation and readiness to deploy. Moreover, our framework complements, in a unified way, SHAP-based explainability, confidence interval estimation, McNemar's test, stratified cross-validation, and readiness for deployment analysis for those models designed for actual cybersecurity practice.

3 Methodology

This study proposes and implements a quantitative experimental strategy to evaluate the effects of class imbalance strategies on classifier performance and false positive rates in AI-based security systems. The experimental design ensures a meaningful comparison across models and eliminates data leakage, allowing an accurate performance study on separate datasets.

3.1 Research Design

This study investigates three different modelling strategies to cope with an extreme class imbalance problem in IDS: A simple baseline model utilising the raw imbalanced dataset, an oversampling SMOTE model and a cost-sensitive reweighted model. Specifically, how the imbalance-management methods can affect classification accuracy and operational reliability was looked into. The approach also incorporates external validation and an explanation of AI analysis to improve reliability and transparency when used in practice [2]. In addition, the experimental setup included statistical validation methods by means of confidence interval calculation, McNemar's significance testing, and fivefold stratified cross-validation to assess the resilience of the designed models. Operational deployment features like model size, training duration, inference delay and output performance were also analyzed to understand the readiness for usage of each imbalance handling approach.

3.2 Dataset Description

The dataset chosen for this work is the CIC-BCCC-NRC TabularIoTAttack-2024 [34]. This dataset, compiled by the Canadian Institute for Cybersecurity (CIC), the BCCC research group and NRC Canada, is created to aid in assessing both IDS and AI-based security systems in IoT networks. It contains various types of traffic, including normal traffic and several types of attacks, namely DDoS, DoS, reconnaissance, MQTT attacks and MITM. A high-class imbalance is prevalent throughout the dataset (99% attack, 1% normal) after

merging and preprocessing, allowing for an investigation into biases and fairness of AI models, as shown in the class distribution in Table 1. Its realism and variety of attack vectors make it an ideal choice to measure real-world reliability and explore methods of mitigating imbalance.

Table 1: Class distribution of the dataset.

Attack	3,245,682
Benign	32,606
Total	3,278,288

The CIC-BCCC-NRC TabularIoTAttack-2024 dataset consists of numerous network flow features which yield statistical, temporal, protocol and behavioural details on network traffic, shown in Table 2. The selected features capture different aspects of traffic behaviour, providing them with a high utility in identifying intrusion, and are effective for classifying attacks and detecting anomalies in a machine learning-based security system [8]. After preprocessing the dataset, the four metadata attributes were eliminated because they are not directly used for traffic classification (flow ID, src IP, dst IP, and timestamp), and they may provide an undesirable bias to the models. Thus, the final modeling dataset contains 79 numerical predictive features for training machine learning algorithms.

Table 2: Representative network flow features extracted from the CIC-BCCC-NRC TabularIoTAttack-2024 dataset.

Feature Category	Selected Feature	Description	Relevance to Intrusion Detection
Flow Information	Flow Duration	Total time duration of a network flow	Helps identify abnormal communication sessions and prolonged attack traffic
Packet Statistics	Total Fwd Packets	Total number of packets sent in the forward direction	Useful for detecting traffic flooding and packet anomalies
Packet Length	Fwd Packet Length Mean	Average size of forward packets	Assists in distinguishing malicious packet behaviour from normal traffic
Timing Features	Flow IAT Mean	Mean inter-arrival time between packets	Captures temporal communication patterns associated with attacks
Traffic Rate	Flow Bytes/s	Number of bytes transmitted per second	Useful for identifying unusually high traffic volume during attacks
Header Information	Fwd Header Length	Total forward packet header size	Helps detect malformed or suspicious packet structures
Flag Features	SYN Flag Count	Number of SYN flags observed in the flow	Important for detecting SYN flood and DoS attacks

(Continued)

Table 2 (continued)

Feature Category	Selected Feature	Description	Relevance to Intrusion Detection
TCP Window Features	Init Fwd Win Bytes	Initial TCP window size in the forward direction	Indicates abnormal TCP session behaviour
Active/Idle Features	Active Mean	Average active time before idle state	Useful for modelling traffic activity behaviour
Label Information	Attack_Type	Attack category associated with the traffic flow	Used for identifying specific attack scenarios

To evaluate the model's generalization capabilities on data outside the training distribution, an independent dataset of benign network traffic captured from unseen PCAP files was passed through the same feature extraction pipeline using CICFlowMeter. The dataset is completely benign and was used to evaluate the false positive behaviour of the trained IDS models in a real-world deployment scenario.

3.3 Data Preprocessing

To increase the quality and stability of the acquired dataset of the CIC-BCCC dataset, various preprocessing actions have been performed. By using Python Pandas, many different CSV files were concatenated into a single data-frame. Missing/infinite values as well as duplicate entries were removed during the cleaning of the dataset. Non-numeric features were dropped in case the models required only numeric attributes. Ahmad et al. [9] underline that such data cleaning techniques are vital to minimise the amount of noise in the data and thus ensure model robustness and consistency. Binary classification labels were assigned to benign traffic as '0' and attack traffic as '1', and only network flow attributes of a numeric nature relevant to traffic analysis were preserved. From the inspection of the dataset, it was observed that there were no duplicate records and no missing values in the dataset after preprocessing. The labels are assigned as binary 0 and 1 for benign and attack, respectively. After preprocessing, the resultant dataset was used with 79 numerical features related to flow to build the model. As stressed in Buda et al. [35] to prevent information leakage, all procedures aimed at balancing class imbalance (including SMOTE over-sampling) were applied solely to the training subset. The held-out test set was never used before model training and testing.

3.4 Proposed System Architecture

The architecture of the proposed AI-based intrusion detection system to tackle the issues of bias and false positives under extreme class imbalance is presented in Fig. 1. The framework involves the integration of seven different modules, including the collection of a dataset, preprocessing of data, model training, model validation, explainability of model, validation by a third party, and deployment for real-time purposes. During preprocessing, the duplication of the rows and absence of values are removed, the essential features are identified, labels are converted, and datasets are separated through stratification to train 80% of the data while testing it against the rest 20% without disturbing the prior class balance. Afterwards, the three different types of Random Forest learning methods, such as baseline Random Forest, SMOTE Random Forest, and cost-sensitive RF, were trained under precisely similar environmental conditions. Random Forest classifier was used for its well-established reliability in IDSs, particularly for high-dimensional data such as network traffic and its ability to provide feature importance along with SHAP interpretability for further analysis. Due

to the robust ensemble methods of Random Forest to combat overfitting, and its feature selection capabilities, it is conducive to analysis of the effects of class imbalance reductions to model efficiency, interpretability, and stability to implement for any environment. Hence, in order to study these imbalance handling, a standard, reproducible machine learning structure would have to be utilized.

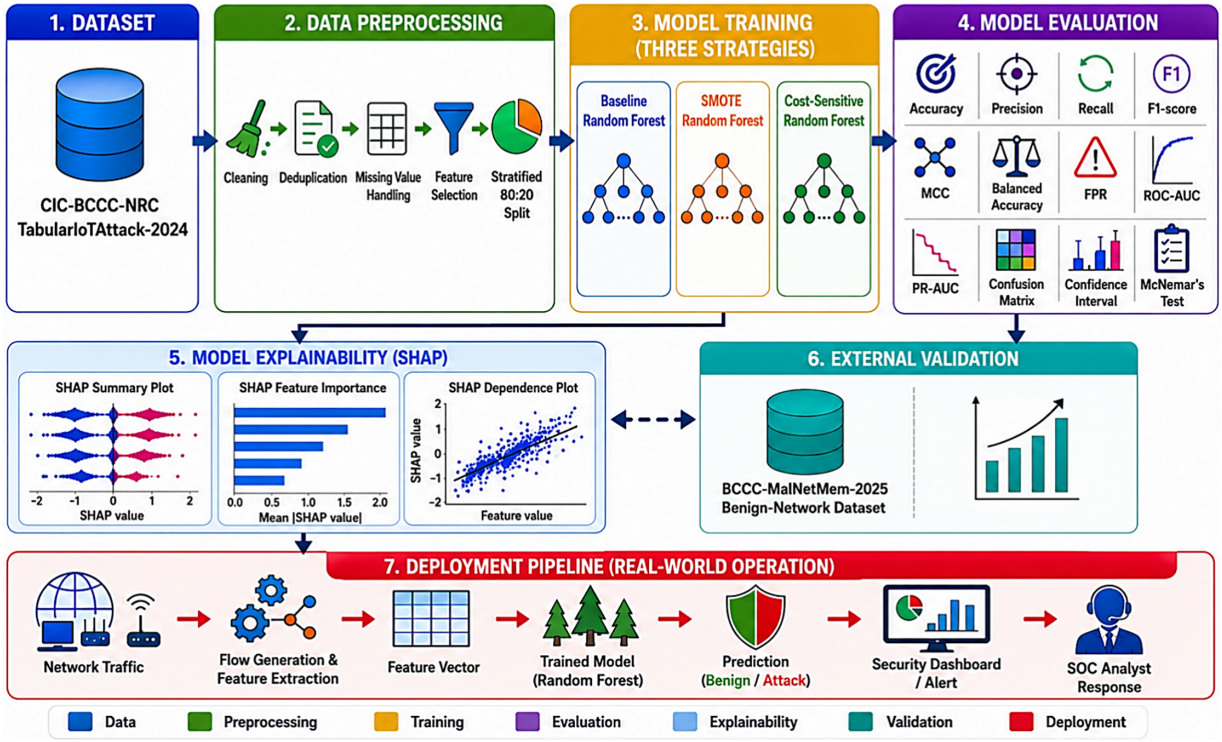

Figure 1: Proposed operational fairness-aware AI-based intrusion detection architecture.

