applied sciences

MDPI

Article

ROSE-BOX: A Lightweight and Efficient Intrusion Detection Framework for Resource-Constrained IIoT Environments †

Silin Peng [1,2], Yu Han [1], Ruonan Li [2], Lichen Liu [2], Jie Liu [3,4] and Zhaoquan Gu [2,5,*]

[1] School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 518107, China; pengslin6@mail2.sysu.edu.cn (S.P.); hanyu25@mail.sysu.edu.cn (Y.H.)
[2] Department of New Networks, Pengcheng Laboratory, Shenzhen 518055, China; lirn@pcl.ac.cn (R.L.); liulch@pcl.ac.cn (L.L.)
[3] International Research Institute for Atificial Intelligence, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China; jieliu@hit.edu.cn
[4] National Key Laboratory of Smart Farm Technologies and Systems, Harbin 150030, China
[5] College of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China
* Correspondence: guzhaoquan@hit.edu.cn

† This paper is an extended version of our paper published in 2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA), Kaifeng, China, 30 October–2 November 2024. https://ieeexplore.ieee.org/document/10885136.

Abstract: The rapid advancement of the Industrial Internet of Things (IIoT) has transformed industrial automation, enabling real-time monitoring and intelligent decision making. However, increased connectivity exposes IIoT systems to sophisticated cyber threats, which may pose significant security risks, especially in resource-constrained IIoT environments where computational efficiency is critical. Existing intrusion detection solutions often suffer from high computational overhead and inadequate adaptability, rendering them impractical for real-time deployment in IIoT environments. To address these challenges, this study introduces a lightweight and efficient intrusion detection framework tailored for resource-constrained IIoT environments. Firstly, an XGBoost-assisted Random Forest (XGB-RF) method is proposed to select the most important features to obtain an optimal feature subset. Moreover, SMOTE (Synthetic Minority Oversampling Technique) is utilized to balance the optimal feature subset to improve detection precision. Furthermore, to reduce computing resource requirements and latency while improving detection performance, Bayesian optimization is applied to fine-tune the parameters of XGBoost (BO-XGBoost) to obtain the best detection results. Finally, extensive experiments on benchmark datasets, including CIC-IDS2017, CSE-CIC-IDS2018, and CIC-DDoS2019, demonstrate that the proposed method, which we call ROSE-BOX (Random Forest, Synthetic Minority Oversampling Technique, and BO-Xgboost), achieves a detection accuracy exceeding 99.85% while maintaining low latency and CPU occupancy rates. Our findings highlight the robustness, lightweight nature, and efficiency of ROSE-BOX, making it well-suited for real-time intrusion detection in resource-constrained IIoT environments.

Keywords: Industrial Internet of Things; intrusion detection; resource-constrained IIoT environments

check for updates

Academic Editors: Sadaf Hina, Miguel García-Pineda and Luis Javier García Villalba

Received: 28 April 2025
Revised: 3 June 2025
Accepted: 5 June 2025
Published: 8 June 2025

Citation: Peng, S.; Han, Y.; Li, R.; Liu, L.; Liu, J.; Gu, Z. ROSE-BOX: A Lightweight and Efficient Intrusion Detection Framework for Resource-Constrained IIoT Environments. *Appl. Sci.* **2025**, *15*, 6448. https://doi.org/10.3390/app15126448

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The Industrial Internet of Things (IIoT) is revolutionizing the industrial landscape by enabling seamless connectivity among devices, sensors, controllers, and cloud services. In modern IIoT networks, communications occur at the millisecond level, devices continuously exchange data, and even transient events can generate massive volumes of traffic. In such dynamic environments, ensuring robust security is not just important—it is critical. With the potential for cyberattacks to propagate almost instantaneously, any delay in threat detection can lead to cascading failures, production interruptions, and even catastrophic safety hazards [1,2].

Resource constraints represent a significant challenge in IIoT deployments. Many IIoT devices are embedded systems or edge devices that operate with limited processing power, memory, and energy reserves [3]. This inherent limitation makes it impractical to deploy computationally intensive security solutions directly on these platforms. Traditional rule-based intrusion detection systems (IDSs) offer limited adaptability to emerging attack vectors. While deep learning-based IDSs have demonstrated high detection accuracy, their heavy computational overhead renders them unsuitable for real-time applications in resource-limited environments [3].

Lightweight and efficient intrusion detection is crucial in IoT contexts, as the rapid communication between devices and the cloud means that even a slight delay in the real-time identification and mitigation of attacks can lead to significant security incidents. For instance, if an attack goes undetected even for a few milliseconds, malicious data packets could compromise critical control systems, lead to unauthorized access, or trigger shutdowns of vital operations. These risks underscore the need for a security solution that not only delivers high detection accuracy but also operates with ultra-low latency and minimal resource consumption [4].

Traditional IDS approaches, including signature-based and threshold-based methods, struggle to detect zero-day attacks and require extensive manual configuration [5–7]. Meanwhile, deep learning-based IDSs, such as convolutional neural networks (CNNs) and long short-term memory (LSTM) models, demonstrate high detection accuracy but suffer from excessive computational overhead, making them unsuitable for IIoT environments with constrained processing power [8]. Furthermore, there are four challenges in IIoT environments: (1) High-dimensional IIoT data often contain redundant features that reduce the model's discriminative power; (2) severe class imbalances in IIoT datasets lead to the under-representation of rare but critical attack instances; (3) suboptimal hyperparameter settings can dramatically affect the performance and generalizability of IDSs; (4) high computational complexity and latency impede real-time threat detection on resource-limited IIoT devices. These challenges necessitate the development of lightweight, high-performance IDS solutions that are capable of detecting multi-class cyber threats while maintaining low latency and minimal resource consumption [9].

These challenges have been highlighted in the recent literature. For instance, ensemble methods such as AdaBoost [10] suffer from limited discrimination across classes, whereas deep learning methods—including LSTM [11] and CNN-LSTM [7]—offer high accuracy but require extensive computational resources, making them impractical for edge IIoT devices. Additionally, previous works, such as that of Kang et al. [12], demonstrated strong recall via autoencoder–XGBoost integration but lacked precision due to imbalance and noise sensitivity. These findings directly motivate our design of ROSE-BOX: a lightweight and efficient multi-class intrusion detection framework specifically designed for resource-constrained IIoT environments. We propose an efficient hybrid feature selector (XGB-RF), SMOTE-based data balancing, and Bayesian-optimized XGBoost to overcome these specific deficiencies. The low computational demand of ROSE-BOX makes it ideally suited for deployment across a wide array of devices, from industrial controllers to edge computing nodes and embedded systems. ROSE-BOX is designed to scale across diverse IIoT platforms by leveraging three principles: (1) early feature reduction via XGBoost-assisted Random Forest to minimize input size, (2) data balancing with SMOTE on a compact subset to prevent unnecessary memory usage and improve detection performance, and (3) the parameters of ROSE-BOX are tuned via offline Bayesian optimization during training to ensure low-latency inference. These steps collectively enable scalable deployment across edge, embedded, and industrial systems. Our findings establish ROSE-BOX as a scalable, lightweight, and real-time IDS solution, bridging the gap between detection performance and practical deployment in resource-limited IIoT applications. The contributions of our work are listed as follows:

- We propose a hybrid feature selection framework that synergistically combines Random Forest and XGBoost algorithms to identify and rank the most salient features from high-dimensional IIoT datasets. By eliminating redundant and non-informative features, this approach significantly reduces computational overhead and enhances detection accuracy, making it highly suitable for deployment in resource-constrained IIoT environments.
- We incorporate the Synthetic Minority Oversampling Technique (SMOTE) [13] to address the class imbalance inherent in intrusion detection datasets and Bayesian optimization (BO) [9] to fine-tune the hyperparameters of XGBoost, further optimizing detection performance while minimizing computational demands.
- The experimental results indicate that the proposed ROSE-BOX framework [14] consistently achieves detection accuracies exceeding 99.85%, markedly surpassing traditional intrusion detection systems. Moreover, ROSE-BOX substantially reduces latency and computational overhead, rendering it highly suitable for real-time deployment in resource-constrained Industrial Internet of Things (IIoT) environments.

The novelty of the ROSE-BOX framework lies in its synergistic integration of XGBoost-assisted Random Forest for efficient and relevant feature extraction, SMOTE for handling class imbalance, and Bayesian optimization for real-time parameter tuning, all tailored for resource-constrained IIoT and smart home environments. This integrated approach allows ROSE-BOX not only to detect intrusions with extremely high accuracy but also to minimize computational and energy overhead—a critical feature for smart home and smart grid systems. Moreover, by facilitating local detection without reliance on cloud-based computation [15], ROSE-BOX inherently enhances privacy protection and reduces latency—two central concerns in edge-based smart home security, agricultural robot networks, and cyber ranges [16–18].

The rest of this paper is designed as follows. Section 2 introduces the related studies on intrusion detection in IIoT. Section 3 proposes the detection model ROSE-BOX. Section 4 introduces the experimental results and analyzes the detection performance of the proposed model. Section 5 summarizes this paper and presents the future directions.

2. Related Work

This section provides a comprehensive review of recent advances in IIoT anomaly detection, with a particular focus on analyzing the strengths and limitations of mainstream detection approaches, including traditional intrusion detection methods and machine learning-based and deep learning-based methods [1].

For traditional intrusion detection methods, on the one hand, anomaly-based intrusion detection (AID) [19,20] identifies potential threats by contrasting observed device activities with established normal operational patterns. This methodology employs dynamically computed thresholds (THs) to quantify behavioral deviations. When a device exceeds predefined TH values, it is initially flagged for monitoring. Persistent anomalies over subsequent time intervals trigger classification as malicious, prompting the isolation of the device from the network. Researchers have developed diverse AID variants to enhance IoT security, including machine learning-driven pattern recognition, entropy-based traffic analysis, and adaptive threshold optimization frameworks. For instance, some approaches utilize temporal correlation models to minimize false alarms, while others incorporate contextual awareness to differentiate between operational anomalies and genuine cyberattacks.

