Dealing with Dataset Class Imbalance for Multi-Class Network Intrusion Detection Systems

Fareeha Ashraf, Muhammad Siraj Rathore
Capital University of Science and Technology, Islamabad.
*Correspondence: fareeha.ashraf@cust.edu.pk, siraj.rathore@cust.edu.pk

Citation | Ashraf. F, Rathore. M. S, "Dealing with Dataset Class Imbalance for Multi-Class Network Intrusion Detection Systems", IJIST, Special Issue pp 545-561, May 2026
DOI | https://doi.org/10.33411/IJIST/1805
Received | March 29, 2026 **Revised |** May 08, 2026 **Accepted |** May 13, 2026 **Published |** May 16, 2026.

Intrusion Detection Systems (IDS) are now more crucial for protecting network environments due to the increasing number of IoT devices. However, class imbalance substantially impacts the performance of IDS, especially in multi-class classification, where some attack classes are prevalent, and others are rare. Our research explores the effects of class imbalance by experimenting with the CICIoT2023 dataset in three distinct classification settings: 2-class, 8-class, and 33-class. Our experiments reveal that, although the binary classification performance is high (accuracy ≈ 0.99), the performance drops in multi-class settings, with accuracy of ≈ 0.85 and macro-F1 of 0.59 in the 8-class scenario, and further drops to accuracy of ≈ 0.76 and macro-F1 of 0.47 in the 33-class setting due to the severe class imbalance. To resolve this problem, we examine SMOTE, class weighting, and a combined SMOTE-ENN method. Our results show that the SMOTE-ENN approach effectively balances classes (up to 0.97 accuracy and macro-F1≈0.96 on the test dataset after training with balanced data) and enhances detection performance for minority classes. By contrast, baseline models have lower macro-F1 and recall for minority classes. These results show that hybrid resampling not only boosts classification accuracy but also the ability to detect rare attacks, thus serving as an effective strategy for robust multi-class IoT intrusion detection systems.

Keywords: Intrusion Detection System (IDS), Internet of Things (IoT), Class Imbalance, SMOTE-ENN, Multi-Class Classification, Machine Learning, CICIoT2023 Dataset, Network Security

Introduction:

The growth of the Internet of Things (IoT) has facilitated the deployment of a wide range of interconnected devices in various applications, including smart homes, healthcare, transportation, and industrial applications. This interconnectedness improves automation and efficiency in these systems but also opens up a larger attack surface, making IoT networks vulnerable to a wide range of cyber threats. The constrained processing power and energy of IoT devices make it essential to provide efficient security measures. As a result, Intrusion Detection Systems (IDS) are a crucial element for network monitoring and anomaly detection in IoT networks [1].

Machine learning (ML) approaches have become increasingly popular in IDS to capture intricate patterns and cope with the dynamic nature of attacks. But the performance of ML-based IDS relies on the availability of realistic and labelled data. The CICIoT2023 dataset has recently been introduced as a holistic dataset for IoT intrusion detection that includes various attack types, protocols, and device interactions that are typical of today's IoT environments [2].

Yet, class imbalance is a pervasive issue in multi-class IDS. In many IoT datasets, such as CICIoT2023, some attack categories are overrepresented, while others are underrepresented, sometimes making up only a minuscule fraction of the dataset. This results in model bias towards frequently occurring classes, and suboptimal detection accuracy for minority but more dangerous attacks. This issue is exacerbated in more complex tasks such as multi-class classification (e.g., 8-class and 33-class scenarios), where the model must learn more intricate decision boundaries [3].

The latest research (2024-2026) has shown that even state-of-the-art machine learning and ensemble models are affected by such class imbalances. For example, hybrid machine learning models, which use ensembles of algorithms like XGBoost, Random Forest, and LightGBM, have maintained excellent overall accuracy (greater than 96%), but are still prone to poor detection of minority attack classes [4]. Likewise, large-scale benchmarking research on CICIoT2023 shows class imbalance has a profound effect on model robustness, requiring tailored data balancing techniques to enhance the detection of all types of attack classes [5].

In recent years, there has been a growing interest in advanced resampling and hybrid data balancing strategies to overcome these challenges. Techniques like SMOTE-ENN and extensions (e.g., ESMOTE) have demonstrated effectiveness through the synthesis of new samples and the removal of noisy samples to enhance class separability and support for minority classes [6]. Moreover, deep learning methods such as generative adversarial networks (GANs) and hybrid convolutional neural networks have been explored to produce synthetic samples, achieving notable performance enhancements with imbalanced IoT data [7].

Moreover, recent studies on encrypted IoT traffic and anomaly detection demonstrate that class imbalance still greatly impacts detection in real-world network environments. Recent work shows that sampling procedures have a significant impact on recall and F1-score, and underlines the role of data pre-processing in class-imbalance-aware intrusion-detection systems [8].

However, current research either limits itself to binary classification or involves highly tuned and controlled experiments, making it challenging to apply to practical multi-class intrusion detection scenarios. Specifically, there is a need for systematic analysis of various class levels with consistent and leak-free approaches. To bridge this gap, this research explores the effect of class imbalance in IoT network intrusion detection using the CICIoT2023 dataset under three scenarios: binary (2-class), coarse multi-class (8-class), and fine-grained multi-class (33-class) classifications.

The main goal of this research is to understand the effect of class imbalance on multi-class IoT intrusion detection and to assess the effectiveness of various approaches to enhance detection accuracy. In particular, this work seeks to:

Examine the impact of class imbalance on detection performance at various levels (2-class, 8-class, and 33-class) in the CICIoT2023 dataset.

Compare the effects of various imbalance handling approaches, such as SMOTE, class weighting, and hybrid SMOTE-ENN.

Examine the impact of hybrid resampling on the detection of minority classes using macro-level metrics like macro-F1, recall, and macro precision.

Offer a leakage-safe and consistent evaluation methodology for comparing imbalance handling techniques.

Novelty and Contributions:

This work presents a number of new elements in the area of class imbalance management in IoT intrusion detection based on the CICIoT2023 dataset. This research presents a consistent and leakage-aware classification evaluation setting for various classification granularities, rather than focusing solely on one of the classification tasks or using an optimal configuration. The main novelties and contributions of this study are:

A holistic assessment of the impact of class imbalance across multiple classification granularities (2-class, 8-class, and 33-class), which are seldom considered simultaneously in CICIoT2023-based works.

Leakage-aware resampling pipeline where all preprocessing and balancing methods are only applied to training data, to ensure a fair and practical performance assessment.

Evaluation of several techniques for class imbalance handling, such as SMOTE, class weighting, and a hybrid of SMOTE and ENN under identical experimental conditions.

Comprehensive evaluation using macro and class-specific performance metrics, which shows the trade-offs between the overall accuracy and the detection of minority classes. An open and systematic approach that supports a fair evaluation of imbalance handling strategies for multi-class IoT intrusion detection.

Literature Review:

In intrusion detection systems (IDS), class imbalance is a basic problem, especially in Internet of Things (IoT) situations where network traffic is naturally distributed unevenly. In these situations, the dataset is dominated by a limited number of classes, and unusual but crucial attack types are still underrepresented. Machine learning models become biased toward majority classes as a result of this imbalance, which makes it difficult to detect minority attacks. According to recent studies, metrics like recall, F1-score, and macro-averaged measures must be utilized to accurately assess model performance under imbalanced conditions, as typical assessment metrics like accuracy are insufficient in such scenarios [9][10]. Recent research (2024–2026) further highlights the rapid evolution of imbalance handling techniques, with increasing focus on hybrid and adaptive resampling approaches for a multi-class IoT intrusion detection system.