3.5 Imbalance Mitigation Techniques

Three different modeling strategies were considered: The first was the baseline model developed with the original imbalanced data set (as a control). In the second approach, the SMOTE was applied, which balances the dataset by synthesising artificial samples through interpolation between neighbouring minority instances, to generate synthetic minority instances and reduce the bias introduced by the severe class imbalance while preserving the characteristics of the majority class [36]. This oversampling only considered the training dataset in order to avoid the loss of information. For the purpose of the comparison within the cost-sensitive learning study, SMOTE was only used as another means to address the class imbalance; it was part of a similar, consistent experimental setup with no intention of creating real synthetic benign traffic for operational use. Rather, the goal of this comparison is to assess how the different means of imbalance prevention affect classifier performance, interpretability, positive false alarms, and robustness in an operational setting; therefore, SMOTE is not presented here as a desirable deployable classifier for IDS. The third technique is the implementation of cost-sensitive learning using the inverse class weighting method, which aims to increase the misclassification costs of the minority instances while maintaining the original data distribution. In this way, the RF classifier attempts to emphasize the minority instance discovery [35] by assigning class weights inversely proportional to the class frequency. To ensure a balanced comparison of results, the same hyperparameters were applied to both the baseline model and the SMOTE-based and cost-sensitive models for Random Forests. The only difference in the three experimental setups was how they dealt with the dataset imbalance.

3.6 Machine Learning Model

The Random Forest classifier was selected because it provides significant benefit in the cybersecurity context due to its ability to handle cybersecurity datasets, resistance to overfitting, scalability on huge data, and ability to handle high-dimensional data features. It has been widely used in IDS research, achieving outstanding classification performance and robustness with noisy data [12]. An RF classifier was used with 100 decision trees, bootstrap aggregation, random selection of features at each node, and split criteria such as entropy. All the other hyperparameters were the same for the three experimental models shown in Table 3. The baseline model was trained on the original dataset while the SMOTE and cost-sensitive models used SMOTE and cost-sensitive approaches during the training. The trained models were saved and subsequently reused for explainability analysis, external validation, and deployment evaluation.

Table 3: Hyperparameter settings for the random forest classifier.

Parameter	Value
Trees	100
Criterion	Entropy
Bootstrap	True
Max Features	Sqrt
Random State	42
Class Weight	None/Balanced
n_jobs	−1

The selected hyperparameters were maintained across all three experimental models to ensure a fair comparison, with the exception of the class weighting strategy applied in the Cost-Sensitive Random Forest model.

3.7 Performance Evaluation Metrics

Performance evaluation of security systems depends on a number of factors to understand the model's efficiency in attack detection predictions. Model performance was evaluated using the following metrics: accuracy, precision, recall, F1-score, Matthew's correlation coefficient (MCC), balanced accuracy, false positive rate (FPR), receiver operating characteristic area under the curve (ROC-AUC), and precision-recall area under the curve (PR-AUC). Due to the extreme class imbalance in the dataset, the metrics MCC, balanced accuracy, PR-AUC, and FPR are highlighted as important because they provide a more reliable evaluation. An important measure within security is the False Positive Rate (FPR), which essentially indicates the amount of benign traffic that is identified as attack traffic. It is important to note that false positive rates can decrease overall trustworthiness in the security system [22]. Furthermore, statistical significance of observed performance differences of competing models was analyzed using confidence intervals (95%) and the McNemar test to ascertain if differences were due to the model's characteristics instead of random chance. The performance analysis was supplemented with visual examination of classification outputs through the construction of confusion matrices, receiver operating characteristic (ROC) curves, and precision-recall (PR) curves, which visualize classification accuracy and represent the trade-off between detection and false positive rates. This research aims to estimate operational fairness by comparing the performance of a set of class-imbalance mitigating techniques to that of biassed classification under class-imbalance conditions; thus, operational fairness is understood through the measurement of false positive behaviour reduction, balanced accuracy, Matthew's correlation coefficient, explainability, and external validation.

3.8 Explainability Analysis

SHAP analysis was also used to identify features that affect machine learning models and their corresponding impact. The model decision behaviour was analyzed using SHAP, and it assisted the team in providing interpretability in machine learning models fairly. For the sake of comprehending the model's global behaviour, SHAP analysis using global feature importance, summary plots, dependence plots, and feature interaction analysis, as well as interpreting the local prediction behaviour, were explored. The explainability for the most successful Random Forest model to determine the most discriminatory network flow features was applied and has enhanced explainability in the decisions made by the intrusion detection model. SHAP is widely recognized in the field of interpretable machine learning, as defined by Lundberg et al. [37].

3.9 External Validation Procedure

To assess the generalizability of the proposed intrusion detection framework beyond the benchmark dataset, only Cost-Sensitive Random Forest, selected as the final deployment model following the comparative evaluation, was externally validated using an independent benign network traffic dataset. The external dataset contained 2250 network flow samples generated from 15 packet capture (PCAP) files (tcpdumpUTG1 to tcpdumpUTG15) taken from the BCCC-MalNetMem-2025 dataset. Each original PCAP file was pre-processed using CICFlowMeter to convert raw network packets to bidirectional flows and extract statistical flow features that can be readily used by machine learning-based IDS [38]. The pre-processing chain used for the model development, which included feature selection, imputation, non-feature column removal, and same formatting of features, was applied to the external dataset.

The same features were passed to the models for prediction as the ones used during training and external validation, and no SMOTE or any other form of up/down sampling was employed during external validation to ensure realistic performance simulation on real-world data. The external validation set contained purely benign network traffic. Since no correlated labeled attack traffic can be sourced from the BCCC-MalNetMem-2025 benign collection (as they are collected in different environments or are not made available to researchers), this external validation serves to evaluate the models under more realistic circumstances by primarily focusing on false positive performance—excessive false alarms negatively impact network administrators' ability to work and diminish trust in AI-driven intrusion detection solutions. However, further real-world external validation utilizing labeled attack traffic collected from various network environments would better understand the detection capacity and generalizability of the trained intrusion detection models [39].

3.10 Experimental Environment

The experimental implementation of this study was performed using Python 3.12 and multiple scientific computing and machine learning libraries within a Jupyter Notebook based on the Anaconda Python distribution. Python was chosen for the experimentation process because of its flexibility, extensibility and large community support in cybersecurity and AI research [9]. Pandas 2.2 and NumPy 2.2 were used for data manipulation, pre-processing and computations and also for storing and operating large-sized tabular datasets. Scikit-learn 1.6 was used for machine learning implementation like data splitting. The imbalanced-learn library was used for SMOTE oversampling to overcome the heavy class imbalance of the dataset. SHAP version 0.52.0 was used to produce model explainability and feature importance, thus promoting transparency and operation fairness analysis of the model. Matplotlib 3.10 was used to plot experimental results such as feature importance plots, confusion matrices, and class distribution plots. The experimental system facilitated rapid dataset processing, repeatable experimentation, explainable AI and scalable implementation of the presented operation fairness intrusion detection system. The datasets used were also public cybersecurity ones, and no personal or confidential human data was used. To conduct

further explainability analysis and external validation experiments as well as some initial deployment experiments, Joblib was used to serialize the trained Random Forest models. Experiments were carried out on a Windows workstation with an Intel Core i5 multi-core processor and 12 GB RAM to conduct a flow-based, large-scale intrusion detection experiment.

3.11 Deployment Framework

For deployment in real-time network monitoring, the proposed intrusion detection system architecture is intended to operate using a flow-based processing pipeline. Input network packets are captured and aggregated to network flows, and the feature extracted from flows is subjected to the same preprocessing pipeline as was applied to the training dataset to ensure consistency between the training and operating environments. The feature vectors thus produced by the pipeline are subsequently used to classify flows with the trained Random Forest models generated using baseline, SMOTE, and cost-sensitive learning approaches. By using flow-based intrusion detection, a large-scale monitoring approach can be achieved in a scalable and effective way with a high detection capability [21].

Considering the experimental outcomes, it can be concluded that Cost Sensitive Random Forest is best suited for the production environment. Compared with the other techniques, it provides a strong trade-off between overall classification performance and attack detection, achieving the fewest false negatives under the extreme class imbalance condition. The prediction made by the models is sent to the security monitoring tool to generate an alert, and the decision process can be clearly understood and trusted due to the implementation of SHAP based explanation. It is advisable to regularly retrain the models using newly collected labelled traffic to overcome issues of concept drift and pattern changes that can affect their long-term performance [40].