On the other hand, Amin et al. [21] proposed a Bloom filter-based intrusion detection system (IDS) for IP-enabled ubiquitous sensor networks. However, the system's primary limitations stem from its dependence on static signature databases, rendering it ineffective against zero-day attacks and communication latency incurred during distributed node-to-filter data aggregation.

For machine learning-based detection methods, earlier studies in IIoT intrusion detection have investigated various machine learning techniques that combine conventional feature engineering with ensemble-based classifiers. For instance, Yulianto et al. [10] integrated Principal Component Analysis (PCA), Ensemble Feature Selection (EFS), and SMOTE to enhance the performance of AdaBoost. Although this approach provided improved detection capability, its relatively modest discrimination between benign and malicious traffic suggests that further advanced feature engineering or alternative ensemble frameworks could be beneficial. In a similar vein, Li et al. [22] applied recursive feature elimination (RFE) to identify optimal feature subsets and subsequently utilized a genetic algorithm to optimize a LightGBM classifier. While this method demonstrated a significant boost in detection performance, the computational expense associated with genetic algorithms poses challenges for real-time applications and large-scale systems.

Kang et al. [12] integrated autoencoders with XGBoost for intrusion detection, demonstrating strong recall performance but facing limitations in precision, which may lead to increased false positives—particularly in imbalanced data scenarios without the aid of resampling techniques. In XGBoost, the multi-class classification is handled by employing the Softmax function to map the output of the decision trees to multiple class probabilities. It constructs and optimizes multiple trees sequentially, with each tree aiming to minimize the loss function, which is based on the difference between the predicted probabilities and the true labels. The final predicted class is the one with the highest probability after combining the outputs of all the trees. As for large-scale IoT data, Jemili et al. [23] developed a tailored multi-class ensemble framework. By combining multiple base learners (e.g., decision trees, KNN, naive Bayes) via weighted voting, they achieved robust detection across diverse attack types. Although ensemble methods improve overall accuracy and resilience to concept drift, they typically demand substantial memory and incur high inference latency—constraints in resource-limited IIoT deployments.

Other studies, such as those by Siddartha et al. [24], have employed the KNN algorithm for identifying Denial-of-Service attacks. Although promising in certain contexts, KNN-based approaches are typically hampered by high memory requirements and scalability issues in high-dimensional spaces. In addition, Random Forest selects features by measuring the importance of each feature through its contribution to the impurity reduction (e.g., Gini impurity or information gain) across all trees. Chua et al. [25] combined Random Forest (RF) for feature selection with intrusion detection, yielding high accuracy. However, the RF technique is known to be vulnerable to overfitting, particularly in the presence of noisy data, and its inherent lack of interpretability can be limiting in critical security applications.

In the deep learning domain, several studies have explored the use of autoencoders for detecting previously unseen (zero-day) attacks [5]. Although autoencoders have demonstrated the capacity to outperform conventional anomaly detectors, such as One-Class SVM, their performance remains sensitive to data quality and imbalance, often leading to variable outcomes across different types of attacks. Hybrid deep neural network (DNN) architectures have also been introduced by various researchers [26–28] to achieve scalable multi-class detection. While these methods tend to deliver high detection accuracy, their significant computational requirements for both training and inference can limit their practicality in real-time IIoT deployments.

More complex models that integrate convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and attention mechanisms [2,29,30] have also been proposed. Despite their ability to capture temporal and spatial correlations in network traffic, the resulting model complexity not only increases training and inference time but also poses challenges in avoiding overfitting, especially in scenarios with limited or imbalanced data.

Table 1 summarizes existing IIoT intrusion detection methods and highlights both their strengths (e.g., latency, cost, scalability) and their weaknesses (e.g., high computational cost, poor real-time efficiency, limited adaptability). Although traditional schemes incur low resource overhead, they lack scalability. ML-based methods can certainly improve detection accuracy, but they need to be improved to further reduce detection latency and computational requirements, which is impractical for resource-constrained IIoT environments. Recent DL-based architectures deliver superior accuracy and scalability, yet their high computational requirements, as well as vulnerability to overfitting, hinder real-time deployment on the edge. To overcome these challenges, we introduce ROSE-BOX, a lightweight and efficient intrusion detection framework specifically designed for resource-constrained IIoT environments. Three key motivations guide our design:

Table 1. Condensed comparison of intrusion detection methods.

Method Category	Core Techniques	Latency	Cost	Scalability
Traditional methods	Signature-based IDS [21]; A threshold-based IDS [19,20]	Medium–High	Low	Poor

ML-based methods	PCA/EFS + SMOTE + AdaBoost [10];
	RFE + GAO-LightGBM [22];
	KNN [24]; RF [25];
	AE + XGBoost [12];
	Ensembles [23]
		Medium	Medium	Moderate

DL-based methods	Autoencoders [5];
	CNN [7,30];
	Hybrid DNNs [26–28];
	BiLSTM-GMM [29];
	CNN-LSTM-Attention [8]
		Medium–Low	High	Strong

Ours	ROSE-BOX	Low	Medium–Low	Strong

- Addressing resource-constrained scenarios: Given the stringent resource limitations of IIoT devices, we proposed a framework for minimizing computational complexity in IIoT systems. By focusing on efficient feature selection and model pruning techniques, we ensure that the processing overhead is drastically reduced.
- Ultra-low latency processing: In an IIoT environment, where communications between devices and the cloud are measured in milliseconds, the ability to perform instant threat detection is vital. Without real-time deployment, a delayed response could allow malicious activities to spread unchecked, potentially leading to critical system failures. Each data sample is processed within microseconds, which is crucial for maintaining real-time responsiveness.
- Scalable adaptability and lightweight deployment: Using XGBoost-assisted Random Forest for feature selection and Bayesian optimization for hyperparameter tuning, the

low computational demand of ROSE-BOX makes it ideally suited for deployment across a wide array of devices, from industrial controllers to edge computing nodes and embedded systems. This scalability, coupled with a reduction in required computing resources, allows for broad deployment without overburdening system resources or incurring significant operational costs.

In summary, these design choices position ROSE-BOX to meet the stringent real-time and resource constraints of modern IIoT deployments while providing comprehensive, multi-class intrusion detection, as detailed in the following sections.

3. Methodology

In this section, we introduce the overall architecture of a detection model based on ROSE-BOX.

3.1. Overall Framework

Figure 1 illustrates the architecture of ROSE-BOX, a lightweight and efficient anomaly detection framework designed for real-time intrusion detection in resource-constrained IIoT environments. The system initiates with the low-overhead preprocessing of IIoT datasets, ensuring minimal computational cost. Random Forest is employed for feature selection, prioritizing the most relevant attributes to reduce dimensionality and accelerate model inference. In order to maximize the detection rate of ROSE-BOX, XGBoost is utilized to assist Random Forest in feature selection while simultaneously performing anomaly detection. The process terminates when the detection rate reaches 100%. In contrast to deep learning-based feature selection techniques, which demand considerable training resources and exhibit latency during inference, the proposed XGBoost-assisted Random Forest framework ensures lightweight and accurate feature selection [31,32]. Random Forest provides a stable feature importance ranking, and XGBoost achieves intrusion identification capabilities through multi-class classification. This synergy offers a more practical and computationally feasible alternative to mutual information-based or neural network-based selection methods, especially in constrained IIoT deployments. Importantly, to maintain high detection efficiency and reduce unnecessary processing overhead, an adaptive EarlyStopping mechanism is integrated. This mechanism halts feature selection when detection performance stabilizes, thereby optimizing feature selection by using XGBoost-assisted Random Forest, and the optimal feature subset is obtained after feature selection.

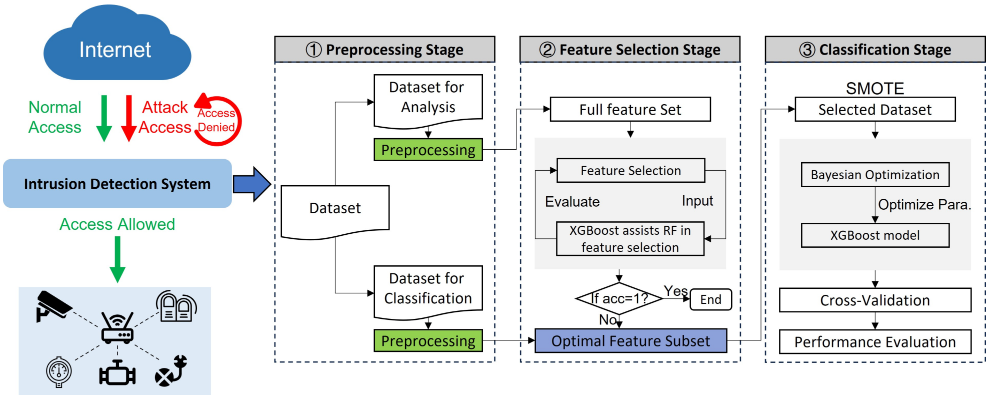

Figure 1. Overall architecture of detection model based on ROSE-BOX.