CICIoT2023 is now a commonly used benchmark for intrusion detection research due to the rise of contemporary IoT-specific datasets. In contrast to previous datasets, CICIoT2023 is well suited for binary and multi-class classification tasks since it captures a variety of attack kinds, numerous protocols, and realistic IoT traffic patterns [11][12]. This dataset has been used in a number of recent studies to assess machine learning models, showing that although high accuracy can be attained in binary classification, class imbalance and increased classification complexity cause performance to drastically decline in multi-class scenarios [13][14].

Using CICIoT2023, several machine learning and deep learning techniques have been investigated for IoT intrusion detection. While ensemble techniques like Random Forest and

Gradient Boosting have demonstrated excellent performance in binary settings, they find it difficult to sustain performance consistency in multi-class environments, especially for minority classes. Similarly, when an imbalance is not appropriately handled, hybrid machine learning models increase overall accuracy but still struggle to identify uncommon attack types.

Data-level strategies like oversampling and under sampling have been routinely used to reduce class imbalance. One of the most popular methods is SMOTE (Synthetic Minority Oversampling Technique), which creates synthetic samples for minority classes to increase their representation. Research using SMOTE on IoT datasets, such as CICIoT2023 and Bot-IoT, has shown improvements in minority class detection [15][16]. Nevertheless, SMOTE can produce noisy samples close to decision boundaries and is sensitive to parameter selection, which can have a detrimental effect on classification results.

Hybrid resampling techniques have been offered as a solution to these restrictions. Techniques like SMOTE-ENN reduce false positives and increase class separability by combining artificial oversampling with noise reduction using Edited Nearest Neighbors (ENN) [17][18]. Hybrid approaches perform better than standalone oversampling strategies, especially in severely imbalanced multi-class circumstances, according to recent benchmarking studies using CICIoT2023 and related IoT datasets.

Additionally, various hybrid resampling methods for managing class imbalance in IoT intrusion detection have been examined in recent works. On contemporary datasets, such as CICIoT2023 and associated benchmarks, hybrid techniques including SMOTE-ENN, GAN-based resampling, and density-based algorithms have been assessed. Among them, SMOTE-ENN has been repeatedly shown to offer a good compromise between noise reduction and minority class augmentation, resulting in better classification performance. Although sophisticated approaches like GAN-based and adaptive hybrid sampling techniques show encouraging outcomes, they frequently result in increased processing complexity. For multi-class IoT intrusion detection, on the other hand, SMOTE-ENN provides a more effective and reliable approach, especially in situations with severe class imbalance [19].

A number of advanced hybrid and adaptive strategies have been introduced in addition to SMOTE-based methods. By taking class density into account and eliminating noisy samples close to decision boundaries, instance density-based hybrid resampling (IDHR) enhances sample creation and increases performance across several classifiers. Similarly, to improve minority class recognition and overall model performance, hybrid frameworks like HiBBKA integrate resampling with heuristic feature selection [20]. Other methods combine SMOTE with under-sampling to produce balanced datasets while maintaining crucial information, showing better outcomes in IoT intrusion detection tasks [21]. Recent studies have investigated deep learning and generative approaches for addressing imbalances in addition to conventional and hybrid resampling techniques [22]. Minority class detection performance is greatly enhanced by GAN-based adaptive hybrid resampling approaches, which dynamically produce high-quality synthetic samples based on class characteristics [23]. Similarly, in both binary and multi-class IoT intrusion detection scenarios, deep learning-based IDS frameworks that incorporate hybrid sampling techniques like ADASYN-SMOTE and ENN have attained high accuracy and robustness [24].

Hybrid frameworks that combine machine learning models with resampling are among the best methods for managing unbalanced datasets, according to recent surveys. These findings highlight that no single technique consistently outperforms others across all datasets and that a classification model's efficacy is strongly dependent on the resampling strategy selected. Multi-class imbalance handling is further complicated by additional issues, such as idea drift, streaming data, and changing attack patterns, which are identified by current research in IoT and fog computing settings [25].

There are still a number of research gaps in spite of these developments. Much current research is limited in its applicability to real-world intrusion detection systems since they mainly concentrate on binary classification or single-level multi-class scenarios.

Additionally, there are frequently insufficient comparisons between various class granularities (such as 2-class, 8-class, and 33-class), and assessment measures are not routinely used. Furthermore, only a small amount of work has been done with leakage-aware experimental pipelines, in which imbalance handling methods are exclusively limited to training data in order to guarantee equitable assessment. This work offers a systematic evaluation of class imbalance in IoT intrusion detection using the CICIoT2023 dataset in three classification settings: 2-class, 8-class, and 33-class, in order to overcome these restrictions. In order to give a thorough examination of multi-class intrusion detection under imbalanced settings, it analyzes several imbalance control algorithms, such as SMOTE, class weighting, and SMOTE-ENN, within a leakage-aware framework and assesses performance using both overall and class-wise metrics.

Methodology:

This section outlines the dataset, preprocessing, methods to handle class imbalance, experimental setting, and performance evaluation strategy used in this research. The processing pipeline is leakage-aware to enable reliable performance measures.

Dataset Description:

The Canadian Institute for Cybersecurity (CIC) provided the CICIoT2023 dataset, which was used in this research. The combined traffic file, which comprises roughly 712,312 network traffic records with 40 attributes, 39 input features, and one target label, was chosen for the experiment. There were 33 distinct attack types and benign traffic in the first dataset. To illustrate the degree of class imbalance in the dataset, the sample distribution across the original labels is first displayed because these attack labels are very detailed and unevenly distributed in Figure 1.