3.12 Threats to Validity

This section describes threats to the validity of proposed methodology in AI-based intrusion detection. The internal validity is discussed in terms of a common preprocessing pipeline and the same hyperparameters of RF to reduce imbalance, while acknowledging that performance may vary slightly due to hyperparameter settings and datasets. Construct validity concerns more reliable evaluation than accuracy (e.g., wide set of evaluation metrics such as precision, recall and MCC) and transparency (e.g., SHAP explainability). Recently, the importance of multi-metrics for trustworthy AI in cybersecurity has been highlighted [41]. External validity was assessed by externally validating the selected Cost-Sensitive Random Forest on an independent benign traffic dataset (BCCC-MalNetMem-2025). Therefore, the external validation findings are limited to this deployment model and should not be generalized to the Baseline Random Forest or SMOTE Random Forest. However, the challenges include the lack of diversity in the representation of cyberattacks and the dynamic nature of attacks. It is recommended to continuously validate using additional datasets and updating deployed models [42].

The credibility of conclusions is the credibility of statistical inferences from results of the experiment. It is based on statistical tests and confidence intervals. The results are important but limited to the data sets and methods applied and need further validation in other models and contexts [43].

4 Results and Discussion

This section describes the results of the experiments conducted using the base, SMOTE, and cost-sensitive reweighted intrusion detection models. This section also examines the classification accuracy, false positive performance, feature importance analysis, explainability, and real-world performance in highly imbalanced scenarios.

4.1 Model Performance Evaluation

Table 4 shows the comparison results of baseline Random Forest, SMOTE Random Forest and Cost-Sensitive Random Forest in the extreme class imbalance situation for flow-based intrusion detection. The accuracy of all three models was nearly perfect, with a classification accuracy of greater than 99.88%. The baseline Random Forest achieved the highest total accuracy, F1-score, and Matthew's correlation coefficient, with values of 99.9278%, 99.9635%, and 0.9640, respectively, indicating high-quality predictions and a strong alignment between predicted and actual classes, as evidenced by a high ROC-AUC of 0.999642 and a PR-AUC of 0.999993. The SMOTE Random Forest gained the highest specificity (98.8652%), the lowest false positive rate (1.1348%), and the best-balanced accuracy (99.3810%). However, it showed a slightly lower MCC (0.9459) because its performance on negative samples decreased compared to the baseline model, even though it still achieved good detection performance in efficiently differentiating attacks from normal samples. The Cost-Sensitive Random Forest attained accuracy (99.9246%), precision (99.9722%), recall (99.9516%), F1-score (99.9619%), and MCC (0.9622) close to other models with a slightly lower false positive rate (2.7603%) than the SMOTE Random Forest but provided overall stable performance. All three models performed extremely well in classification even at a high level of class imbalance, with different operational characteristics depending on which strategy was used to mitigate class imbalance. The Baseline Random Forest had the best overall predictive performance and allows for a workable notion of a model that balances two operational extremes of known detection power, while the SMOTE Random Forest had the best-balanced accuracy and lowest false positive rate, indicative of better separation between attack and benign traffic. Given the very high imbalance of power in the dataset, the classifier performance measures were interpreted using precision, recall, false positive rate, balanced accuracy, MCC, and PR-AUC rather than accuracy. In fact, the statistical testing (Section 4.7) shows that two of the three pairwise classifier comparisons are statistically significant, and there is no statistically significant difference between the baseline Random Forest and Cost-Sensitive Random Forest (McNemar's, $p = 0.1665$).

Table 4: Performance evaluation results.

Metric	Baseline RF	SMOTE RF	Cost-Sensitive RF
Accuracy	99.9278	99.8866	99.9246
Precision	99.9784	99.9886	99.9722
Recall	99.9487	99.8969	99.9516
F1-score	99.9635	99.9427	99.9619
Specificity	97.8531	98.8652	97.2397
False Positive Rate	2.1469	1.1348	2.7603
Balanced Accuracy	98.9009	99.3810	98.5956
MCC	0.9640	0.9459	0.9622
ROC-AUC	0.999642	0.999847	0.999787
PR-AUC	0.999993	0.999998	0.999996

4.2 The Confusion Matrix of Three Models

The confusion matrices for the Baseline Random Forest, SMOTE Random Forest, and Cost-Sensitive Random Forest on the 80:20 stratified test set are shown in Fig. 2. All of the three classifiers provided satisfactory performance but varied in their responses to false positives and false negatives during the extreme class imbalance situation. The Baseline Random Forest correctly classified 6381 benign flows and 648,373 attack flows with 140 false positives and 333 false negatives, achieving an accuracy of 99.9278%. The

SMOTE Random Forest recorded the least number of false positives (74) and classified 6447 benign flows; therefore, it performed with the highest specificity.

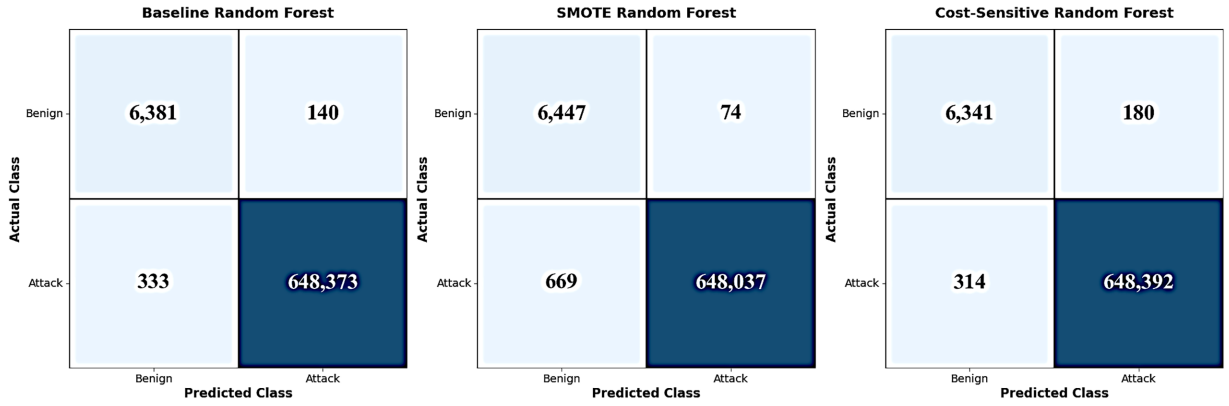

Figure 2: The confusion matrix of three models.

Nevertheless, it also experienced an increased number of false negatives (669), hence leading to reduced recall, although balanced accuracy was high. Cost-Sensitive Random Forest identified attack cases with 648,392 correct classifications for attacks, and it recorded the fewest false negatives (314) and the highest false positives (180). Therefore, from the confusion matrix, the performance differences between three algorithms with three methods of imbalance resolution are clearly demonstrated with respect to three different aspects, with Baseline Random Forest creating balance between recall and precision, SMOTE Random Forest having minimum false alarms, and Cost-Sensitive Random Forest being successful in identifying attacks. The confusion matrices were generated from the complete stratified test partition without additional subsampling. The study's outcome demonstrates that there should be a proper understanding and balance of false alarms and identification rates of attacks to select a specific method for developing an intrusion detection system that is practical and realistic for cybersecurity applications.

4.3 The AUC-ROC Curves of Three Models

As Fig. 3a indicates, the three Random Forest models, Baseline, SMOTE, and Cost-Sensitive, have excellent classification performance, with ROC curves tending towards the upper-left corner. It indicates they have a strong separation between attack and benign traffic. With an ROC-AUC value of 0.999847 for SMOTE Random Forest, followed by Cost-Sensitive (0.999787) and Baseline (0.999642), all models nearly achieve a perfect discrimination. To zoom in on the region of low false positive rate, the result is shown in Fig. 3b. The SMOTE model shows a slightly better performance in this range, and the performance gap between the models is reduced. The ROC values of the three models all have true positive rates higher than 99.7% when the false positive rate is below 3%, which shows that all three models can resist class imbalance very well.

Finally, based on the results of the ROC curve, all three Random Forest models perform well in classification for the imbalanced data. ROC can be optimistic under severe class imbalance and therefore PR analysis is also presented Although the SMOTE model has the highest ROC-AUC value, the difference between its ROC-AUC and those of the Cost-Sensitive and Baseline models is very small. Hence, each of them can reduce the class imbalance problem to have a positive intrusion detection effect. When selecting models in real applications, consider not only ROC-AUC but also other indices like False Positive Rate,

Matthew's Correlation Coefficient (MCC), Balanced Accuracy and Precision-Recall AUC for imbalanced cybersecurity data.