IIoT datasets often exhibit severe class imbalances, where normal operations vastly outnumber abnormal or attack events [31]. To overcome this challenge, SMOTE is utilized to oversample minority-class attack instances for the optimal feature subset, thereby improving the robustness of the XGBoost model without significantly increasing computational complexity [32]. This approach ensures that even infrequent, high-impact attacks are accurately identified. The selected dataset is obtained after balancing the optimal feature subset, which is then partitioned into training, validation, and test sets to ensure optimal model generalization. Although the framework partitions the dataset into separate stages for feature analysis and classification, this design minimizes total computation by reducing feature dimensionality early. The significant gains in model inference speed and memory reduction offset the cost of feature selection. Moreover, SMOTE is applied only to the compact subset of relevant features, further reducing data expansion and ensuring efficient resource usage. Therefore, the framework achieves a net decrease in computational overhead despite multi-stage processing. Our approach enhances the discriminative power of the model, which is crucial for accurately identifying diverse attack types and facilitating subsequent lightweight and efficient intrusion detection. The operational latency and computational overhead are drastically reduced, showing that this method is suitable for resource-constrained IIoT environments. Although the ROSE-BOX framework integrates multiple modules, each is chosen to maintain computational efficiency. Feature selection using XGBoost-assisted Random Forest significantly reduces dimensionality and noise. SMOTE is applied only after feature reduction, ensuring minimal data expansion. Bayesian optimization is performed during training only, not inference. These design decisions collectively mitigate information overhead, as evidenced by our low latency and CPU metrics in Section 4.6.

To further enhance detection efficiency, a Bayesian optimization algorithm is applied to fine-tune the hyperparameters of the XGBoost model (BO-XGBoost), thereby achieving an optimal trade-off between detection accuracy and computational cost. Ultimately, ROSE-BOX is achieved by implementing these processes, which provide a low-latency, high-accuracy detection mechanism suitable for real-time IIoT cybersecurity applications. The architecture of ROSE-BOX is designed to strike a balance between detection performance and computational feasibility, making it ideal for IIoT systems where low power consumption, real-time response, and scalable deployment are critical.

3.2. The XGBoost Model

As the core component of the proposed architecture, XGBoost is instrumental in achieving high-performance multi-class intrusion detection in the Industrial Internet of Things (IIoT). Therefore, we will introduce the XGBoost model subsequently. We assume that the number of categories is K, and the number of samples is N. Before training begins, initialize the score belonging to each category k for each sample i as $f_{ik}^{(0)} = 0$. Using the Softmax function, the initial prediction probability can be calculated as

$$p_{ik}^{(0)} = \frac{\exp\left(f_{ik}^{(0)}\right)}{\sum_{j=1}^{K} \exp\left(f_{ij}^{(0)}\right)} \tag{1}$$

XGBoost maps the output of the tree to multiple class probability spaces using the Softmax function, and it gradually fits multiple trees through gradient boosting [12]. In each iteration, the model constructs and optimizes a new tree to minimize the current loss. The loss function can be represented as

$$L = -\sum_{i=1}^{N} \sum_{k=1}^{K} y_{ik} \log p_{ik} \tag{2}$$

where y_{ik} is the true label of sample i. If it belongs to the k-th class, then $y_{ik} = 1$; otherwise, $y_{ik} = 0$, and p_{ik} is the probability that sample i is predicted as the k-th class. To measure the deviation between model predictions and true labels, it is necessary to calculate the first derivative g_{ik} and the second derivative h_{ik} of the current score f_{ik}, which can be expressed as

$$g_{ik} = \frac{\partial L}{\partial f_{ik}} = p_{ik} - y_{ik} \tag{3}$$

$$h_{ik} = \frac{\partial^2 L}{\partial f_{ik}^2} = p_{ik}(1 - p_{ik}) \tag{4}$$

We assume that the j-th leaf node of the tree is j, and λ is a regularization parameter that controls the complexity of the model. Furthermore, XGBoost builds a tree for each category in the current iteration and uses splitting gains to select feature splitting points to increase the accuracy of category prediction. The leaf node weights of the tree are optimized based on h_{ik} and g_{ik}, and the weight calculation formula is

$$w_j = -\frac{\sum_{i \in j} g_{ik}}{\sum_{i \in j} h_{ik} + \lambda} \tag{5}$$

After the new tree is generated, add its output value to the current score. For each sample i and its corresponding category k, the score update formula is

$$f_{ik}^{(t+1)} = f_{ik}^{(t)} + w_{ik}^{(t)} \tag{6}$$

Here, $w_{ik}^{(t)}$ is the output of the new tree for the k-th class in this iteration. After each iteration, the updated score is mapped to the corresponding category probability for better gradient calculation in the next step. The predicted probability for each sample is

$$p_{ik}^{(t+1)} = \frac{\exp\left(f_{ik}^{(t+1)}\right)}{\sum_{j=1}^{K} \exp\left(f_{ij}^{(t+1)}\right)} \tag{7}$$

Then, continue iterating and updating to gradually build new trees, updating scores until the preset stopping conditions are met. The stopping condition can be reached when the maximum number of trees is reached or convergence is lost.

After training, the output of the model is the category score f_{ik} corresponding to each sample i. Use the Softmax function to convert scores into predicted probabilities:

$$p_{ik} = \frac{\exp(f_{ik})}{\sum_{j=1}^{K} \exp(f_{ij})} \tag{8}$$

The final predicted category is the category with the highest probability, which is $\arg\max_k p_{ik}$. Therefore, XGBoost can achieve efficient multi-class learning.

3.3. The ROSE-BOX Algorithm

As illustrated in Algorithm 1, the ROSE-BOX algorithm takes as input the IIoT dataset D, the objective function O of the Bayesian optimization algorithm, a parameter set W for XGBoost, the acquisition function F, the SMOTE sampling algorithm, and the classifiers Random Forest (RF) and XGBoost.

Initially, both the best accuracy b_a and the best number of features b_n_f are initialized to zero. After preprocessing, the dataset D is partitioned into training and testing subsets: X_{train}, X_{test}, y_{train}, and y_{test}. The Random Forest model is then trained on X_{train} and y_{train} to compute feature importance rankings, from which the selected feature subset X_{train_s} is derived.

Following the feature selection process, XGBoost is trained on the selected features. To mitigate overfitting and determine the optimal classification performance, EarlyStopping is incorporated during the training phase. Subsequently, the most discriminative features are leveraged to construct a balanced dataset D' through the SMOTE oversampling technique. To further enhance the performance of multi-class intrusion detection while avoiding overfitting, Bayesian optimization is utilized to fine-tune the XGBoost parameters. This is achieved by iteratively training the model on dataset D' and maximizing the objective function. Through this systematic approach, the BO-XGBoost model is ultimately developed for accurate multi-class intrusion detection.

Algorithm 1: The detection algorithm ROSE-BOX

```
Input: $D, W, O, F, RF, SMOTE, XGBoost$
Output: ROSE-BOX
$X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}} \leftarrow \text{Preprocess}(D)$;
Initial($b\_a, b\_n\_f$);
$X_{\text{train}\_s} \leftarrow \text{RF}(X, y)$;
for $n\_f = 1, 2, \ldots, X[1] + 1$ do
    XGBoost($X_{\text{train}\_s}, y_{\text{train}}$);
    if $a > b\_a$ then
        $b\_a \leftarrow a$;
        $b\_n\_f \leftarrow n\_f$;
        EarlyStopping
    end
end
$D' \leftarrow \text{SMOTE}(X_{\text{train}\_s}, y_{\text{train}})$;
$i \leftarrow |D'|$;
Initsamples($O, W$);
for $i = 1, 2, \ldots, N$ do
    $p(f|w, D') \leftarrow \text{XGBoost}(D')$;
    $w_i \leftarrow \arg\max_{w \in W} F(w, p(f|w, D'))$;
    $f_i \leftarrow O(w_i)$;
    $D' \leftarrow D' \cup (w_i, f_i)$
end
$\hat{y} = \arg\max_{k \in \{0,\ldots,K-1\}} \sum_{m=1}^{M} w_m 1(T_m(x) = k)$
ROSE-BOX($D'$) $\leftarrow$ BO-XGBoost($D'$)
```

The detailed procedure of the Bayesian optimization algorithm for XGBoost is elaborated as follows.

As shown in Algorithm 1, N represents the number of iterations for the Bayesian optimization process. Given a set of parameters for XGBoost, denoted as

$$W = \{w_1, w_2, \ldots, w_n\}, w \to R \tag{9}$$

where each w maps to a real number R, and an objective function O, we commence by pairing samples (O, W) to compile a dataset comprising several data pairs (w, f). Additionally, a function F is harnessed to identify the next observation point for evaluation [33].

Central to this approach is the calculation of the posterior probability $p(f|w, D')$, which represents the likelihood of the forthcoming parameter set. This is accomplished by fitting XGBoost with the dataset D'.

Moreover, by utilizing the acquisition function F, we determine the maximum value w_i grounded on the current posterior probability and parameter w. Consequentially, the outcome related to the parameter f_i is derived by applying $O(w_i)$. The process iteratively updates the dataset D' until the optimal parameters are detected.

The culmination of this iterative optimization is the identification of the ideal parameter set $w*$, achieved by maximizing the objective function, which is depicted as follows:

$$w* = \arg\max_{w \in W} O(w) \tag{10}$$

To aggregate predictions from the Bayesian-optimized XGBoost (BO-XGBoost), ROSE-BOX applies a weighted majority-voting consensus mechanism. Let $T_1, ..., T_M$ be the set of M trees, and let w_m denote the weight of tree T_m, proportional to its classification accuracy on the validation set. For each sample x, the predicted class $\hat{y}$ is

$$\hat{y} = \arg \max_{k \in \{0,\ldots,K-1\}} \sum_{m=1}^{M} w_m 1(T_m(x) = k) \tag{11}$$

where $1(\cdot)$ is the indicator function. If the mth tree classifies x into class k, it is equal to 1. Finally, an effective intrusion detection approach, ROSE-BOX, is proposed to detect multi-class attacks based on Random Forest, SMOTE, and BO-XGBoost in IIoT.

4. Experimental Results

In this section, we provide the experimental details and multiple experimental results. Firstly, we introduce the specific details of the IIoT datasets. Secondly, we describe the evaluation indicators of the detection models. Moreover, we present the results of feature selection using Random Forest and XGBoost, providing the optimal parameter combination results after Bayesian optimization. Finally, we provide an analysis of the detection results.