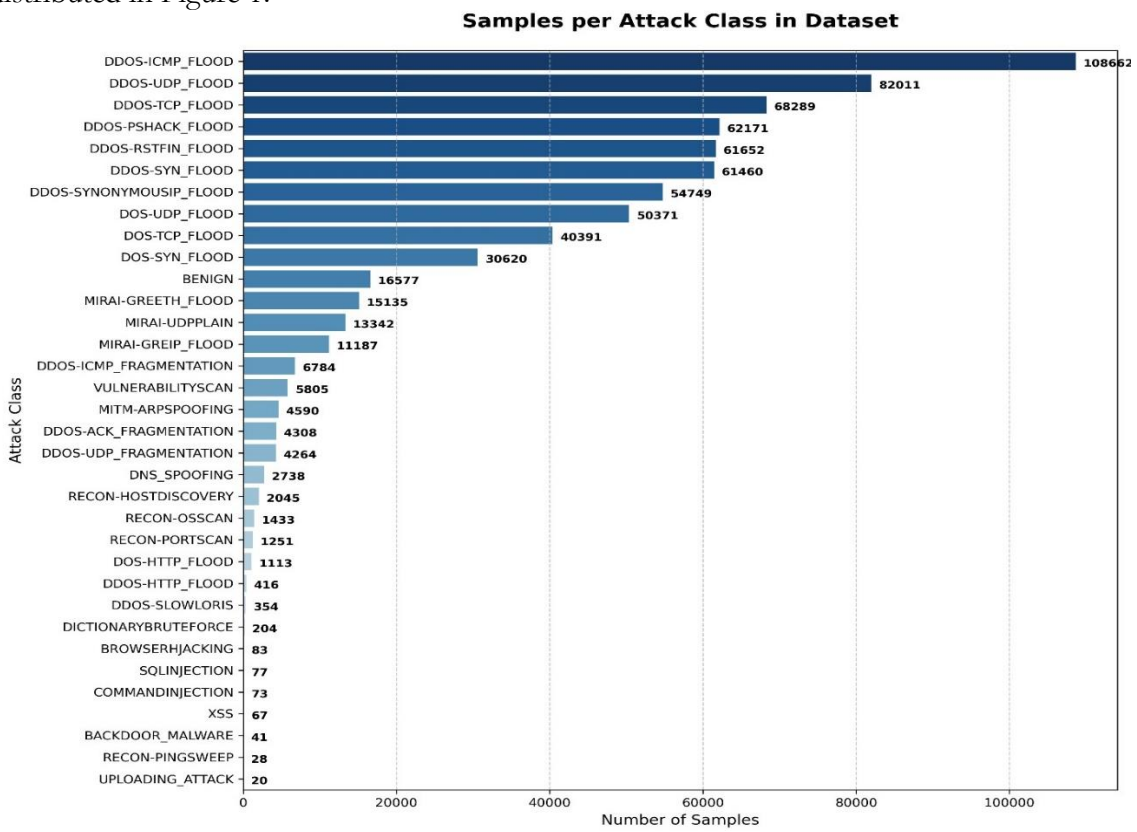


Figure 1. Distribution of samples across the original CICIoT2023 attack labels

Label construction and Class Merging:

The initial 33 attack categories were combined into 8 primary classes based on attack similarities and behavior to make the multi-class classification effort more manageable and relevant. Benign, DDoS, DoS, Mirai, Recon, Spoofing, Web-based, and Brute Force are the final categories. By reducing label fragmentation, this merging phase enables the models to learn more comprehensive attack patterns rather than numerous smaller attack categories. Additionally, it clarifies the experimental analysis, particularly when examining how different class imbalance management strategies affect the main attack groups, as shown in Table 1.

Table 1. Mapping of original CICIoT2023 attack labels into the final 8-class setting.

Final Class (8-class)	Original Labels Merged (from CICIoT2023)
Benign	Benign
DDoS	Ddos-icmp flood, ddos-udp flood, ddos-tcp flood, ddos-pshack flood, ddos-rstfinflood, ddos-syn Flood, ddos-synonymousip flood, ddos-icmp fragmentation, ddos-ack fragmentation, ddos-Udp fragmentation, ddos-http flood, ddos-Slowloris
DoS	Dos-udp flood, dos-tcp flood, dos-syn flood, doshttp flood
Mirai	Mirai-greeth flood, mirai-udpplain, mirai-greip flood
Recon	Recon-hostdiscovery, recon-osscan, recon-Portscan, recon-pingsweep
Spoofing	DNS spoofing, MITM-ARP spoofing
Web-based	SQL injection, XSS, command injection, browser hijacking
Brute Force	Dictionary brute-force, uploading attack

Figure 2 shows the resulting class imbalance after the original labels were divided into eight groups.

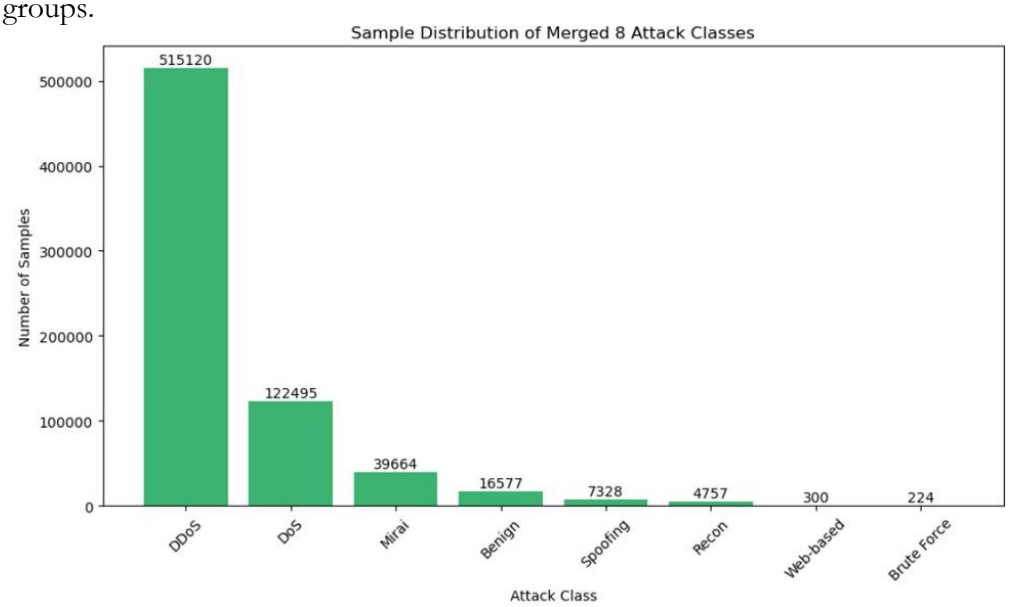


Figure 2. Sample distribution of the merged 8 attack classes

Class Imbalance Analysis:

The dataset remained extremely unbalanced even after the original labels were combined into eight major classes. While classes like Web-based and Brute Force had very few records, the DDoS and DoS classes held the majority of the samples. Due to this imbalance, the classifier may become biased in favor of majority classes and be less able to accurately identify minority attacks. Consequently, a thorough analysis of the class distribution and the application of appropriate imbalance-handling techniques were required before model training, as shown in Table 2.

Table 2. Class distribution after merging the original CICIoT2023 labels into 8 classes

Class	Samples	Percentage (%)
DDoS	515,120	72.31
DoS	122,495	17.20
Mirai	39,664	5.57
Benign	16,577	2.33
Spoofing	7,328	1.03
Recon	4,757	0.67
Web-based	300	0.04
Brute Force	224	0.03

Data Preprocessing:

To guarantee data quality and consistent model training, data preprocessing was applied. Initially, infinite values (±∞) were converted to missing values (NaN). Next, missing values were replaced with the training data median to preserve the statistical characteristics of the data and prevent biased models.

Second, we used outlier handling to minimize the impact of outlier values, which can skew model performance. These preprocessing techniques were implemented to ensure consistency between the training and test data sets.

A leakage-safe approach was adopted by fitting the preprocessing transformations only on the training set and using the same fitted parameters for the test set.

Train–Test Split:

The data was split into training and test sets using an 80/20 stratified split to ensure the same class distribution in the training and testing sets. In imbalanced data, stratification is necessary to include minority classes in training and test data. The test set was not touched to avoid data leakage and ensure a realistic test of the algorithm's performance. Preprocessing and class balancing were only performed on the training set.

Class Imbalance Handling:

We experimented with three approaches to tackle the class imbalance problem in our dataset:

SMOTE, class weighting, and SMOTE-ENN.

SMOTE (Synthetic Minority Oversampling Technique):

SMOTE creates new samples for minority classes by considering the differences between neighboring samples. It helps to balance the classes, but it can insert noise in the overlap region of classes.

Class Weighting:

Class weights were set inversely to class frequencies and used during training. This boosts the influence of minority classes during training without changing the size of the data.