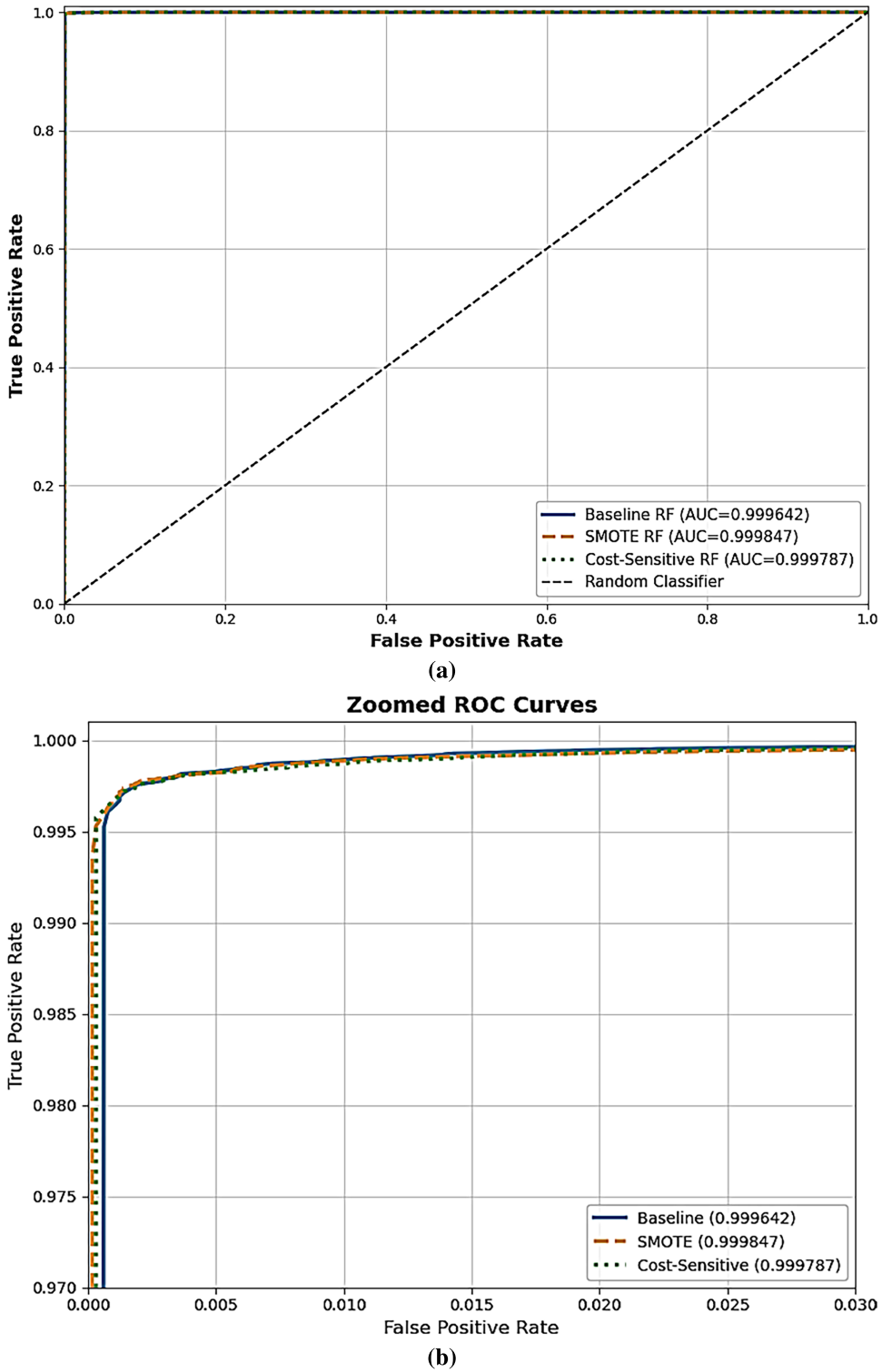

Figure 3: (**a**): AUC-ROC curves of three models. (**b**): Zoomed AUC-ROC curves of three models.

4.4 Precision Recall Curve Analysis

As shown in Fig. 4, the baseline Random Forest, SMOTE Random Forest, and Cost-Sensitive Random Forest PR curves also indicate similar top performances around the top right corner of the graph as expected, where both metrics are close to their optimum (meaning high precision and high recall). In this context, the SMOTE Random Forest shows the highest value in terms of PR-AUC (AP = 0.999998) and also outperforms both Cost-Sensitive (AP = 0.999996) and Baseline Random Forest (AP = 0.999993). All curves remain fairly flat until recall reaches close to 1.0, meaning a low decrease in precision while detecting attacks. This highlights the positive ability of the Random Forest methods to work with class imbalanced intrusion data. Moreover, these excellent PR-AUC values confirm that the designed framework efficiently detects malicious traffic while minimizing false alarms in a heavily class imbalanced setup.

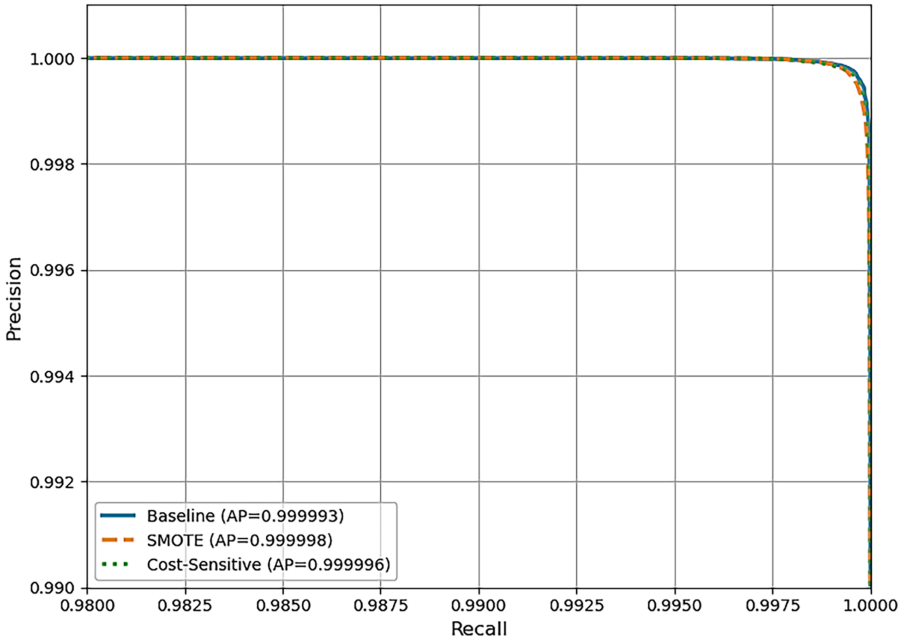

Figure 4: Precision recall curves.

4.5 Feature Importance Analysis

Fig. 5 compares global feature importance across the three models using mean absolute SHAP values. The baseline Random Forest shows relatively small absolute contributions in all the features, and these are relatively distributed among predictors. In contrast, the SMOTE Random Forest model places the greatest emphasis on packet size-related features, such as Packet Length Mean, Fwd Packet Length Mean, and Packet Length Max, as well as ACK Flag Count; notably, the latter is significantly more sensitive to these packet size characteristics compared to the baseline model. Packet Length Mean determines the difference in packet size patterns between benign and malicious flows, while ACK Flag Count indicates the malicious TCP acknowledgement behaviour associated with denial-of-service and flooding attack types. The Src Port and Flow Bytes/s features reveal that the cost-sensitive model utilises protocol patterns along with traffic volume, thus representing the malignant patterns in a better-balanced way.

The Cost-Sensitive Random Forest also reveals high importance for ACK Flag Count, which, in order of importance, is followed by Src Port, Flow Bytes/s, PSH Flag Count and Bwd Init Win Bytes, suggesting protocol behaviour and flow statistics play a much more significant role in distinguishing between

malicious and normal traffic. Therefore, despite achieving promising results with respect to predictive performance, the applied imbalance-reduction methods provide a considerable differentiation regarding relative feature importance. Cost-Sensitive Random Forest achieved a relatively more balanced feature importance distribution while maintaining satisfactory interpretability of the models. In conclusion, SHAP feature importance illustrates that the model performance of classifiers changes due to different imbalance mitigation techniques, and their decision-making processes have also been altered. Using Cost Sensitive Random Forest allows us to make use of the properties of the protocol, traffic flow and time features in a more balanced way, achieving a relatively high level of predictive performance with more explainability.