4.1. Datasets

After data preprocessing, three publicly available IIoT-related intrusion detection datasets were employed in this study:

- **CIC-IDS2017**: The CIC-IDS2017 dataset simulates benign and attack traffic during normal working hours, capturing common threats such as DoS, PortScan, and Brute Force attacks. The total number of flow records after preprocessing is 692,703.
- **CSE-CIC-IDS2018**: The CSE-CIC-IDS2018 dataset contains traffic from various realistic attack scenarios, including infiltration, botnets, and web attacks. It comprises 1,048,574 flow records and was processed using CICFlowMeter to extract standardized network features.
- **CIC-DDoS2019**: The CIC-DDoS2019 dataset is designed to capture DDoS attack patterns targeting multiple protocols (e.g., HTTP, TCP, UDP); this dataset reflects high-volume traffic characteristics. After preprocessing, it includes 431,371 records.

The distribution of class labels across these datasets is summarized in Table 2. In this context, the Benign label denotes normal traffic, whereas all other labels are considered anomalous. To address the prevalent issue of class imbalance, where benign traffic significantly outweighs attack samples, the SMOTE (Synthetic Minority Oversampling Technique) [13] algorithm was applied. Post-processing, the number of samples in each class was balanced to match the count of the majority class, ensuring an equal representation of all categories during model training and evaluation.

Table 2. The composition of CIC-IDS2017, CSE-CIC-IDS2018, and CIC-DDoS2019.

Dataset	Label	Size
CIC-IDS2017	BENIGN	440,031
	DoS Hulk	231,073
	DoS GoldenEye	10,293
	DoS slowloris	5796
	DoS Slowhttptest	54,969
	Heartbleed	11
CSE-CIC-IDS2018	DoS attacks-Hulk	461,912
	BENIGN	446,772
	DoS attacks-SlowHTTPTest	139,890
CIC-DDoS2019	DrDoS_NTP	121,368
	TFTP	98,917
	Benign	97,831
	Syn	49,373
	UDP	18,090
	DrDoS_UDP	10,420
	UDP-lag	8872
	MSSQL	8523
	DrDoS_MSSQL	6212
	DrDoS_DNS	3669
	DrDoS_SNMP	2717
	LDAP	1906
	DrDoS_LDAP	1440
	Portmap	685
	NetBIOS	644
	DrDoS_NetBIOS	598
	UDPLag	55
	WebDDoS	51

4.2. Evaluation Metrics

The confusion matrix and ROC (Receiver Operating Characteristic) curve are considered evaluation criteria for anomaly detection. Importantly, TP denotes the number of normal samples correctly detected, while FN denotes the number of samples incorrectly detected as abnormal. TN denotes the number of correctly detected abnormal samples, while FP denotes the number of samples that were incorrectly detected as normal. Incidentally, the AUROC value is the area size below the ROC curve. It is worth mentioning that *accuracy*, *F1 score*, *recall*, and *precision* can be calculated from the confusion matrix, as shown below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

$$F1Score = \frac{2 \times Precion \times Recall}{Precision + Recall} \tag{13}$$

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

4.3. The Results of Feature Selection and Bayesian Optimization

Figure 2 presents an in-depth analysis of the feature selection strategy and its influence on intrusion detection performance across three widely used benchmark datasets, CIC-IDS2017, CSE-CIC-IDS2018, and CIC-DDoS2019, by using the XGBoost-assisted Random

Forest, which enables a synergistic approach to both feature selection and detection. Random Forest ranks the importance of features, and XGBoost performs intrusion detection on the features with the highest importance rankings.

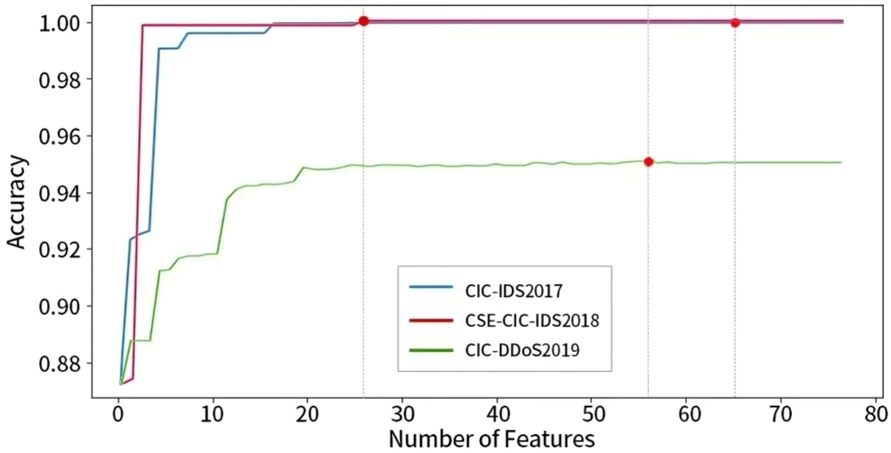

Figure 2. Relationship between the number of selected features and detection accuracy using Random Forest for feature selection and XGBoost for intrusion detection.

The comparative evaluation across datasets demonstrates that the method effectively reduces feature dimensionality while preserving—and, in some cases, enhancing—detection accuracy. Specifically, the relationship between the number of selected features and XGBoost's classification accuracy reveals that, for CIC-IDS2017, the accuracy peaks at 99.982% when 65 top-ranked features are utilized. In the case of CSE-CIC-IDS2018, the model achieves 100% accuracy with only 26 features, underscoring its ability to isolate a compact and highly discriminative subset of features. Similarly, on CIC-DDoS2019, selecting the 56 most informative features results in the highest accuracy of 95.042%. These findings illustrate the model's robustness in identifying critical attributes while eliminating redundant or irrelevant information.

The visualization of feature ranking for datasets CIC-IDS2017, CSE-CIC-IDS2018, and CIC-DDoS2019 are shown in Figures 3–5. The results emphasize the model's focus on a concise set of high-impact features, which improves computational efficiency and further enhances intrusion detection performance. This is particularly advantageous in resource-constrained IIoT scenarios, where real-time processing and low-latency response are essential.

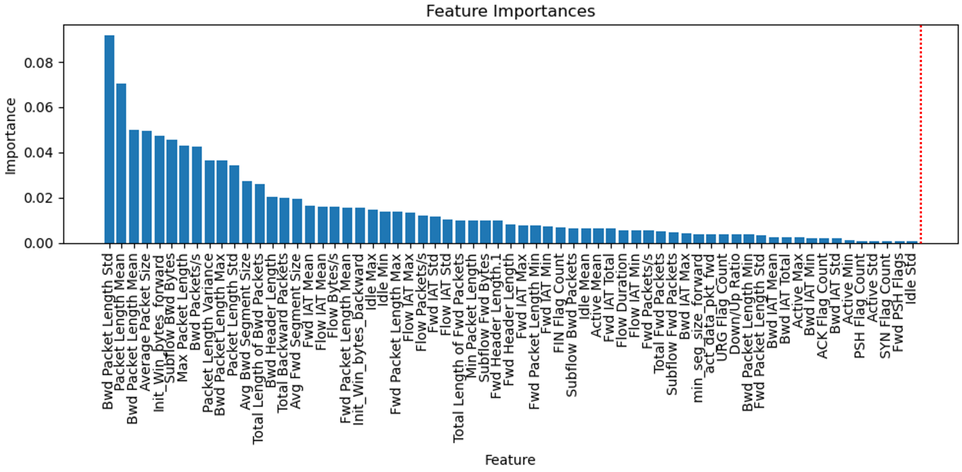

Figure 3. Feature importance of CIC-IDS2017.

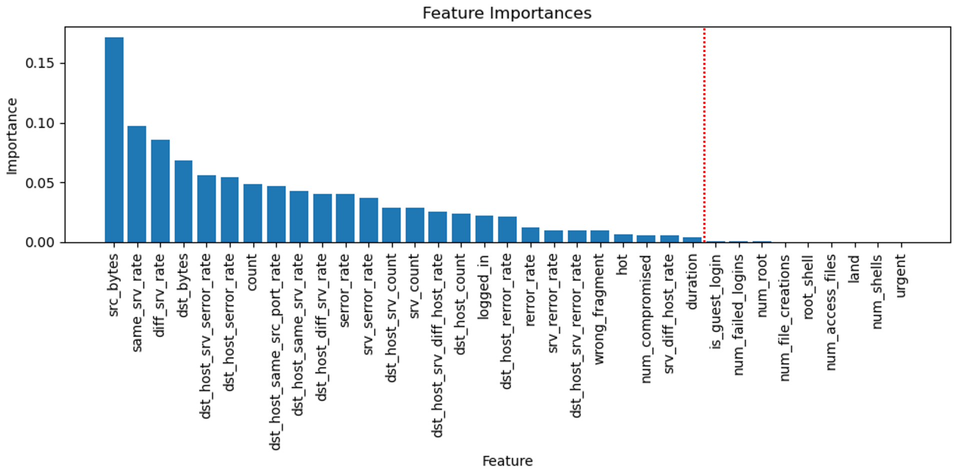

Figure 4. Feature importance of CSE-CIC-IDS2018.

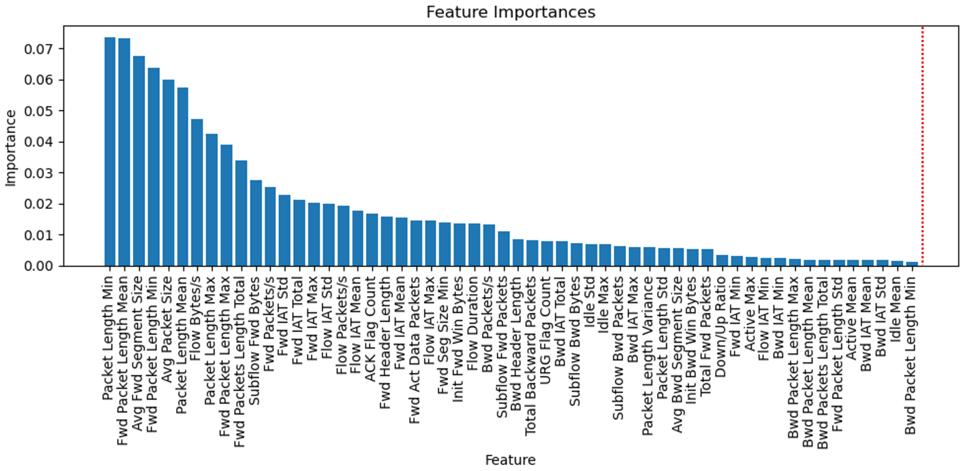

Figure 5. Feature importance of CIC-DDoS2019.