SMOTE-ENN (Hybrid Approach):

SMOTE-ENN is oversampling with data cleaning. Step 1: SMOTE creates synthetic samples of the minority class. In the second step, Edited Nearest Neighbors (ENN) cleans the data by removing misclassified samples.

Given the highly unbalanced nature of CICIoT2023, SMOTE-ENN was chosen as the primary hybrid sampling method because it not only enhances the minority representation but also enhances class separability by removing uncertain samples. Moreover, to mitigate the effect of the majority classes, random under sampling was used to limit the size of the large classes before SMOTE-ENN. Imbalance correction techniques (both under sampling and oversampling) were only applied to the training set, and the test set was left untouched. This approach guarantees an objective test, avoiding overfitting and data leakage problems.

Classification Models:

In this study, we used two lightweight tree-based classifiers:

Random Forest (RF):

An ensemble of 200 trees, trained in parallel (n_jobs = -1), and with a random state set to 42 for consistency.

LightGBM (LGBM):

Set up for multi-class classification using efficient gradient boosting, suitable for large tabular datasets. TabNet was considered but not further investigated due to increased run-time and no improvement over tree-based models.

Experimental Setup:

Experiments were run on a CPU with four logical cores and 7.9 GB RAM on Python (Anaconda). Both Random Forest and LightGBM used multi-threading.

Evaluation Metrics:

The following multi-class performance metrics were used:

Accuracy
Precision
Recall
F1-score
Macro-averaged metrics
Confusion Matrix

Macro-averaged metrics were highlighted to ensure that all classes were weighted equally, especially minority attacks.

Experimental Pipeline:

Our experimental pipeline is shown in Figure 3. It starts with the CICIoT2023 dataset, which is preprocessed by class merging. Stratified sampling is used to divide the data into training and testing sets. We use class balancing (SMOTE, class weighting, SMOTE-ENN) only on the training set. The enhanced training data is then used to train the classification models, with testing conducted on the original test set. This process is reproducible, avoids data leakage, and allows a fair evaluation of methods to handle class imbalance in multi-class intrusion detection.

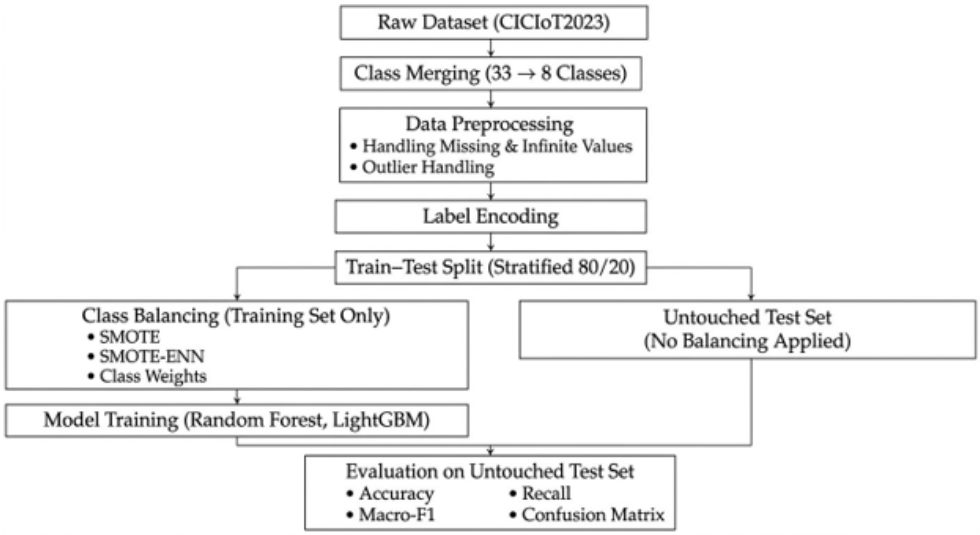


Figure 3. Proposed methodology illustrating preprocessing, leakage-aware class balancing, model training, and evaluation workflow.

Results and Discussion:

The results of the experiments on the CICIoT2023 dataset are presented and discussed in this section. In order to provide a realistic measure of performance, all the techniques used

to preprocess and balance class distributions were only applied to the training set (80%) while the test set (20%) was left unmodified. We ran the experiments over three different classification granularities: 2-class (Attack vs. Benign), 8-class (merged attack families), and 33-class (fine-grained attack types) to investigate the effects of class imbalance at different fine-grained levels.

Baseline with Class Imbalance:

The baseline performance of various classifiers under different classification granularities and without an imbalance correction strategy is presented in Table 3.

Table 3. Performance comparison of TabNet, LightGBM, and Random Forest across 2-class, 8-class, and 33-class settings.

Attack Classifier	Classification Type	Accuracy	Precision	Recall	F1-Score
TabNet	2-class	0.99	0.99	0.99	0.99
	8-class	0.84	0.83	0.84	0.81
	33-class	0.76	0.78	0.76	0.74
LightGBM	2-class	0.99	0.99	0.99	0.99
	8-class	0.85	0.84	0.85	0.83
	33-class	0.77	0.77	0.77	0.75
Random Forest	2-class	0.99	0.99	0.99	0.99
	8-class	0.85	0.83	0.85	0.83
	33-class	0.78	0.78	0.78	0.77

The findings show that all models achieve almost perfect performance in the binary classification (accuracy, precision, recall, and F1-scores around 0.99). But as the number of classes increases, a noticeable drop in performance occurs. In the 8-class classification, accuracy is reduced to around 0.85, signifying the increased complexity in classifying different attack types. This trend is more pronounced in the 33-class case, where the highest-performing model (Random Forest) only attains an accuracy of 0.78 with a corresponding drop in F1-score. This pattern suggests the significant influence of class imbalances and fine-grained classes on model accuracy. Specifically, the 33-class scenario's minority classes are often missed, thus affecting the recall and the overall detection accuracy. This suggests fine-grained intrusion detection is indeed harder and requires robust strategies to handle class imbalance.

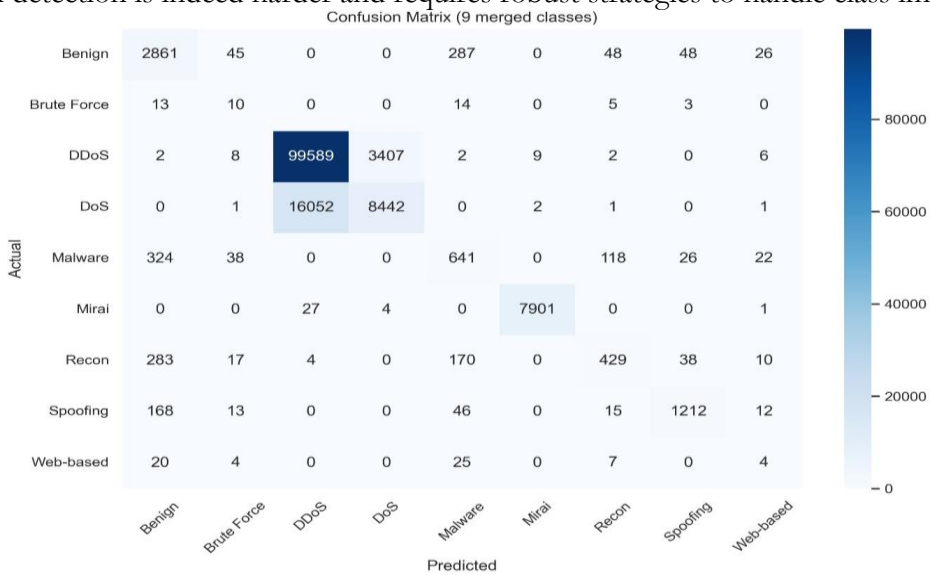


Figure 4. Confusion matrix of the baseline model in the 8-class setting before applying class balancing.