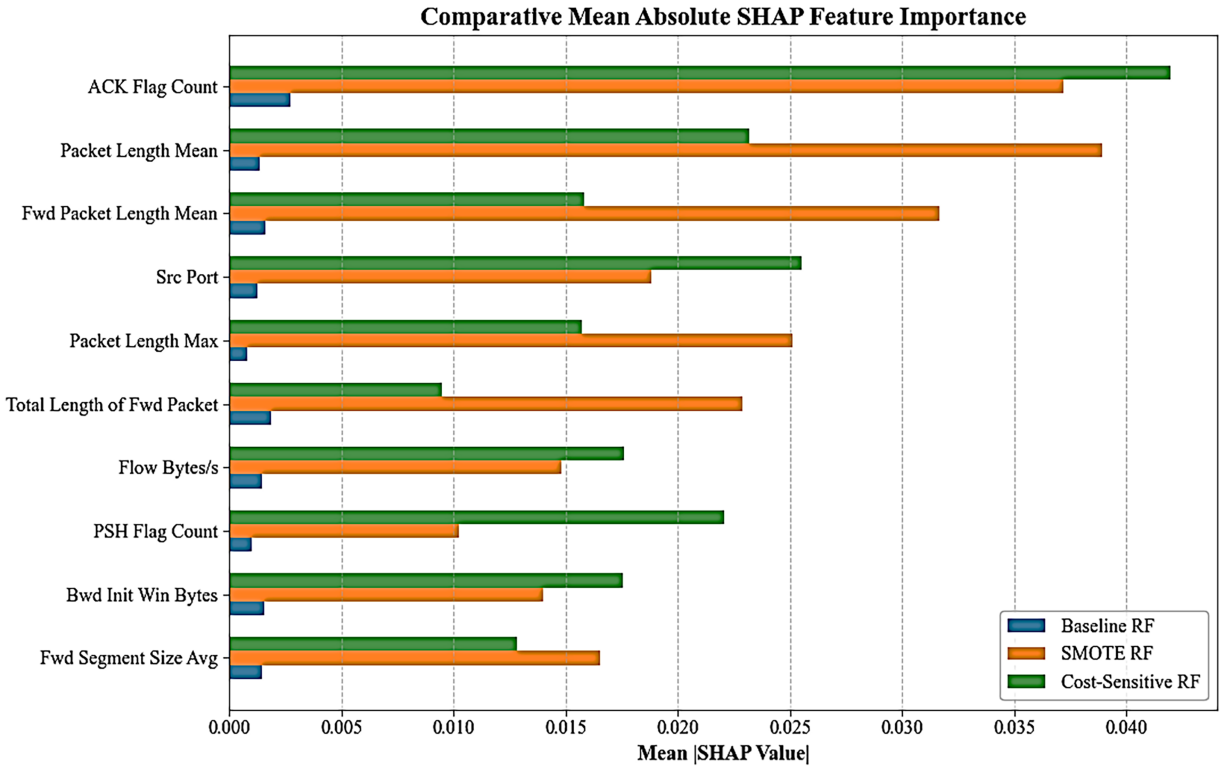

Figure 5: Comparative mean absolute SHAP feature importance.

4.6 SHAP Explainability Analysis

To improve transparency and to produce fair interpretations, SHAP (SHapley Additive Explanations) analysis was performed to understand the influence of each feature in the predictions of Random Forest models. In Fig. 6, SHAP summary plots are produced for the baseline Random Forest, the SMOTE Random Forest and the Cost-Sensitive Random Forest, displaying the most important features' contributions towards intrusion detection. The Baseline Random Forest primarily relies on temporal flow characteristics, indicating that packet arrival timing serves as the dominant indicator for distinguishing malicious from benign traffic. i.e., Fwd IAT Mean, Flow Duration, Fwd IAT Min, Flow IAT Min, and ACK Flag Count, to classify intrusion traffic. It appears that arrival times of traffic and traffic flow behaviour were the most important variables driving intrusion decision-making. After synthetic oversampling, the SMOTE Random Forest shifts its attention toward packet size related characteristics. This suggests that balancing the training data enables the classifier to exploit structural packet attributes that may otherwise be overshadowed by the original class distribution. Similarly, the Cost Sensitive Random Forest demonstrates a more distributed attribution pattern across protocol, packet, and traffic flow features, indicating that the model derives its decisions

from complementary network characteristics rather than relying heavily on a single feature group. In this approach, the contributions of the most important features for making decisions seem to spread over protocol-level, packet-level, and traffic flow-based variables, indicating a balanced contribution across the entire set of network traffic features. As observed from the SHAP plots, all three imbalance mitigation approaches offer different patterns of feature attributions; however, the results remain interpretable across various strategies. In summary, the SHAP explainability study revealed that all three imbalance mitigation strategies develop varied decision behaviours despite obtaining equivalent performance metrics. Among them, Cost Sensitive Random Forest presents the most robust and least changing patterns of feature importance, which confirms that imbalance mitigation at the algorithm level yields the optimal balance of performance, robustness, and explainability.

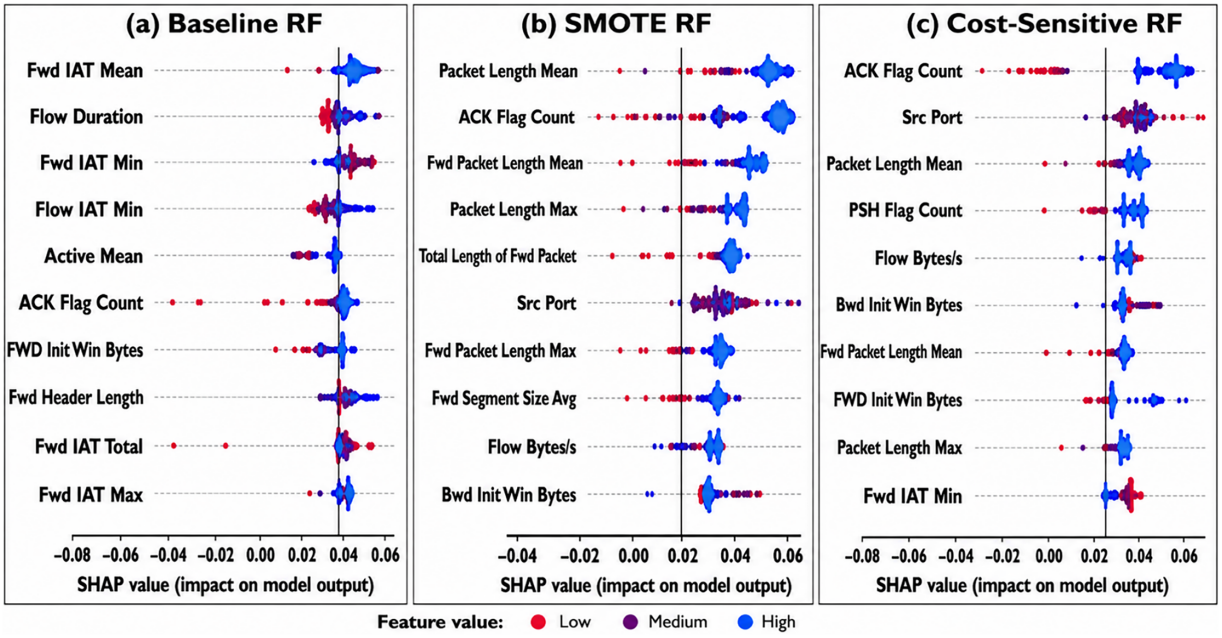

Figure 6: Comparative SHAP summary (Beeswarm) plots.

SHAP summary plots provide additional insights into both the direction and magnitude of feature contributions to the intrusion detection predictions. A positive SHAP value increases the probability that a network flow is predicted as malicious, while a negative value will bias the prediction toward the normal class. Features with higher absolute SHAP values will have more influence on the model predictions, while those with SHAP values centered close to 0 will have little impact. Also, the range of the SHAP values indicates some measure of model stability. The SHAP values in Cost-Sensitive Random Forest have a narrower more clustered and consistent spread for the top contributing features, indicating a more consistent and less sensitive decision behaviour of the model. In contrast, in the SMOTE model, the spread of the packet-level features is larger, indicating that feature contributions may be more variable after synthetic oversampling. Between the two models, the Baseline Random Forest depends mostly on temporal features in the network flow data with moderately variable contributions for some features. The overall findings of the SHAP analysis provide strong support that the chosen imbalanced data sampling techniques affect the overall accuracy of the model but, more interestingly, affect how the model's decision process is constructed.

4.7 Statistical Validation

Table 5 presents the 95% confidence intervals for the classification accuracy of each Random Forest model. The 95% confidence intervals were generated to assess the statistical validity of the trained models. From Table 5, the stability of the achieved accuracies (using the size of the respective confidence intervals) can be seen. The confidence intervals achieved are very small, which signifies very little sampling variation across the dataset used for evaluation. The Baseline Random Forest has an accuracy of 99.9278%, which has a 95%CI of [99.9213; 99.9343], while the cost sensitive Random Forest gained a comparable accuracy of 99.9246% (95%CI: [99.9180; 99.9313]). SMOTE Random Forest is at 99.8866% (95% CI: [99.8785, 99.8948]). The first two confidence intervals are very close to one another, meaning their performance was similar, but SMOTE could not surpass that level.

Table 5: 95% Confidence intervals of the random forest models.