The integration of Random Forest for feature ranking and XGBoost for multi-class intrusion detection enables the model to accurately prioritize the most influential features, thereby reducing noise and improving both computational efficiency and detection effectiveness. The application of EarlyStopping further ensures that the model avoids overfitting by retaining only the most relevant features during training. In addition, the incorporation of SMOTE for oversampling addresses the prevalent issue of class imbalance in cybersecurity datasets, where malicious instances are typically under-represented. This mechanism enhances the model's sensitivity to anomalies in the minority class, thereby reducing the risk of misclassifying rare yet critical attack patterns.

After using XGBoost-assisted Random Forest for feature selection, we performed Bayesian optimization on XGBoost. Since the detection rate of CSE-CIC-IDS2018 using XGBoost reached 100%, it does not require subsequent Bayesian optimization. After five-fold cross-validation and 20 iterations of Bayesian optimization, the optimal parameter combinations for detecting CIC-IDS2017 and CIC-DDoS2019 using BO-XGBoost were obtained and are shown in Table 3.

Table 3. The optimal parameter combinations for detecting CIC-IDS2017 and CIC-DDoS2019 using BO-XGBoost.

Parameter	Range	CIC-IDS2017	CIC-DDoS2019
learning_rate	(0.01, 1.0)	0.46206	0.2854
min_child_weight	(0, 10)	10	3
max_depth	(3, 10)	10	10
subsample	(0.5, 1.0)	0.93196	0.86975
colsample_bytree	(0.5, 1.0)	0.72514	0.70505
n_estimators	(50, 300)	300	218
reg_alpha	$(1 \times 10^{-9}, 1.0)$	4.23191	5.33418
reg_lambda	$(1 \times 10^{-9}, 100)$	0.00068	7.24844

4.4. Baseline Methods for Intrusion Detection

The proposed ROSE-BOX framework is evaluated against nine other intrusion detection methods. Belief descriptions of the baseline methods used for the experiments are listed as follows.

- **KNN** [24]: K-Nearest Neighbors (KNNs) classifies new traffic instances by measuring similarities (e.g., Euclidean distance) to labeled examples, making it straightforward to detect anomalies in IIoT flows.
- **RF** [25]: Random Forest (RF) leverages ensemble learning with multiple decision trees through majority voting, mitigates overfitting via feature randomness and bootstrap aggregation (bagging), and provides robust anomaly detection by evaluating feature importance.
- **AdaBoost** [10]: Adaptive Boosting (AdaBoost) builds a strong classifier by sequentially weighting misclassified samples and combining weak learners (typically decision stumps), which improves the detection of rare attack types in imbalanced IIoT datasets.
- **DNN** [26]: Deep neural networks (DNNs) leverage multiple fully connected layers to connect a large number of neurons to learn hierarchical representations of network features, enabling the detection of complex attack signatures.
- **LSTM** [11]: Long short-term memory (LSTM) is a recurrent architecture designed to capture long-range temporal dependencies in network traffic.
- **Transformer** [34]: Transformer embeds each categorical flow attribute into dense vectors and processes them through Transformer encoder layers to produce contextualized feature representations, which are then concatenated with numeric features and passed to an MLP that discriminates among multiple attack and normal classes.
- **CNN-LSTM** [7]: CNN-LSTM combines convolutional neural networks (CNNs) for spatial feature extraction (e.g., packet-level patterns) with LSTM layers for temporal analysis, yielding enhanced detection accuracy in IIoT attack classification.
- **XGBoost** [12]: eXtreme Gradient Boosting (XGBoost) is an optimized gradient-boosted tree ensemble that incorporates regularization, sparsity awareness, and parallel tree construction, consistently achieving state-of-the-art accuracy in multi-class IIoT intrusion detection benchmarks.
- **XGB-RF** [35]: XGBoost with RF (XGB-RF) is the proposed ROSE-BOX model without SMOTE and Bayesian optimization, that is, XGBoost-assisted Random Forest, as mentioned above. When RF is used to rank features for the IIoT datasets, XGBoost applies regularized gradient-boosted trees on the selected subset for high-precision anomaly detection simultaneously.

4.5. Analysis of Detection Results

Our simulations were conducted using Python 3.9, leveraging libraries such as PyTorch 1.11 for model implementation and training and scikit-learn 1.5.2 for evaluation metrics. The experiments were run on a system equipped with an Intel i7 processor and 16 GB of RAM.

As depicted in Tables 4–6, the proposed ROSE-BOX algorithm demonstrates intrusion detection capabilities across three cybersecurity benchmark datasets, CIC-IDS2017, CSE-CIC-IDS2018, and CIC-DDoS2019, achieving superior detection accuracy and operational efficiency compared to contemporary machine learning paradigms. In order to more intuitively compare the intrusion detection performance of ROSE-BOX with other methods on the three datasets, we used bar charts to visualize their accuracy, precision, recall, F1 score, and AUROC value, as shown in Figure 6, Figure 7, and Figure 8, respectively. Through comprehensive evaluation against nine baseline methodologies spanning classical ensemble learning (Random Forest [25], AdaBoost [10], and XGBoost [12]), deep neural architectures (DNN [26], CNN-LSTM [12], and Transformer [7]), and hybrid approaches (XGB-RF [35]), our framework exhibits enhanced robustness in identifying zero-day attacks while maintaining sub-second inference latency. Particularly noteworthy is its consistent performance advantage over temporal models, such as LSTM [11], and proximity-based classifiers (KNN [24]), establishing ROSE-BOX as a versatile solution for real-time network intrusion detection across diverse threat landscapes.

Table 4. Comparison of the ROSE-BOX model with other methods for detecting CIC-IDS2017.

Algorithm	***Accuray***	***Precision***	***Recall***	***F1 Score***	***AUROC***
KNN [24]	99.39%	98.733%	99.003%	98.867%	0.99409
RF [25]	99.915%	99.643%	99.787%	99.715%	0.9988
AdaBoost [10]	77.72%	30.355%	38.4%	29.796%	0.67736
DNN [26]	98.958%	93.342%	99.243%	95.732%	0.99953
LSTM [11]	83.126%	34.383%	30.966%	31.859%	0.73295
Transformer [34]	98.767%	98.582%	98.739%	98.651%	0.99947
CNN-LSTM [7]	98.909%	96.51%	99.223%	97.876%	0.99895
XGBoost [12]	99.963%	99.733%	99.784%	99.759%	0.99787
XGB-RF [35]	99.982%	99.891%	99.804%	99.772%	0.99989
ROSE-BOX (Ours)	99.991%	99.993%	99.987%	99.989%	1.0

Table 5. Comparison of the ROSE-BOX model with other methods for detecting CSE-CIC-IDS2018.

Algorithm	***Accuray***	***Precision***	***Recall***	***F1 Score***	***AUROC***
KNN [24]	99.785%	99.837%	99.833%	99.835%	0.99999
RF [25]	99.999%	99.999%	99.999%	99.999%	0.99999
AdaBoost [10]	99.823%	99.866%	99.862%	99.864%	0.99999
DNN [26]	99.837%	99.877%	99.872%	99.875%	0.99918
LSTM [11]	13.602%	40.89%	33.534%	12.853%	0.74749
Transformer [34]	99.84%	99.88%	99.875%	99.877%	0.99967
CNN-LSTM [7]	99.102%	98.353%	99.624%	98.91%	0.99979
XGBoost [12]	99.999%	99.999%	99.999%	99.999%	0.99999
ROSE-BOX (Ours)	100%	100%	100%	100%	1.0

Table 6. Comparison of the ROSE-BOX model with other methods for detecting CIC-DDoS2019.

Algorithm	*Accuray*	*Precision*	*Recall*	*F1 Score*	*AUROC*
KNN [24]	90.82%	47.829%	45.243%	46.095%	0.73662
RF [25]	92.636%	48.948%	48.549%	48.682%	0.74064
AdaBoost [10]	65.706%	29.818%	29.429%	25.381%	0.78124
DNN [26]	93.192%	50.796%	50.335%	47.265%	0.99223
LSTM [11]	59.624%	15.618%	15.87%	14.194%	0.77945
Transformer [34]	93.422%	53.859%	50.672%	46.547%	0.95346
CNN-LSTM [7]	99.814%	58.8%	57.566%	57.278%	0.99916
XGBoost [12]	94.166%	53.989%	54.988%	56.987%	0.78616
XGB-RF [35]	95.042%	61.145%	60.212%	62.019%	0.81572
ROSE-BOX (Ours)	99.851%	63.563%	65.687%	64.989%	0.99395

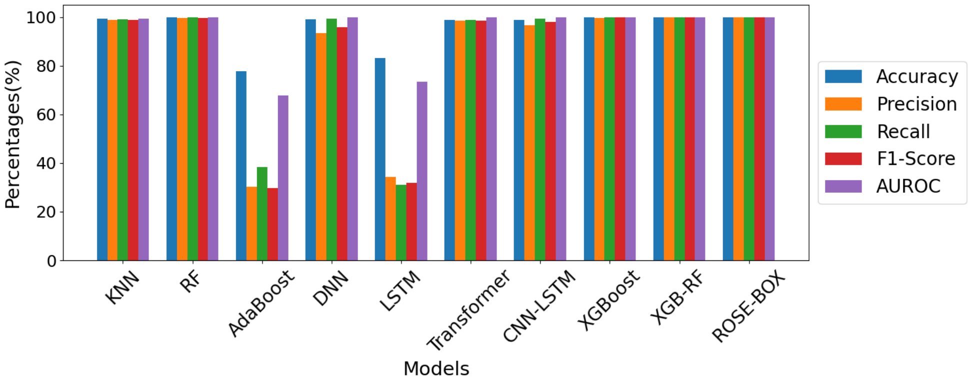

Figure 6. Comparison of ROSE-BOX with other models using CIC-IDS2017.