The confusion matrices for the 8-class configuration before using SMOTE-ENN are shown in Figure 4. Many minority class samples, including Brute Force, Web-based, and

Recon, are incorrectly classed as majority classes, like DDoS and DoS, in the baseline model (Figure 4). Due to the extreme class disparity, this suggests a substantial bias in favor of the majority classes.

Class Imbalance:

The data is highly imbalanced, as illustrated in Table 5, where the majority of the samples belong to the classes of DDoS and DoS attacks, while classes such as Web-based and Brute Force are underrepresented.

Impact of Class Imbalance Solutions:

Three approaches were used to handle the class imbalance problem: SMOTE, class weighting, and SMOTE-ENN. The class-wise sample distribution before and after under sampling and SMOTE-ENN is shown in Table 4.

Table 4. Comparison of class-wise sample distribution before and after SMOTE-ENN balancing.

Class Label	Attack Type	Before Balancing	After under sampling	After SMOTE-ENN
0	Benign	16,577	16,577	5,390
1	Brute Force	224	224	13,018
2	DDoS	515,120	10,000	9,589
3	DoS	122,495	10,000	9,343
4	Mirai	39,664	10,000	16,456
5	Recon	10,603	10,603	7,018
6	Spoofing	7,328	7,328	8,856
7	Web-based	300	300	12,861

The table shows that SMOTE-ENN successfully rebalances the data by removing samples from majority classes (under-sampling) and adding samples to minority classes (synthetic minority oversampling). This results in a better balance between classes and enables the model to learn more effectively from all classes. Likewise, Table 5 shows the effect of class weights, where minority classes are given greater weight without changing the dataset.

Table 5. Comparison of class-wise sample distribution before and after Class Weights balancing

Class Label	Attack Type	Before Balancing	Class Weight	Effective Contribution
0	Benign	16,577	0.21	3,481
1	Brute Force	224	15.52	3,476
2	DDoS	515,120	0.01	5,151
3	DoS	122,495	0.04	4,899
4	Mirai	39,664	0.13	5,156
5	Recon	10,603	0.49	5,195
6	Spoofing	7,328	0.71	5,200
7	Web-based	300	17.30	5,190

Table 6 illustrates the impact of SMOTE, in which minority classes are oversampled while the majority classes are not removed. The model may more effectively learn patterns linked to underrepresented attack types as a result of the class distribution becoming more balanced. However, SMOTE may increase ambiguity close to class borders since it creates synthetic samples without taking into account noisy or overlapped regions. Misclassification may arise as a result, especially in intricate multi-class situations. SMOTE does not necessarily provide better overall performance, even while it increases recall for minority classes. Synthetic noise can occasionally have a detrimental effect on class separability and precision.

Consequently, even while SMOTE solves class imbalance to some extent, its shortcomings underscore the need for more sophisticated hybrid techniques like SMOTE-ENN.

Table 6. Comparison of class-wise sample distribution before and after SMOTE

Class Label	Attack Type	Before Balancing	After SMOTE
0	Benign	10,000	16,577
1	Brute Force	10,000	19,500
2	DDoS	515,120	515,120
3	DoS	122,495	122,495
4	Mirai	39,664	39,664
5	Recon	10,603	50,603
6	Spoofing	17,328	10,200
7	Web-based	10,000	75,000

SMOTE boosts the number of minority class samples, but also preserves overlapping and noisy samples, which could harm model performance.

Effectiveness of Balancing Techniques (8-Class Setting):

Table 7 shows the performance of Random Forest and LightGBM models with various imbalance handling techniques in the 8-class setting.

Table 7. Attack Classifier performance with different class balancing techniques for 8 classes

Attack Classifier	Balancing / Method	Accuracy	Precision	Recall	F1-score
LightGBM	Without Class balancing	0.85	0.84	0.85	0.83
	SMOTE	0.79	0.77	0.79	0.78
	Class Weights	0.75	0.96	0.75	0.84
	SMOTE-ENN	0.97	0.97	0.97	0.97
Random Forest	Without Class balancing	0.85	0.83	0.85	0.83
	SMOTE	0.78	0.76	0.78	0.76
	Class Weights	0.78	0.76	0.78	0.85
	SMOTE-ENN	0.98	0.98	0.98	0.98

Neither model performs well on unbalanced data (accuracy around 0.85). But SMOTE shows a slight drop in performance, suggesting that naive oversampling may harm the model's ability to generalize. Weighting increases the impact of the minority classes but has variable results and even a slight decrease in recall in some instances. Instead, the SMOTE-ENN approach leads to a substantial improvement in all metrics. LightGBM's accuracy is around 0.97, while Random Forest's is around 0.98. More significantly, the macro-recall and F1 score are also improved, suggesting improved class balance.

Class-wise Performance Analysis:

Figure 5 shows the class distribution before and after SMOTE-ENN.

Because SMOTE-ENN can combine oversampling and noise reduction, it was chosen to handle the severe class imbalance seen in the dataset.

Confusion matrix-based evaluation is carried out for the 8-class setup in order to further examine the efficacy of the suggested balancing strategy.

The classification of minority classes clearly improves after using SMOTE-ENN (Figure 6). While misclassification is much decreased, the number of accurately predicted occurrences rises. Classes like Recon and Spoofing, for instance, exhibit enhanced diagonal dominance, which suggests higher recall and classification accuracy. The improved

representation of minority classes and the elimination of noisy samples during the ENN cleaning procedure are responsible for these improvements.

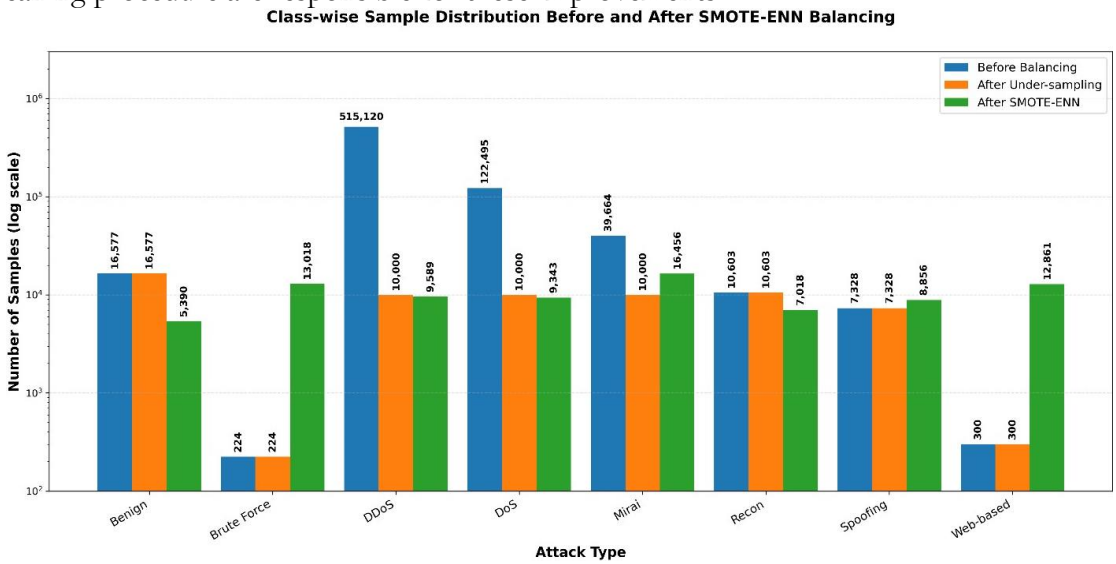


Figure 5. Class-wise sample distribution before and after SMOTE-ENN balancing.