Model	Accuracy (%)	95% Confidence Interval
Baseline Random Forest	99.9278	99.9213–99.9343
SMOTE Random Forest	99.8866	99.8785–99.8948
Cost-Sensitive Random Forest	99.9246	99.9180–99.9313

Furthermore, to investigate whether the difference between the classifiers could be found to be statistically significant, the McNemar's test for each individual pairwise comparison is shown in Table 6. The result demonstrated that there was a statistically significant difference between the Baseline Random Forest and the SMOTE Random Forest ($\chi^2 = 162.975$, $p < 0.001$). There was also a statistically significant difference between the SMOTE Random Forest and the Cost-Sensitive Random Forest ($\chi^2 = 132.267$, $p < 0.001$). However, the difference between the Baseline Random Forest and the Cost-Sensitive Random Forest was not statistically significant ($\chi^2 = 1.914$, $p = 0.1665$). Thus, there is a statistically significant difference between two of the pairwise comparisons, while the Baseline Random Forest and Cost-Sensitive Random Forest have statistically equivalent classification performance. This observation is consistent with the overlapping confidence intervals and suggests that the Cost-Sensitive Random Forest preserves the predictive behaviour of the baseline model while incorporating cost-sensitive learning to address severe class imbalances.

Table 6: McNemar's test results for pairwise model comparisons.

Model Comparison	χ^2 Statistic	*p*-Value	Decision
Baseline RF vs. SMOTE RF	162.975	<0.001	Significant
Baseline RF vs. Cost-Sensitive RF	1.914	0.1665	Not Significant
SMOTE RF vs. Cost-Sensitive RF	132.267	<0.001	Significant

4.8 External Validation

To evaluate the practical robustness of the proposed intrusion detection framework, the trained Cost-Sensitive Random Forest model was externally validated using 2250 previously unseen benign network flows collected from 15 PCAP files (tcpdumpUTG1 to tcpdumpUTG15) within the BCCC-MalNetMem-2025 dataset. The Cost-Sensitive Random Forest was selected for external validation because it demonstrated the most appropriate balance between predictive performance, operational robustness, and deployment suitability during the comparative evaluation. The original packet capture files were converted into bidirectional network flow records using CICFlowMeter, after which the identical preprocessing pipeline employed during model training was applied before prediction. The external validation results are presented in Table 7, while

the corresponding confusion matrix is shown in Table 8. The external evaluation correctly classified 1721 benign flows as benign (True Negatives) and incorrectly classified 529 benign flows as malicious (False Positives), resulting in an external False Positive Rate (FPR) of 23.51% with a 95% Wilson confidence interval of 21.80% to 25.31%. This value is much larger than the internal benchmark FPR of 2.76%, indicating that the good performance obtained in benchmark testing does not fully translate to real-world deployment. The observed increase in false alarms indicates a significant generalization gap due to the differences between the benchmark dataset and the previously unseen operational network traffic. Although the external validation dataset consisted entirely of benign traffic and therefore could not be used to evaluate attack detection capability, it provides a realistic assessment of the model's susceptibility to false alarms during operational deployment. High false positive rates increase alert fatigue, consume valuable analyst resources, and reduce confidence in automated IDS. Consequently, these findings demonstrate that benchmark evaluation alone is insufficient for assessing operational reliability. Independent external validation using previously unseen network traffic should therefore be considered an essential component of the evaluation framework for AI-based IDS intended for practical cybersecurity deployment.

Table 7: External validation performance.

Metric	Value
External Dataset	BCCC-MalNetMem-2025
Number of Benign Samples	2250
True Negatives (TN)	1721
False Positives (FP)	529
External False Positive Rate (FPR)	23.51%
95% Wilson Confidence Interval	21.80%–25.31%

Table 8: External validation confusion matrix.

Actual Class	Predicted Benign	Predicted Attack
Benign	1721	529

4.9 Deployment Analysis

Table 9 presents the deployment evaluation of the three Random Forest models and highlights their differing computational efficiencies. The computational efficiencies of the three Random Forest models differ from each other. Cost-Sensitive Random Forest had the highest performance on deployment since it took only 8.34 min to train and 3.04 s to predict and had the highest inference throughput (215,876 flows/s), which was ideal for real-time intrusion detection. Baseline Random Forest had moderate training time (16.40 min), inference throughput (161,143 flows/s), and model size (40.83 MB).

Table 9: Deployment performance comparison.

Model	Training Time (min)	Prediction Time (s)	Throughput (Flows/s)	Model Size (MB)
Baseline RF	16.40	4.07	161,142.96	40.83
SMOTE RF	162.98	3.36	195,051.01	79.81
Cost-Sensitive RF	8.34	3.04	215,876.03	50.70

However, SMOTE Random Forest took a very high computational cost during training (162.98 min) and also produced the largest model size (79.81 MB) while maintaining a satisfactory inference throughput (195,051 flows/s) in Fig. 7a,b. Thus, synthetic oversampling increases training complexity and storage costs. However, cost-sensitive learning produces a model with less computation cost while avoiding synthetic oversampling overhead. So, the best trade-off between computational efficiency and availability is the Cost-Sensitive Random Forest. The reduced prediction time and increased throughput of the Cost-Sensitive Random Forest indicate greater suitability for real-time intrusion detection. The shorter training time observed for the Cost-Sensitive Random Forest is attributed to its weighted learning strategy, which avoids the increase in effective training samples associated with resampling while maintaining the original dataset size, thereby reducing computational overhead during model training.

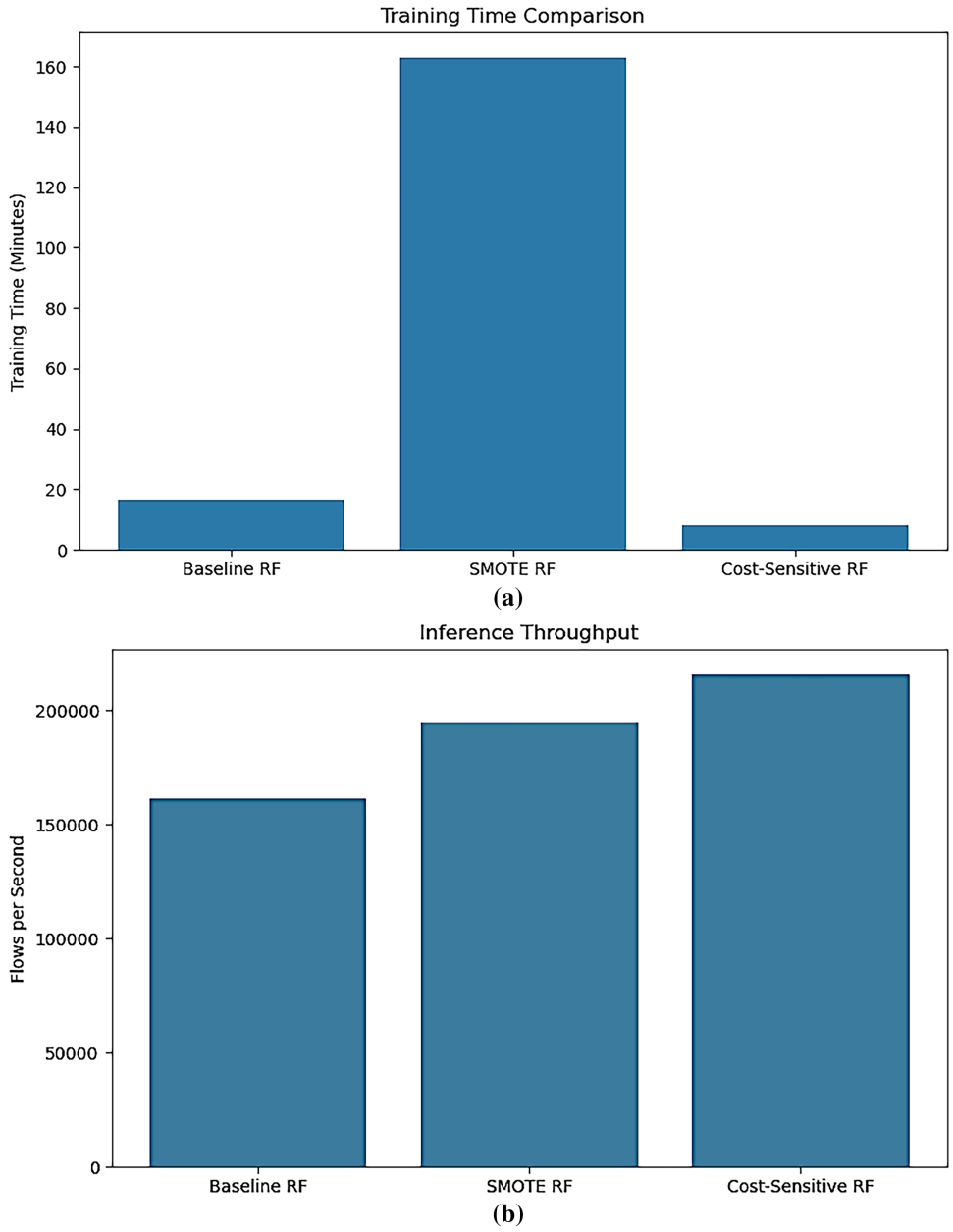

Figure 7: (**a**): Training time comparison. (**b**): Inference throughput comparison.