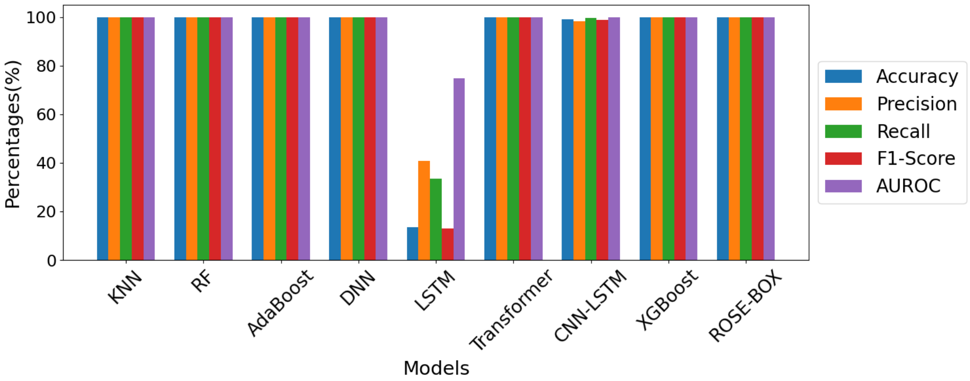

Figure 7. Comparison of ROSE-BOX with other models using CSE-CIC-IDS2018.

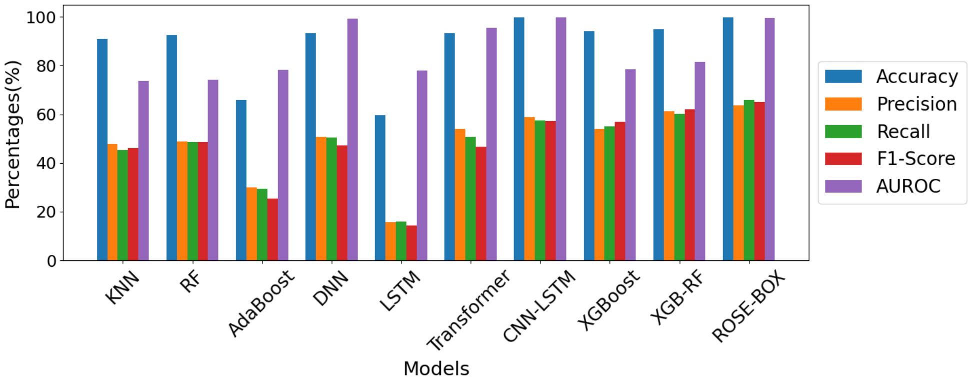

Figure 8. Comparison of ROSE-BOX with other models using CIC-DDoS2019.

As shown in Table 4, the AUROC of ROSE-BOX is 1.0, indicating that the model effectively distinguishes between normal and anomalous instances, which provides a clear view of classification performance across six classes (Class 0–5) on the CIC-IDS2017 dataset, demonstrating an almost flawless match between true and predicted labels. Table 4 shows the specific detection metrics, where ROSE-BOX achieves an accuracy of 99.991%, a precision of 99.993%, a recall of 99.987%, and an F1 score of 99.989%. These near-perfect metrics confirm that ROSE-BOX consistently outperforms other models across all evaluation criteria. Methods such as KNN, DNN, LSTM, CNN-LSTM, and Transformer struggle with the high dimensionality and complexity of IIoT data. In contrast, XGB-RF (XGBoost-assisted Random Forest for feature selection) has better detection performance than using XGBoost alone, and its detection performance without SMOTE and Bayesian optimization (XGB-RF) is inferior to ROSE-BOX.

As shown in Table 5, ROSE-BOX achieves perfect classification, with an AUROC of 1.0, indicating ideal detection performance and showing perfect classification across four classes (Class 0–3) on the CSE-CIC-IDS2018 dataset. Table 5 highlights that the model reached 100% accuracy, precision, recall, and F1 score, demonstrating flawless performance across all key metrics. Therefore, the ROSE-BOX model is equal to the XGB-RF model at this time. The performance of the proposed model surpasses that of XGBoost without feature selection. Traditional ML-based methods, such as KNN and AdaBoost, often struggle to adapt to the dynamic nature of real-time industrial data. In contrast, deep learning models, including DNN, LSTM, and CNN-LSTM, although powerful, can require significant computational resources, thereby limiting their practical applicability. ROSE-BOX, with its efficient feature selection and optimization, offers a balance between accuracy and computational feasibility, making it more practical for IIoT applications.

As shown in Figure 9, the ROC curve, which spans 18 classes (Class 0–17), shows the strong detection performance of ROSE-BOX, with an AUROC of 0.99395, indicating the model's high ability to separate various types of network intrusion attacks from normal traffic on the CIC-DDoS2019 dataset. Table 6 reveals that ROSE-BOX achieved an accuracy of 99.851%, with a precision of 63.563%, a recall of 65.687%, and an F1 score of 64.989%. While the precision and recall metrics are slightly lower than those in the previous datasets, this result is still superior to other methods, especially when considering the complex and imbalanced nature of DDoS attacks. Models such as DNN, LSTM, CNN-LSTM, and Transformer tend to struggle with this level of complexity, while simpler methods, like KNN or RF, lack the adaptability to capture the nuanced patterns in high-dimensional data. Although XGB-RF achieves outstanding performance, particularly on CIC-IDS2017 and CSE-CIC-IDS2018, its detection performance slightly degrades in highly imbalanced scenarios, such as for CIC-DDoS2019, as shown in Table 6. The primary reason is that XGB-RF does not incorporate SMOTE to handle class imbalance, nor does it employ Bayesian optimization to fine-tune its parameters. These two mechanisms, central to ROSE-BOX, allow it to maintain high recall for rare attacks and optimize performance dynamically across varying network conditions. Thus, ROSE-BOX outperforms XGB-RF and XGBoost in recall and F1 score in datasets with more complex or imbalanced structures.

Across three standard IIoT intrusion detection benchmarks, ROSE-BOX attains record-setting overall accuracies of 99.991% on CIC-IDS2017, 100% on CSE-CIC-IDS2018, and 99.851% on CIC-DDoS2019, surpassing the strongest competing algorithms by margins of 0.008 to 9.2 percentage points. This exceptional performance arises from the synergistic integration of XGBoost-assisted Random Forest feature ranking to eliminate irrelevant dimensions, SMOTE-mediated minority-class augmentation to mitigate label imbalance, Bayesian optimization for principled hyperparameter calibration, and an adaptive EarlyStopping criterion to prevent overfitting. Together, these components furnish ROSE-BOX with both highly discriminative modeling capacity and rapid, resource-efficient convergence, rendering it ideally suited for real-time deployment in resource-constrained IIoT environments.

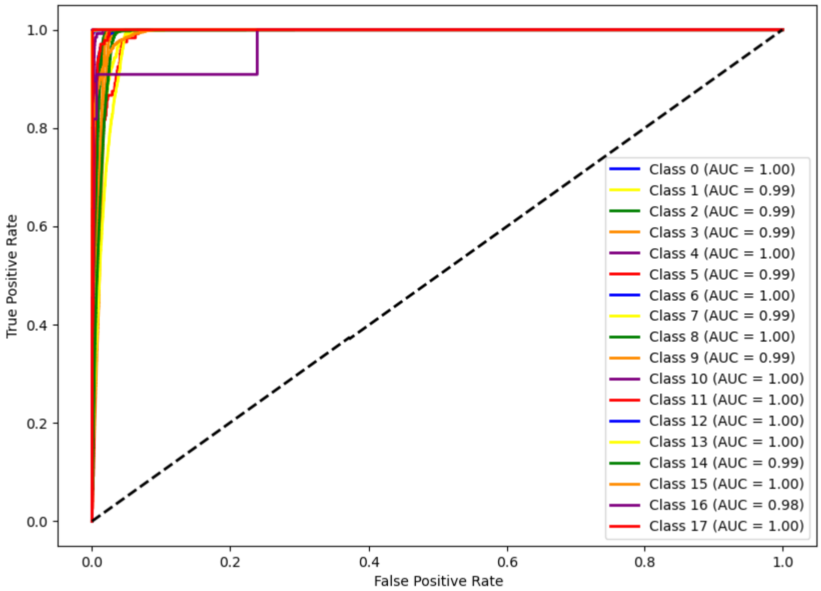

Figure 9. The ROC curve of the ROSE-BOX algorithm for anomaly detection in CIC-DDoS2019.

In summary, ROSE-BOX demonstrates clear superiority over ML-based and DL-based methods. On the other hand, it successfully addresses these challenges by balancing multi-class samples through SMOTE and fine-tuning model hyperparameters with Bayesian optimization, which ensures more precise detection even in complex IIoT scenarios of multi-class intrusion detection. This makes it particularly well-suited for resource-constrained IIoT environments.

4.6. Complexity Analysis

In order to further analyze the performance of the proposed model, we conducted a complexity comparison analysis of the proposed model, including detection time for each sample and CPU occupancy rate. Comparisons of the complexity of the proposed model with other models for CIC-IDS2017, CSE-CICIDS2018, and CIC-DDoS2019 are shown in Figure 10, Figure 11, and Figure 12, respectively.