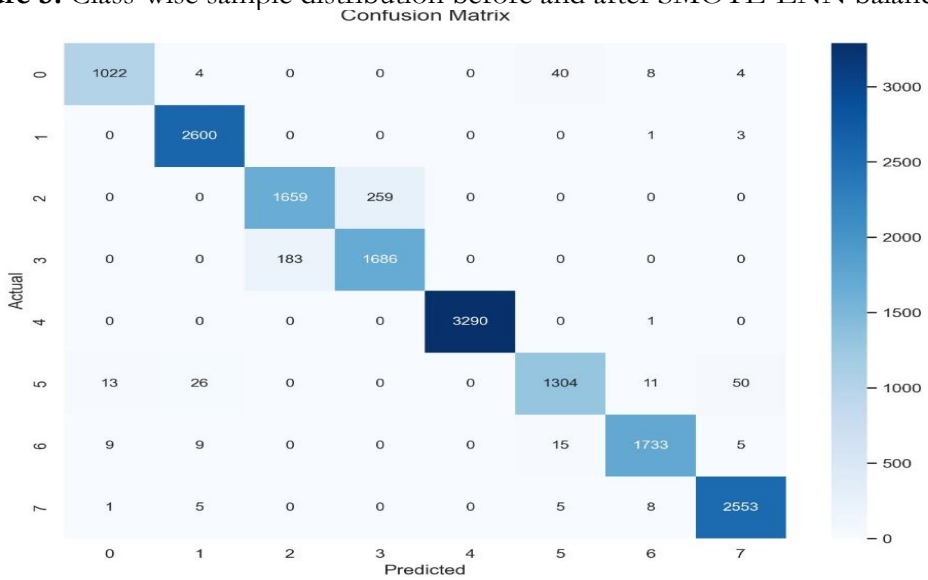


Figure 6. Confusion matrix after applying SMOTE-ENN in the 8-class setting.

This demonstrates that the data becomes well-balanced after SMOTE-ENN, allowing all classes to be learned.

Figures 7 and 8 show the precision, recall, and F1-score for LightGBM and Random Forest. The classwise precision, recall, and F1-score for LightGBM and Random Forest, respectively, under various imbalance control strategies in the 8-class setup. When compared to the imbalanced baseline, a thorough examination reveals that the hybrid SMOTE-ENN strategy greatly enhances the detection performance of minority classes.

After using SMOTE-ENN, minority classes, including Brute Force, Web-based, and Recon, show significant gains in recall and F1-score. Due to inadequate representation in the training data, these classes have poor recall in the baseline scenario. Nevertheless, the models can accurately identify more examples from these classes after balancing, which lowers the number of false negatives.

Additionally, by eliminating noisy and overlapping samples through the ENN cleaning stage, the SMOTE-ENN technique improves class separability. Without lowering the

detection accuracy of dominant classes like DDoS and DoS, this leads to more consistent performance across minority and majority classes.

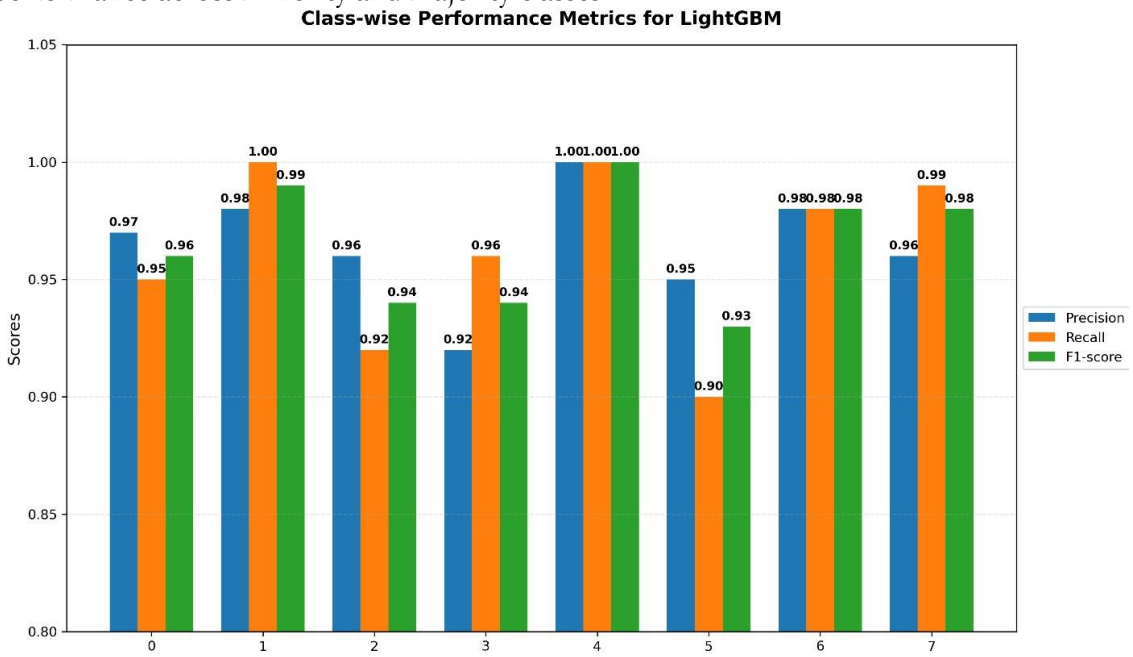


Figure 7. Class-wise precision, recall, and F1-score for LightGBM in the 8-class setting.

The Random Forest classifier consistently exhibits strong precision, recall, and F1-score across all classes in the 8-class situation, as illustrated in Figure 8. For some classes, like class 4, the model performs almost flawlessly, demonstrating great separability and efficient learning of dominating patterns. Additionally, minority classes show notable progress as well, with balanced precision and recall scores that demonstrate the efficacy of the used class balancing strategy. Even while some classes, like class 5, show minor fluctuations, overall performance is steady and well-balanced. These findings demonstrate that, when paired with suitable imbalance management techniques, Random Forest may sustain strong classification performance across the majority and minority classes.