4.10 Discussion

The experiment results showed that all three Random Forest models exhibited excellent performance on intrusion detection under extreme class imbalance with classification accuracies greater than 99.88%. Nevertheless, the evaluation showed that benchmark accuracy is not sufficient for choosing a fully operational intrusion detection model. The baseline Random Forest performed the highest in overall classification accuracy (99.9278%), F1-score (99.9635%), and Matthew's Correlation Coefficient (MCC = 0.9640), demonstrating its superiority in general prediction ability.

The SMOTE Random Forest, on the other hand, had the highest balanced accuracy (99.3810%), specificity (98.8652%), ROC-AUC (0.999847), and PR-AUC (0.999998). This means that it was better at telling the difference between malicious and benign traffic after synthetic oversampling. The Cost-Sensitive Random Forest obtained comparable results (accuracy = 99.9246%, recall = 99.9516%, F1-score = 99.9619%, MCC = 0.9622) while maintaining a balance between malicious traffic detection capabilities and misclassification risks. These results reinforce that selection of an appropriate technique to handle imbalance should be based on operational deployment objectives, rather than just high performance on a benchmark dataset.

The SHAP analysis demonstrates that feature importance differs across the three Random Forest models. For the Baseline Random Forest, temporal inter-arrival features, particularly Flow IAT Mean and Flow Duration, contributed most strongly to intrusion detection decisions. In contrast, the SMOTE Random Forest assigned greater importance to packet-size related features, while the Cost-Sensitive Random Forest was primarily influenced by ACK Flag Count, Src Port, and Flow Bytes/s. These differences indicate that each imbalance mitigation strategy learns distinct decision patterns while maintaining strong overall predictive performance. SHAP distributions highlighted the variation of model behaviour, where Cost-Sensitive Random Forest produced more focused and stable feature attribution patterns, demonstrating consistent decision-making under severe class imbalance. Statistical validation also assured the trustworthiness of the experimental results. The narrow 95% confidence intervals for different metrics demonstrated the stability of the reported performance. McNemar's test also showed statistical significance of performance differences between the baseline and SMOTE models, as well as between the SMOTE and cost-sensitive models.

External validation revealed a critical finding. When evaluated on 2250 unseen benign network flows from the BCCC-MalNetMem-2025 dataset, the Cost-Sensitive Random Forest incorrectly identified 529 of them as attacks (false positives) while only 1721 benign flows were classified correctly. This resulted in a high external false positive rate of 23.51% (95% Wilson confidence interval: 21.80%–25.31%), which is far greater than the internal benchmark FPR of 2.76%. This indicates a large generalization gap in applying models trained on benchmark data to operational scenarios, as they may have learned dataset-specific patterns that do not reflect real-world network traffic. The generalization gap shows that some highly ranked traffic features may capture benchmark-specific characteristics rather than generalizable attack behaviour. Packet counts and timings are features that can vary quite a bit with the environment and user behaviour. Therefore, some features are effective at discriminating attacks from normal traffic, but it does not mean they are useful in operational networks with different traffic patterns from benchmark conditions. It emphasizes the necessity of considering dataset-specific learnings and the need for external validation and interpretability studies.

In conclusion, the results suggest that the performance of an AI-based intrusion detection system should not be assessed by benchmark performance only but by a holistic approach considering imbalanced learning strategies, explainability, statistical validation, external validation, and operational deployment. This will allow building more trustworthy and resilient AI solutions that can be safely deployed in real-world cybersecurity scenarios.

5 Conclusion

This study investigated the effect of class imbalance mitigation techniques on performance, interpretability, operational fairness, and operational reliability of AI-based IDS under extreme class imbalance conditions. Using the CIC-BCCC-NRC TabularIoTAttack-2024 dataset, three Random Forest learning strategies, namely the baseline model, SMOTE oversampling, and cost-sensitive learning, were systematically evaluated within a comprehensive framework incorporating explainable AI (SHAP), statistical validation, external validation, and deployment analysis. The results demonstrate that class imbalance mitigation substantially influences intrusion detection behaviour, particularly in terms of attack detection capability, false positive behaviour, model interpretability, and operational reliability.

Although all three models achieved excellent classification performance on the benchmark dataset, the results indicate that benchmark accuracy alone is insufficient for selecting an intrusion detection model for practical deployment. The SHAP explainability analysis revealed that temporal traffic characteristics, protocol-related attributes, and packet transmission behaviours consistently played dominant roles in intrusion detection decisions, while the statistical validation confirmed that the observed performance differences were statistically significant. Moreover, external validation on unseen benign network traffic revealed a large generalization gap, as the external false positive rate increased to 23.51% (95% Wilson confidence interval: 21.80%–25.31%) despite strong benchmark performance. This result suggests that models with high benchmark performance can suffer large performance drops when deployed to real operational environments.

Thus, this research makes a significant contribution to the development of an integrated evaluation framework that combines imbalance-aware learning, explainable artificial intelligence, statistical validation, external validation, and deployment analysis into a single intrusion detection framework. Unlike many existing studies, which mostly report benchmark classification accuracy, this work provides a more comprehensive evaluation of model transparency, robustness, and operational reliability. The study proposes an evaluation framework that provides a practical approach to evaluate AI-based IDS for deployment in real-world cybersecurity environments.

The study focuses on the Random Forest classifier, which means the findings should be interpreted within the context of this classifier family. The proposed evaluation framework is applicable to other machine learning models. However, the external validation dataset only included normal traffic, which means it cannot accurately assess the attack detection rate in a real-world working situation. Also, since only the Cost-Sensitive Random Forest was externally validated, the external validation findings should not be generalized to the Baseline Random Forest or SMOTE Random Forest. Future research can include other imbalance mitigation techniques, deep learning and transformer-based intrusion detection models, adversarial robustness, continual learning, operationally fair and trustworthy AI-based IDS, and concept drift adaptation on a real-world, enterprise-level operational network. As cyber threats become more complex and pervasive in size, robust and reliable AI-based IDS need to be evaluated with regard to interpretation, statistical validation, external validation, and deployment readiness apart from baseline performance. The unified evaluation framework designed in this study could further address this challenge in supporting real-world operational networks to implement timely and effective intelligent threat detection for building trustworthy AI-powered cybersecurity solutions.

Acknowledgement: None.

Funding Statement: There has been no funding from any organization or individual in the course of this research. The authors conducted this study independently and did not receive financial support from any external sources. Any expenses incurred during the research were covered by the authors themselves.

Author Contributions: Conceptualization, John Ojo Ajayi and Grace Egenti; methodology, John Ojo Ajayi and Grace Egenti; software, John Ojo Ajayi; validation, John Ojo Ajayi and Grace Egenti; formal analysis, John Ojo Ajayi; investigation, John Ojo Ajayi; data curation, John Ojo Ajayi; visualization, John Ojo Ajayi; writing, original draft preparation, John Ojo Ajayi; writing, review and editing, John Ojo Ajayi and Grace Egenti; supervision, Grace Egenti; project administration, Grace Egenti. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The CIC-BCCC-NRC TabularIoTAttack-2024 dataset and the BCCC-MalNetMem-2025 dataset are publicly available through the Canadian Institute for Cybersecurity. The source code and trained models used in this study are available from the corresponding author upon reasonable request.

Ethics Approval: None.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bengio Y, Clare S, Prunkl C, Andriushchenko M, Bucknall B, Murray M, et al. International AI safety report 2026 [Online]. [cited 2026 Jan 1]. Available from: https://internationalaisafetyreport.org/.
2. Gallegos IO, Rossi RA, Barrow J, Tanjim MM, Kim S, Dernoncourt F, et al. Bias and fairness in large language models: a survey. Comput Linguist. 2024;50(3):1097–179. doi:10.1162/coli_a_00524.
3. Chadha KS. Bias and fairness in artificial intelligence: methods and mitigation strategies. Int J Res Publ Semin. 2024;15(3):36–49. doi:10.36676/jrps.v15.i3.1425.
4. Johnson JM, Khoshgoftaar TM. Survey on deep learning with class imbalance. J Big Data. 2019;6(1):27. doi:10.1186/s40537-019-0192-5.
5. Agbasiere CL, Nze-Igwe GR. Algorithmic fairness in recruitment: designing AI-powered hiring tools to identify and reduce biases in candidate selection. Path Sci. 2025;11(4):5001. doi:10.22178/pos.116-10.
6. Amershi S, Begel A, Bird C, DeLine R, Gall H, Kamar E, et al. Software engineering for machine learning: a case study. In: Proceedings of the 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP); 2019 May 25–31; Montreal, QC, Canada. p. 291–300. doi:10.1109/ICSE-SEIP.2019.00042.
7. Ghosh M. Artificial intelligence (AI) and ethical concerns: a review and research agenda. Cogent Bus Manag. 2025;12(1):2551809. doi:10.1080/23311975.2025.2551809.
8. Apruzzese G, Pajola L, Conti M. The cross-evaluation of machine learning-based network intrusion detection systems. IEEE Trans Netw Serv Manag. 2022;19(4):5152–69. doi:10.1109/TNSM.2022.3157344.
9. Ahmad Z, Khan SA, Shiang WC, Abdullah J, Ahmad F. Network intrusion detection system: a systematic study of machine learning and deep learning approaches. Trans Emerg Telecommun Technol. 2021;32(1):e4150. doi:10.1002/ett.4150.
10. Rahman MM, Shakil SA, Mustakim MR. A survey on intrusion detection system in IoT networks. Cyber Secur Appl. 2025;3(1):100082. doi:10.1016/j.csa.2024.100082.
11. Kocher G, Kumar G. Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges. Soft Comput. 2021;25(15):9731–63. doi:10.1007/s00500-021-05893-0.
12. Maseer ZK, Yusof R, Bahaman N, Mostafa SA, Foozy CFM. Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. IEEE Access. 2021;9:22351–70. doi:10.1109/ACCESS.2021.3056614.
13. Gu J, Lu S. An effective intrusion detection approach using SVM with naïve bayes feature embedding. Comput Secur. 2021;103(3):102158. doi:10.1016/j.cose.2020.102158.
14. Abbas A, Khan S, Ali M. Ensemble learning based intrusion detection system for IoT networks. IEEE Access. 2022;10:84521–36. doi:10.1109/ACCESS.2022.3198456.
15. Abirami MS, Yash U, Singh S. Building an ensemble learning based algorithm for improving intrusion detection system. In: Dash SS, Lakshmi C, Das S, Panigrahi BK, editors. Artificial intelligence and evolutionary computations