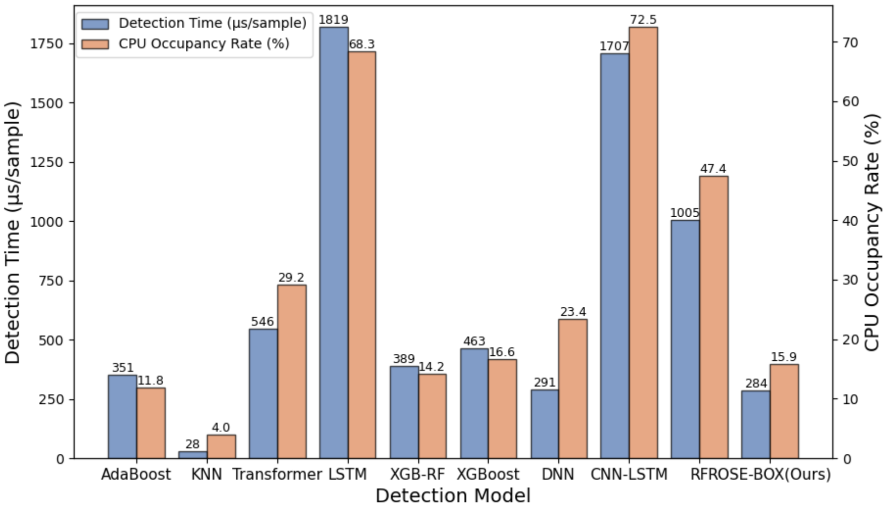

Figure 10. Comparison of the complexity of the proposed model with other models for the CIC-IDS2017 dataset.

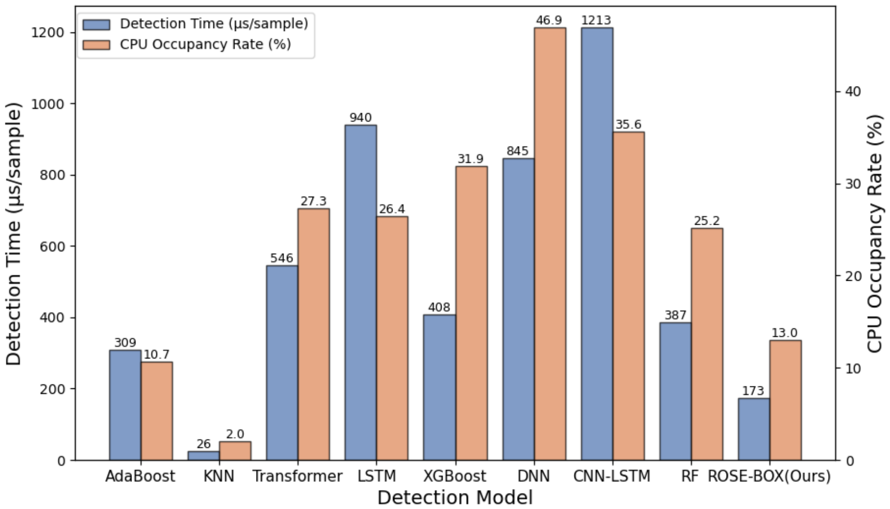

Figure 11. Comparison of the complexity of the proposed model with other models for the CSE-CIC-IDS2018 dataset.

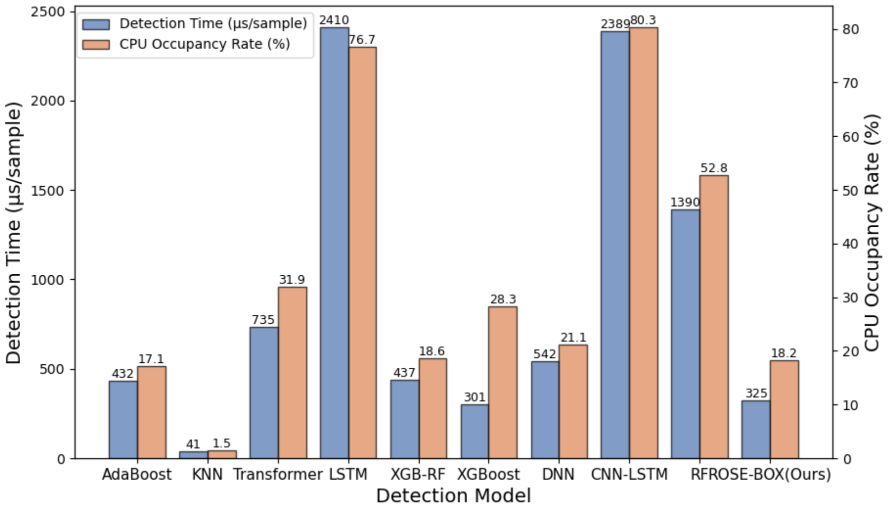

Figure 12. Comparison of the complexity of the proposed model with other models for the CIC-DDoS2019 dataset.

The complexity comparison results for CIC-IDS2017, CSE-CICIDS2018, and CIC-DDoS2019 demonstrate that ROSE-BOX significantly outperforms other models in both detection speed and resource efficiency. The experimental results indicate that ROSE-BOX achieves the lowest detection time per sample across all datasets, with 284 μs for CIC-IDS2017, 173 μs for CSE-CICIDS2018, and 325 μs for CIC-DDoS2019. These values are considerably lower than LSTM, DNN, and CNN-LSTM, which exhibit much higher detection latencies. Additionally, ROSE-BOX consistently achieves the lowest CPU occupancy rate, maintaining 15.9% for CIC-IDS2017, 13.0% for CSE-CICIDS2018, and 18.2% for CIC-DDoS2019. At the same time, deep learning models like LSTM, DNN, and CNN-LSTM consume significantly more computational resources, often exceeding 26% to 81% CPU usage. Although KNN demonstrates respectable accuracy and a low training cost, its detection effect is poor relative to ROSE-BOX due to memory-based inference and poor adaptability in high-dimensional feature spaces. Therefore, its scalability and real-time performance are limited. Furthermore, through improvement, ROSE-BOX can significantly reduce latency and CPU usage compared to XGBoost and XGB-RF. These findings highlight the superior computational efficiency of ROSE-BOX in handling multi-class intrusion detection tasks in resource-constrained IIoT environments.

The key advantage of ROSE-BOX lies in its suitability for real-time intrusion detection in resource-constrained environments. Its low detection latency ensures rapid threat response, making it ideal for real-time cybersecurity applications such as IoT security, edge computing, and cloud-based IDS solutions. Unlike deep learning models, such as LSTM, which are computationally expensive and impractical for low-power devices, ROSE-BOX achieves an optimal balance between accuracy and efficiency, making it highly deployable in environments with limited processing power. Additionally, its low CPU footprint enhances scalability, allowing deployment in large-scale security frameworks without overloading system resources. These advantages position ROSE-BOX as an effective, lightweight, and real-time detection solution that is well-suited for modern cybersecurity challenges, where efficiency and speed are critical.

5. Conclusions and Future Direction

In this work, we proposed ROSE-BOX, a lightweight and efficient intrusion detection framework tailored to real-time cybersecurity applications in resource-constrained IIoT environments. Integrating XGBoost-assisted Random Forest for feature selection, SMOTE for class balancing, and Bayesian-optimized XGBoost for detection, ROSE-BOX achieves a compelling balance between detection accuracy and computational efficiency. The experimental results on CIC-IDS2017, CSE-CIC-IDS2018, and CIC-DDoS2019 confirmed its superior performance, consistently exceeding 99.85% detection accuracy while significantly reducing latency and CPU usage, which is suitable for resource-constrained IIoT environments. In the future, we will collect datasets covering specific attack scenarios to further verify the effectiveness of our algorithm.

The key contributions of ROSE-BOX include the following:

- Feature selection optimization: Leveraging XGBoost-assisted Random Forest to select the most relevant features, reducing data dimensionality while maintaining high detection performance;
- Class Imbalance handling: Applying SMOTE to balance intrusion datasets, improving model robustness against rare but critical cyberattacks;
- Computational efficiency: Utilizing Bayesian optimization to fine-tune XGBoost, ensuring superior anomaly detection with minimal resource consumption;
- Scalability for IIoT applications: Demonstrating feasibility for deployment in industrial control systems, edge computing platforms, and embedded security solutions.

The ROSE-BOX framework also has certain limitations that should be considered for future improvements and practical deployments:

1. Feature Selection:
    - Limitation: The Random Forest and XGBoost-based feature selection might not generalize well to datasets with significantly different feature distributions or when new types of attacks are introduced;
    - Future Work: Further research is needed to develop more adaptive feature selection mechanisms that can dynamically adjust to evolving attack patterns and data distributions.
2. Class Imbalance Handling:
    - Limitation: SMOTE may introduce synthetic samples that do not fully capture the complexity of real-world attack instances. This could potentially lead to overfitting on synthetic data and reduced generalization to unseen data.

- Future work: Exploring advanced resampling techniques or hybrid methods that combine oversampling with other strategies (e.g., undersampling majority classes) could improve model robustness and generalization.

3. Computational Efficiency and Real-Time Deployment:
    - Limitation: The computational overhead of Bayesian optimization and XGBoost parameter tuning might still be prohibitive for extremely resource-constrained devices, such as low-power IoT sensors;
    - Future work: Investigating more lightweight optimization algorithms and model architectures that can further reduce computational requirements without compromising detection accuracy is essential.
4. Scalability and Deployment:
    - Limitation: The framework's scalability in large-scale IIoT environments with thousands of devices and high-frequency data streams has not been extensively tested. Certain types of attacks, such as zero-day, stealthy, multi-stage, and false data injection attacks, remain challenging due to their sophisticated nature and ability to evade detection.
    - Future work: Developing distributed implementations of ROSE-BOX; exploring federated learning integration, deployment on resource-constrained devices, and online learning mechanisms for evolving IIoT threats in industrial control systems; and enhancing ROSE-BOX with advanced detection techniques, continuous learning mechanisms, and context-aware feature selection processes to improve its robustness and adaptability to evolving threats.