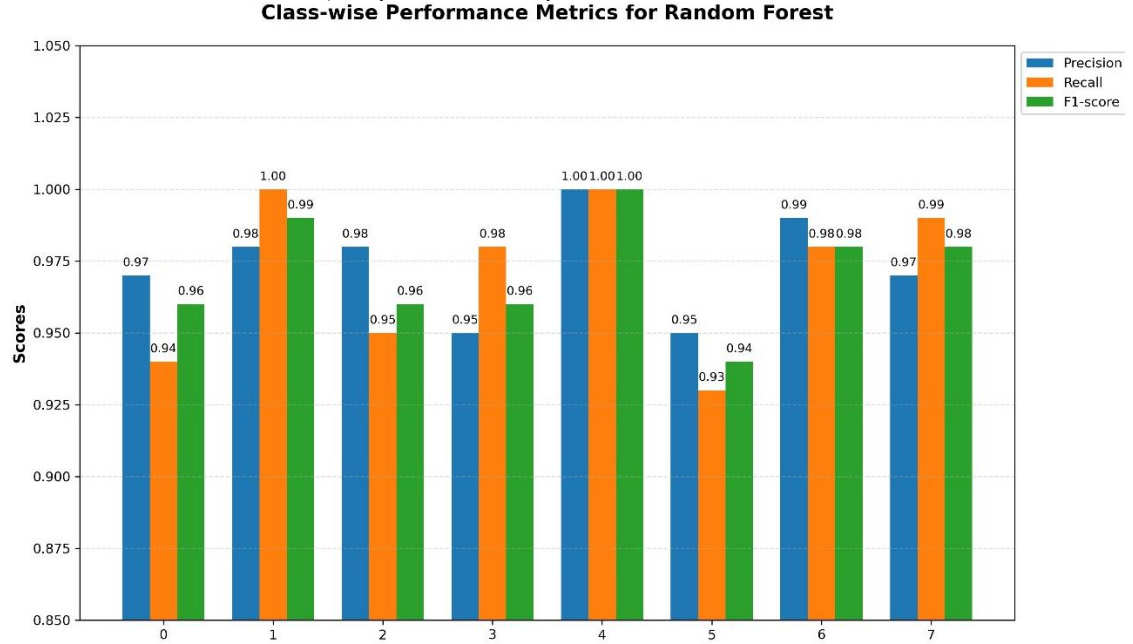


Figure 8. Class-wise precision, recall, and F1-score for Random Forest in the 8-class setting.

The findings show that both models perform well across all classes after balancing. Specifically, the minority classes (Brute Force, Web-based, and Recon) have higher recall rates than the imbalanced baseline. This suggests that SMOTE-ENN helps improve the detection of minority classes while ensuring good performance for majority classes.

Discussion:

Our experimental findings have shown that class imbalance has a substantial impact on the performance of intrusion detection, especially in multi-class classification. The models excel in binary classification, but not with increasing classes. Of the approaches tested, SMOTE does not yield significant improvement as it cannot remove noisy and overlapping instances. Weighting improves the balance, but the results are unstable. However, SMOTE-ENN is the most successful technique. This technique oversamples and also removes noise, allowing a cleaner and more balanced data set, leading to better classification and generalization. However, there are still difficulties in identifying very rare classes, especially in the 33-class scenario. This indicates the need for more research on advanced imbalance and feature engineering approaches for fine-grained intrusion detection.

Conclusion:

In this research, we explored the effect of class imbalance on IoT intrusion detection, using the CICIoT2023 dataset with three different levels of classification: binary (2-class), 8-class, and 33-class. The findings show that although machine learning classifiers perform almost flawlessly in binary classification, they struggle to perform well in multi-class classification, especially in the 33-class fine-grained setting. This decline is largely attributed to class imbalance and the presence of minority attack classes.

To overcome this limitation, we tested three class imbalance mitigation strategies: SMOTE, class weighting, and SMOTE-ENN. In particular, the combined SMOTE-ENN technique consistently performed better than other techniques, achieving better class balance and detection accuracy. The approach of generating new instances and eliminating noisy samples allowed models to better define decision boundaries and achieve higher accuracy, recall, and F1-score, particularly for minority classes. The results demonstrate the need for adequate handling of class imbalance to achieve robust multiclass IoT intrusion detection. Specifically, the research indicates that macro-level metrics and class-specific analysis for evaluating imbalanced data are required to understand the performance of models. Yet, while enhanced, there are still some limitations. The 33-class performance is still limited by the extreme imbalance and the small number of samples for some attacks. Moreover, this work is based on offline analysis, and the real-time deployment issues and streaming settings are not taken into account.

In conclusion, this study offers a comprehensive and leakage-aware evaluation framework for the study of class imbalance in IoT intrusion detection, and shows that hybrid resampling methods are effective in improving multi-class detection performance.

Implication of this Study:

This study has several implications for research and practical deployment of IoT intrusion detection systems (IDS). In practice, the findings suggest that hybrid methods of addressing class imbalance, such as SMOTE-ENN, can greatly enhance detection rates for underrepresented attack types, leading to more effective IDS deployment in real-world IoT systems where class imbalance is prevalent.

From a theoretical perspective, this work demonstrates that it is essential to assess model performance using macro-level metrics, rather than only accuracy, as conventional metrics can mask sub-optimal performance on minority classes. The findings also highlight the importance of leakage-free experimental design for reproducible results.

For practical IoT deployment in real time, the combination of lightweight tree-based models (Random Forest and LightGBM) and efficient imbalance handling techniques offers

a practical solution to ensure both accuracy and speed. However, more research is needed to adapt the results for streaming and large-scale real-time systems.

Despite these contributions, several limitations of this study should be acknowledged. First, the computational study is restricted to limited hardware configurations, and a thorough assessment of computational cost in large-scale or resource-constrained IoT contexts has not been thoroughly investigated. Second, because just one dataset (CICIoT2023) was used for the tests, the results may not be as applicable to other datasets with distinct traffic patterns or attack distributions. Third, real-time deployment factors, including streaming data processing, latency limitations, and dynamic attack behavior, are not taken into account because the study is focused on offline evaluation. Future research should address these elements since they are essential for realistic IoT intrusion detection systems.

Future Work:

Although the proposed hybrid imbalance handling methods have been proven to be effective for multi-class IoT intrusion detection, there are several avenues for future work.

First, more improvements can be made for the 33-class fine-grained classification task. Given the severe class imbalance and scarcity of data for some of the attack types, future research could experiment with more sophisticated methods such as threshold adjustment, cost-sensitive learning, or dynamic decision boundaries to enhance the detection of minority classes.

Second, this work is performed in an offline manner. Real-time intrusion detection using streaming data would help to extend the proposed approach and gain a deeper understanding of the feasibility of deployment in dynamic IoT networks.

Third, further validation of the proposed approach should be conducted using different IoT intrusion datasets (e.g., UNSW-NB15, CICIDS2017) to evaluate the effectiveness and adaptability of the proposed approach under various network and attack scenarios.

Moreover, future work could explore deep learning-based approaches for handling imbalanced datasets, such as Generative Adversarial Network (GAN)-based data augmentation, to further improve minority class representation without noise.

Lastly, incorporating lightweight models with efficient feature selection and hardware-oriented deployment strategies may enhance scalability and efficiency for practical IoT deployment.

These avenues will help further enhance the practicality of intrusion detection systems under highly imbalanced multi-class settings.

References:

[1] "(PDF) Optimizing Random Forest for IoT Cyberattack Detection using SMOTE: A Study on CIC-IoT2023 Dataset." Accessed: May 08, 2026. [Online]. Available: https://www.researchgate.net/publication/399485131_Optimizing_Random_Forest_for_IoT_Cyberattack_Detection_using_SMOTE_A_Study_on_CIC-IoT2023_Dataset

[2] "A two-tier optimization strategy for feature selection in robust adversarial attack mitigation on internet of things network security | Scientific Reports." Accessed: May 08, 2026. [Online]. Available: https://www.nature.com/articles/s41598-025-85878-3

[3] S. Wali and I. Khan, "Explainable AI and Random Forest Based Reliable Intrusion Detection system," *Comput. Sci. Eng.*, Dec. 2021, doi: 10.36227/TECHRXIV.17169080.V1.