in engineering systems. Vol. 1056. Berlin/Heidelberg, Germany: Springer; 2020. p. 1–10. doi:10.1007/978-981-15-0199-9_55.
16. Naz N, Khan MA, Khan MA, Khan MA, Jan SU, Shah SA, et al. A comparison of ensemble learning for intrusion detection in telemetry data. In: Saeed F, Mohammed F, Mohammed E, Al-Hadhrami T, Al-Sarem M, editors. Advances on intelligent computing and data science. Vol. 179. Berlin/Heidelberg, Germany: Springer; 2023. p. 1–12. doi:10.1007/978-3-031-36258-3_40.
17. Ajayi OJ, Sodiya AS, Bagiwa MA, Olowookere TA. A super learner ensemble-based intrusion detection system to mitigate network attacks. In: Proceedings of the 2024 5th International Conference on Data Analytics for Business and Industry (ICDABI); 2024 Oct 23–24; Zallaq, Bahrain. p. 207–12. doi:10.1109/ICDABI63787.2024.10800423.
18. Bedi P, Gupta N, Jindal V. Siam-IDS: handling class imbalance problem in intrusion detection systems using siamese neural network. Procedia Comput Sci. 2020;171(6):780–9. doi:10.1016/j.procs.2020.04.085.
19. Chen W, Almamy DF. XGBoost-driven intrusion detection method: integrating SMOTE-based class imbalance mitigation and multi-phase learning. IAENG Int J Comput Sci. 2025;52(7):2234–47.
20. Larriva-Novo X, Sánchez-Zas C, Villagrá VA, Marín-Lopez A, Berrocal J. Leveraging explainable artificial intelligence in real-time cyberattack identification: intrusion detection system approach. Appl Sci. 2023;13(15):8587. doi:10.3390/app13158587.
21. Arreche O, Guntur TR, Roberts JW, Abdallah M. E-XAI: evaluating black-box explainable AI frameworks for network intrusion detection. IEEE Access. 2024;12(6):23954–88. doi:10.1109/ACCESS.2024.3365140.
22. Ferrara E. Fairness and bias in artificial intelligence: a brief survey of sources, impacts, and mitigation strategies. Sci. 2023;6(1):3. doi:10.3390/sci6010003.
23. Pagano TP, Loureiro RB, Lisboa FVN, Peixoto RM, Guimarães GAS, Cruz GOR, et al. Bias and Unfairness in machine learning models: a systematic review on datasets, tools, fairness metrics, and identification and mitigation methods. Big Data Cogn Comput. 2023;7(1):15. doi:10.3390/bdcc7010015.
24. Zhou H, Zhang J, Luo T, Yang Y, Lei J. Debiased scene graph generation for dual imbalance learning. IEEE Trans Pattern Anal Mach Intell. 2022;45(4):4274–88. doi:10.1109/tpami.2022.3198965.
25. Yang Y, Tang X, Liu Z, Cheng J, Fang H, Zhang C. Diff-IDS: a network intrusion detection model based on diffusion model for imbalanced data samples. Comput Mater Contin. 2025;82(3):4389–408. doi:10.32604/cmc.2025.060357.
26. Islam Z. AI-enabled intrusion detection in enterprise networks: a systematic review of methods, datasets, and evaluation metrics (2018–2026). Am J Interdiscip Stud. 2026;07(1):355–86. doi:10.63125/k4t9f683.
27. Nasir K, Badri SK, Alghazzawi DM, Alghamdi MY, Alkhozae M, Almakky A, et al. HED-ID: an edge-deployable and explainable intrusion detection system optimized via metaheuristic learning. Sci Rep. 2026;16(1):2313. doi:10.1038/s41598-025-32183-8.
28. Asry CEL, Benchaji I, Douzi S, Ouahidi BEL. Enhancing cybersecurity: a high-performance intrusion detection approach through boosting minority class recognition. PLoS One. 2025;20(3):e0317346. doi:10.1371/journal.pone.0317346.
29. Najafimehr M, Zarifzadeh S, Mostafavi S. DDoS attacks and machine-learning-based detection methods: a survey and taxonomy. Eng Rep. 2023;5(12):e12697. doi:10.1002/eng2.12697.
30. Mohammad R, Saeed F, Almazroi AA, Alsubaei FS, Almazroi AA. Enhancing intrusion detection systems using a deep learning and data augmentation approach. Systems. 2024;12(3):79. doi:10.3390/systems12030079.
31. Shanmugam V, Razavi-Far R, Hallaji E. Addressing class imbalance in intrusion detection: a comprehensive evaluation of machine learning approaches. Electronics. 2024;14(1):69. doi:10.3390/electronics14010069.
32. Hozouri A, Mirzaei A, Effatparvar M. A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges. Discov Artif Intell. 2025;5(1):314. doi:10.1007/s44163-025-00578-1.
33. Boateng YJ, Mim NJ, Akhter N, Naha R, Mahanti A, Barros A. Application of AI in cyberattack detection: a review. Sensors. 2026;26(5):1518. doi:10.3390/s26051518.
34. Canadian Institute for Cybersecurity. CIC-BCCC-NRC TabularIoTAttack-2024 dataset [Online]. Fredericton, NB, Canada: University of New Brunswick; 2024 [cited 2026 Jan 1]. Available from: https://www.unb.ca/cic/datasets/.

35. Buda M, Maki A, Mazurowski MA. A systematic study of the class imbalance problem in convolutional neural networks. Neural Netw. 2018;106(7):249–59. doi:10.1016/j.neunet.2018.07.011.
36. Fernandez A, Garcia S, Herrera F, Chawla NV. SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. J Artif Intell Res. 2018;61:863–905. doi:10.1613/jair.1.11192.
37. Lundberg SM, Erion G, Chen H, DeGrave A, Prutkin JM, Nair B, et al. From local explanations to global understanding with explainable AI for trees. Nat Mach Intell. 2020;2(1):56–67. doi:10.1038/s42256-019-0138-9.
38. Sharafaldin I, Lashkari HA, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: Proceedings of the 4th International Conference on Information Systems Security and Privacy; 2018 Jan 22–24; Funchal, Madeira, Portugal. p. 108–16. doi:10.5220/0006639801080116.
39. Pawlicki M, Pawlicka A, Szelest S, Komisarek M, Kozik R, Choraś M. ExpLEA-AIner: proposition and development of the model-driven approach to incorporating Explainable AI in network intrusion detection systems for law enforcement agencies. In: Themistocleous M, Bakas N, Kokosalakis G, Papadaki M, editors. Information systems. Vol. 536. Berlin/Heidelberg, Germany: Springer; 2025. p. 1–15. doi:10.1007/978-3-031-81325-2_24.
40. Shukla AK, Sharma A. Enhancing security resilience: dynamic drift detection in cloud-based intrusion detection using the hybrid model. Int J Appl Math. 2025;38(2s):51–9. doi:10.12732/ijam.v38i2s.70.
41. Kemmerzell N, Schreiner A, Khalid H, Schalk M, Bordoli L. Towards a better understanding of evaluating trustworthiness in AI systems. ACM Comput Surv. 2025;57(9):218–38. doi:10.1145/3721976.
42. Hinder F, Vaquet V, Hammer B. Feature-based analyses of concept drift. Neurocomputing. 2024;600(9–10):127968. doi:10.1016/j.neucom.2024.127968.
43. Khan A, Ramli DA, Rehman MZ, Nawi NM. A hybrid approach: meta-heuristic with optimizing machine learning algorithms for intrusion detection systems. Clust Comput. 2026;29(9):572. doi:10.1007/s10586-026-06382-5.