In summary, while ROSE-BOX offers a promising solution for intrusion detection in resource-constrained IIoT environments, addressing these limitations can further enhance its practicality, scalability, and adaptability, making it a more robust and versatile tool for real-world cybersecurity applications.

Author Contributions: Conceptualization, S.P.; methodology, S.P.; software, S.P.; validation, Y.H., Z.G., and J.L.; formal analysis, Y.H.; investigation, Y.H.; resources, Z.G.; data curation, S.P.; writing—original draft preparation, S.P.; writing—review and editing, R.L. and L.L.; visualization, S.P.; supervision, Y.H., Z.G., and J.L.; project administration, Z.G.; funding acquisition, Z.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Shenzhen Science and Technology Program, grant number KJZD20240903103811016; the Major Key Project of PCL, grant number PCL2024A05; and the National Natural Science Foundation of China, grant number 62350710797.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Potentially applicable.

Data Availability Statement: CIC-IDS2017-Wednesday-workingHours.pcap-ISCX, accessed on 5 July 2017 https://www.unb.ca/cic/datasets/ids-2017.html; CSE-CIC-IDS2018-Friday-16-02-2018_TrafffcForML_CICFlowMeter.csv https://www.unb.ca/cic/datasets/ids-2018.html; CIC-DDoS2019 https://data.mendeley.com/datasets/ssnc74xm6r/1.

Acknowledgments: Part of this work has been published in 2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA). https://ieeexplore.ieee.org/document/10885136 accessed on 5 July 2017.

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

References

1. Mohammadian, H.; Ghorbani, A.A.; Lashkari, A.H. A gradient-based approach for adversarial attack on deep learning-based network intrusion detection systems. *Appl. Soft Comput.* **2023**, *137*, 110173. [CrossRef]
2. Ahmim, A.; Maazouzi, F.; Ahmim, M.; Namane, S.; Dhaou, I.B. Distributed denial of service attack detection for the internet of things using hybrid deep learning model. *IEEE Access* **2023**, *11*, 119862–119875. [CrossRef]
3. Ferrag, M.A.; Maglaras, L.; Moschoyiannis, S.; Janicke, H. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *J. Inf. Secur. Appl.* **2020**, *50*, 102419. [CrossRef]
4. Alani, M.M.; Awad, A.I. An intelligent two-layer intrusion detection system for the internet of things. *IEEE Trans. Ind. Inform.* **2023**, *19*, 683–692. [CrossRef]
5. Hindy, H.; Atkinson, R.; Tachtatzis, C.; Colin, J.N.; Bayne, E.; Bellekens, X. Utilising deep learning techniques for effective zero-day attack detection. *Electronics* **2020**, *9*, 1684. [CrossRef]
6. Ring, M.; Wunderlich, S.; Scheuring, D.; Landes, D.; Hotho, A. A survey of network-based intrusion detection data sets. *Comput. Secur.* **2019**, *86*, 147–167. [CrossRef]
7. Halbouni, A.; Gunawan, T.S.; Habaebi, M.H.; Halbouni, M.; Kartiwi, M.; Ahmad, R. Cnn-lstm: Hybrid deep neural network for network intrusion detection system. *IEEE Access* **2022**, *10*, 99837–99849. [CrossRef]
8. Dai, W.; Li, X.; Ji, W.; He, S. Network intrusion detection method based on cnn, bilstm, and attention mechanism. *IEEE Access* **2024**, *12*, 53099–53111. [CrossRef]
9. Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.P.; De Freitas, N. Taking the human out of the loop: A review of bayesian optimization. *Proc. IEEE* **2015**, *104*, 148–175. [CrossRef]
10. Yulianto, A.; Sukarno, P.; Suwastika, N.A. Improving adaboost-based intrusion detection system (ids) performance on cicids 2017 dataset. *J. Phys. Conf. Ser.* **2019**, *1192*, 012018. [CrossRef]
11. Jeong, H.W.; Kim, H.G.; Choi, Y.H. Lstm-based network intrusion detection system and solving data imbalance problem through gan. In Proceedings of the 2025 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Fukuoka, Japan, 18–21 February 2025; IEEE: Piscataway, NJ, USA, 2025; pp. 1156–1159.
12. Kang, Y.; Tan, M.; Lin, D.; Zhao, Z. Intrusion detection model based on autoencoder and xgboost. *J. Phys. Conf. Ser.* **2022**, *2171*, 012053. [CrossRef]
13. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. Smote: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
14. Peng, S.; Han, Y.; Liang, X.; Yang, C.; Gui, W.; Zhou, N. Rose-box: An approach for intrusion detection in industrial internet of things. In Proceedings of the 2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA), Kaifeng, China, 30 October–2 November 2024; pp. 2276–2277. [CrossRef]
15. Li, R.; Qin, Y.; Wang, C.; Li, M.; Chu, X. A Blockchain-Enabled Framework for Enhancing Scalability and Security in IIoT. *IEEE Trans. Ind. Inform.* **2023**, *19*, 7389–7400. [CrossRef]
16. Li, R.; Qin, Y.; Liu, J.; Chu, X.; Li, J. Multipath Based Congestion Propagation via Information Network Interaction in IIoT. *IEEE Trans. Ind. Inform.* **2024**, *20*, 8512–8523. [CrossRef]
17. Li, J.; Li, R.; Xu, L. Multi-stage deep residual collaboration learning framework for complex spatial–temporal traffic data imputation. *Appl. Soft Comput.* **2023**, *147*, 110814. [CrossRef]
18. Li, J.; Xu, L.; Li, R.; Wu, P.; Huang, Z. Deep Spatial-temporal Bi-directional Residual Optimisation based on Tensor Decomposition for Traffic data Imputation on Urban Road Network. *Appl. Intell.* **2022**, *52*, 11363–11381. [CrossRef]
19. Depren, O.; Topallar, M.; Anarim, E.; Ciliz, M.K. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. *Expert Syst. Appl.* **2005**, *29*, 713–722. [CrossRef]
20. Perdisci, G.; Roberto, G.G.; Lee, W. Using an Ensemble of One-Class SVM Classifers to Harden Payload-BasedAnomaly Detection Systems. In Proceedings of the 6th International Conference on Data Mining (ICDM'06), New York, NY, USA, 18–22 December 2006; pp. 488–498.
21. Amin, S.O.; Siddiqui, M.S.; Hong, C.S.; Choe, J. A novel coding scheme to implement signature based IDS in IP based sensor networks. In Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management–Workshops, New York, NY, USA, 1–5 June 2009; pp. 269–274.
22. Li, Z.; Li, X. Intrusion detection method based on genetic algorithm of optimizing lightgbm. In Proceedings of the 2021 5th International Conference on Electronic Information Technology and Com-puter Engineering, Kunming, China, 29–31 October 2021; Association for Computing Machinery: New York, NY, USA, 2022; pp. 1366–1371. [CrossRef]
23. Jemili, F.; Meddeb, R.; Korbaa, O. Intrusion detection based on ensemble learning for big data classification. *Clust. Comput.* **2024**, *27*, 3771–3798. [CrossRef]
24. Siddartha, V.S.; Nagalakshmi, T. Performance analysis of an intrusion detection system for wireless adhoc network in the detection of dos attack using k-means cluster and k-nn algorithm. In *AIP Conference Proceedings, Kanyakumari, India, 9–10 December 2021*; AIP Publishing: Melville, NY, USA, 21 November 2023.

25. Chua, T.H.; Salam, I. Evaluation of machine learning algorithms in network-based intrusion detection using progressive dataset. *Symmetry* **2023**, *15*, 1251. [CrossRef]
26. Vinayakumar, R.; Alazab, M.; Soman, K.P.; Poornachandran, P.; Al-Nemrat, A.; Venkatraman, S. Deep learning approach for intelligent intrusion detection system. *IEEE Access* **2019**, *7*, 41525–41550. [CrossRef]
27. Farhan, R.I.; Maolood, A.T.; Hassan, N. Performance analysis of flow-based attacks detection on cse-cic-ids2018 dataset using deep learning. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *20*, 1413–1418. [CrossRef]
28. Kanimozhi, V.; Jacob, T.P. Artificial intelligence based net-work intrusion detection with hyper-parameter optimization tuning on the realistic cyber dataset cse-cic-ids2018 using cloud computing. In Proceedings of the 2019 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 4–6 April 2019; pp. 0033–0036. [CrossRef]
29. Shieh, C.S.; Lin, W.W.; Nguyen, T.T.; Chen, C.H.; Horng, M.F.; Miu, D. Detection of unknown ddos attacks with deep learning and gaussian mixture model. *Appl. Sci.* **2021**, *11*, 5213. [CrossRef]
30. Chartuni, A.; Márquez, J. Multi-classifier of ddos attacks in computer networks built on neural networks. *Appl. Sci.* **2021**, *11*, 10609. [CrossRef]
31. Zolanvari, M.; Teixeira, M.A.; Jain, R. Effect of Imbalanced Datasets on Security of Industrial IoT Using Machine Learning. In Proceedings of the 2018 IEEE International Conference on Intelligence and Security Informatics (ISI), Miami, FL, USA, 8–10 November 2018; pp. 112–117. [CrossRef]
32. Imani, M.; Beikmohammadi, A.; Arabnia, H.R. Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels. *Technologies* **2025**, *13*, 88. [CrossRef]
33. Brochu, E.; Cora, V.; Freitas, N. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv* **2010**, arXiv:1012.2599.
34. Ke, D. Network intrusion detection based on feature selection and transformer. In Proceedings of the 2023 International Conference on Intelligent Communication and Computer Engineering (ICICCE), Changsha, China, 24–26 November 2023; pp. 23–28. [CrossRef]
35. Wang, Z.; Yuan, F.; Li, R.; Zhang, M.; Luo, X. Hidden as link prediction based on random forest feature selection and gwo-xgboost model. *Comput. Netw.* **2025**, *262*, 111164. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.