[4] W. A. H. Salman and C. H. Yong, "Overview of the CICIoT2023 Dataset for Internet of Things Intrusion Detection Systems," *Mesopotamian J. Big Data*, vol. 2025, pp. 50–60, Jan. 2025, doi: 10.58496/MJBD/2025/004.

[5] S. K. R. Mallidi and R. R. Ramisetty, "Optimizing Intrusion Detection for IoT: A

Systematic Review of Machine Learning and Deep Learning Approaches With Feature Selection and Data Balancing," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 15, no. 2, p. e70008, Jun. 2025, doi: 10.1002/WIDM.70008;JOURNAL:JOURNAL:19424795;ISSUE:ISSUE:DOI.

[6] T. Ammannamma, A. S.N. Chakravarthy, "A hybrid lightweight feature extraction assisted ensemble approach for intrusion detection with ESMOTE-based class imbalance handling in IoT networks," *Comput. Electr. Eng.*, vol. 130, p. 110846, 2026, doi: https://doi.org/10.1016/j.compeleceng.2025.110846.

[7] B. Kiranmayee, M. S. Devi, K. Susheela, R. Dhumpati, K. K. R. Penubaka, and U. G. Naidu, "Developing a Robust Intrusion Detection System Using SMOTE and Hybrid SVNN Model," *4th Int. Conf. Sentim. Anal. Deep Learn. ICSADL 2025 - Proc.*, pp. 369–376, 2025, doi: 10.1109/ICSADL65848.2025.10933456.

[8] "Comparative Analysis of Machine Learning Algorithms for Anomaly Detection in IoT Networks Using CICIoT2023 Dataset | INFOCOMP Journal of Computer Science." Accessed: May 08, 2026. [Online]. Available: https://infocomp.dcc.ufla.br/index.php/infocomp/article/view/5342

[9] "(PDF) Imbalanced Data Problem in Machine Learning: A Review." Accessed: May 08, 2026. [Online]. Available: https://www.researchgate.net/publication/388208416_Imbalanced_Data_problem_in_Machine_Learning_A_review

[10] "Resampling approaches to handle class imbalance: a review from a data perspective | Journal of Big Data | Springer Nature Link." Accessed: May 08, 2026. [Online]. Available: https://link.springer.com/article/10.1186/s40537-025-01119-4

[11] "Enhancing IoT security: A comparative study of feature reduction techniques for intrusion detection system - ScienceDirect." Accessed: May 08, 2026. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667305324000814

[12] S. R. Alve, M. Z. Mahmud, S. Islam, M. A. Chowdhury, and J. Islam, "Resource-Efficient Machine Learning Approaches for Multi-Class Threat Detection in IoT Environments," *2025 IEEE Int. Conf. Quantum Photonics, Artif. Intell. Networking, QPAIN 2025*, 2025, doi: 10.1109/QPAIN66474.2025.11171769.

[13] "Intrusion Detection in IoT Environment Using Hyperparameters Tuned Machine and Deep Learning Models on the CICIoT2023 Dataset | Request PDF." Accessed: May 08, 2026. [Online]. Available: https://www.researchgate.net/publication/396634807_Intrusion_Detection_in_IoT_Environment_Using_Hyperparameters_Tuned_Machine_and_Deep_Learning_Models_on_the_CICIoT2023_Dataset

[14] Y. Kim, C. Won, and H. Kim, "Impact of Data Processing Techniques on AI Models for Attack-Based Imbalanced and Encrypted Traffic within IoT Environments," *Comput. Mater. Contin.*, vol. 86, no. 1, pp. 1–28, Nov. 2025, doi: 10.32604/CMC.2025.069608.

[15] "Detection of IoT Botnet Cyber Attacks using Machine Learning | Request PDF." Accessed: May 08, 2026. [Online]. Available: https://www.researchgate.net/publication/371203020_Detection_of_IoT_Botnet_Cyber_Attacks_using_Machine_Learning

[16] P. Tharun, R. Sumathi, S. Manjula, R. S. Charan, A. Dhanush Saran, and M. L. Reddy, "Convolutional Neural Network for Advanced Intrusion Detection for Data Balancing on System Efficiency," *Proc. 8th Int. Conf. Comput. Methodol. Commun. ICCMC 2025*, pp. 145–152, 2025, doi: 10.1109/ICCMC65190.2025.11140907.

[17] P. Singh, D. Nehra, V. Mangat, and K. Kumar, "Enhancing intrusion detection with ResNet and SMOTE-ENN: a deep learning approach to class imbalance in

CICIDS2017," *Int. J. Inf. Technol. 2025*, pp. 1–10, Nov. 2025, doi: 10.1007/S41870-025-02766-9.

[18] "(PDF) Towards Robust IoT Security: The Impact of Data Quality and Imbalanced Data on AI-Based IDS." Accessed: May 08, 2026. [Online]. Available: https://www.researchgate.net/publication/394324485_Towards_Robust_IoT_Security_The_Impact_of_Data_Quality_and_Imbalanced_Data_on_AI-Based_IDS

[19] Y. J. Park and C. K. Ma, "A novel instance density-based hybrid resampling for imbalanced classification problems," *Soft Comput. 2025 294*, vol. 29, no. 4, pp. 2031–2045, Mar. 2025, doi: 10.1007/S00500-025-10499-X.

[20] Y. Guo, Y. Kou, L. Z. Yi, and G. H. Fu, "HiBBKA: A Hybrid Method With Resampling and Heuristic Feature Selection for Class-Imbalanced Data in Chemometrics," *J. Chemom.*, vol. 39, no. 5, p. e70029, Apr. 2025, doi: 10.1002/CEM.70029;PAGEGROUP:STRING:PUBLICATION.

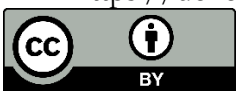
[21] "(PDF) Hybrid Sampling Approach to Enhance Intrusion Detection System in IoT Networks." Accessed: May 08, 2026. [Online]. Available: https://www.researchgate.net/publication/390909719_Hybrid_Sampling_Approach_to_Enhance_Intrusion_Detection_System_in_IoT_Networks

[22] Nahida Nigar, Rashed Mustafa, "Enhanced Intrusion Detection via Hybrid Data Resampling and Feature Optimization," *IEEE Access*, 2025, [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=11141474

[23] Monirah Al-Ajlan, Mourad Ykhlef, "GAN-AHR: A GAN-Based Adaptive Hybrid Resampling Algorithm for Imbalanced Intrusion Detection," *Electronics*, vol. 14, no. 17, p. 3476, 2025, doi: https://doi.org/10.3390/electronics14173476.

[24] "Robust Intrusion Detection System Using an Improved Hybrid Deep Learning Model for Binary and Multi-Class Classification in IoT Networks. | EBSCOhost." Accessed: May 08, 2026. [Online]. Available: https://openurl.ebsco.com/EPDB%3Agcd%3A11%3A10491979/detailv2?sid=ebsco%3Aplink%3Acrawler-gcd&id=ebsco%3Agcd%3A184201869&crl=c&jrnl=22277080&link_origin=www.google.com

[25] Mingming Han, Husheng Guo, "A new data complexity measure for multi-class imbalanced classification tasks," *Pattern Recognit.*, vol. 157, p. 110881, 2025, doi: https://doi.org/10.1016/j.patcog.2024.110881.



Copyright © by authors and 50Sea. This work is licensed under the Creative Commons Attribution 4.0 International License.