JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

Understanding Statistical and Temporal Representations for Large-Scale IoT DDoS Detection Through Ablation-Driven Analysis

Daniel Nomolas Wicaksono [1], De Rosal Ignatius Moses Setiadi [1,2,*], Ajib Susanto [1], Imanuel Harkespan [1], Mohamad Afendee Mohamed [3], and Aceng Sambas [3,4]

[1] Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang 50131, Indonesia; e-mail : danielnomolaswica@gmail.com; moses@dsn.dinus.ac.id; ajib.susanto@dsn.dinus.ac.id; harkespan@dsn.dinus.ac.id
[2] Research Group for Quantum Computing and Materials Informatics, Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang 50131, Indonesia
[3] Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Campus Besut, 22200, Terengganu, Malaysia; e-mail : mafendee@unisza.edu.my; acengsambas@unisza.edu.my
[4] Department of Mechanical Engineering, Universitas Muhammadiyah Tasikmalaya, Tamansari Gobras 46196 Tasikmalaya, Indonesia
* Corresponding Author : De Rosal Ignatius Moses Setiadi

Abstract: Recent Internet of Things (IoT) intrusion detection studies have reported near-perfect benchmark performance for Distributed Denial of Service (DDoS) detection, yet limited attention has been given to understanding how different traffic representations contribute to the detection process under highly imbalanced traffic conditions. This study presents an ablation-driven analysis to investigate the contribution of statistical and temporal representations for large-scale IoT DDoS detection using the CICIoT2023 dataset. Three experimental scenarios are evaluated, including statistical representation, temporal sequence representation, and hybrid statistical–temporal representation. Temporal representations are learned using a one-dimensional Convolutional Neural Network (1D-CNN) with lag-based traffic sequences, while ensemble tree-based classifiers are employed for final classification and representation analysis. In addition, multiple ablation configurations are designed to evaluate the impact of temporal dependency modeling and feature engineering strategies on detection performance. Experimental results show that statistical traffic representations remain highly effective for DDoS detection on CICIoT2023, achieving 99.36% accuracy and 99.31% weighted F1-score in the statistical representation scenario. Feature importance analysis further indicates that engineered statistical features contribute substantially more to the classification process than CNN-based temporal representations. Although temporal modeling captures sequential traffic behavior, its contribution is relatively limited and mainly acts as a complementary representation. Furthermore, the hybrid configuration produces only marginal improvements over the statistical representation alone. These findings highlight the importance of representation-level analysis for understanding the actual contribution of statistical and temporal modeling in modern IoT intrusion detection systems beyond relying solely on benchmark accuracy.

Keywords: Cybersecurity; Deep Learning for Cybersecurity; DDoS Detection; Explainable Machine Learning; Intrusion Detection System; IoT Security; Network Traffic Analysis; Representation Analysis.

Received: April, 29th 2026
Revised: May, 28th 2026
Accepted: May, 29th 2026
Published: May, 30th 2026

1. Introduction

The rapid growth of the Internet of Things (IoT) has accelerated the integration of billions of smart devices across various sectors, including industry, healthcare, transportation, and smart city infrastructures. The heterogeneous, resource-constrained, distributed, and continuously connected nature of IoT environments has also increased their exposure to

cybersecurity threats, particularly Distributed Denial of Service (DDoS) attacks. In recent years, IoT-based DDoS attacks have grown significantly due to the limited computational resources, weak security configurations, and massive scale of IoT devices, which make them vulnerable to exploitation as distributed botnets [1]–[3]. The impact of such attacks extends beyond network service disruption and may also lead to reduced system reliability, financial losses, and disruptions to critical infrastructure [4].

Along with the increasing complexity of attacks and the growing volume of IoT network traffic, research on machine learning- and deep learning-based Intrusion Detection Systems (IDS) has developed rapidly. Early approaches commonly employed classical machine learning algorithms such as Support Vector Machine (SVM), Random Forest, Decision Tree, and XGBoost, which rely on statistical network traffic features to distinguish between benign and malicious traffic [5]–[8]. These approaches are known for their computational efficiency and strong classification performance across various modern IDS benchmark datasets. Subsequently, the advancement of deep learning encouraged the adoption of architectures such as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Transformer models to capture more complex traffic patterns, particularly temporal dependencies and non-linear feature relationships [9]–[12].

Despite the continuous improvement of detection methods, many modern IDS studies still place greater emphasis on overall classification performance evaluation than on feature representation contribution analysis or model interpretability [4], [13], [14]. Many studies propose increasingly complex deep learning and hybrid architectures, while relatively limited attention has been given to systematically analyzing how different feature representations actually contribute to the attack detection process. However, understanding feature representation is important because model performance is influenced not only by architectural complexity, but also by the ability of data representations to capture attack characteristics effectively.

In the context of DDoS detection, statistical representations generally describe network traffic characteristics within a particular traffic snapshot, such as packet counts, TCP flag distributions, protocol ratios, and traffic density. These representations are widely used in classical machine learning approaches because they can directly capture traffic anomalies in an efficient manner [15]. In contrast, temporal representations attempt to model traffic dynamics over time through sequential patterns and historical dependencies among network packets. Temporal modeling has become increasingly popular in deep learning approaches because it is assumed to capture attack patterns that may not be observable from isolated statistical snapshots [16], [17].

Nevertheless, the contribution of statistical and temporal representations is still often overlooked in modern IDS research. Many recent studies report improved detection performance using temporal deep learning or hybrid architectures, although the contribution of each representation type is not always analyzed explicitly [1], [13], [17]. As a result, it remains unclear whether performance improvements are genuinely driven by temporal modeling capabilities or are still primarily dominated by statistical traffic characteristics. Furthermore, evaluations under highly imbalanced traffic conditions often emphasize overall weighted performance, while comparatively less attention is devoted to analyzing model behavior across minority attack classes [10], [18], [19].

Motivated by these limitations, this study focuses on analyzing the contribution of statistical and temporal representations for large-scale IoT DDoS detection. The study employs a hybrid statistical–temporal learning pipeline combining CNN-based temporal representation learning and XGBoost-based classification. CNN is utilized to learn latent temporal representations from sequential traffic patterns, while XGBoost is employed due to its strong performance and robustness in handling high-dimensional, non-linear, and highly imbalanced data. The combination of both approaches is not primarily intended to propose a novel hybrid architecture, but rather to evaluate how temporal representations contribute when combined with engineered statistical traffic features.

To systematically analyze the contribution of each representation, this study adopts an ablation-driven analysis through three primary experimental scenarios: statistical representation, temporal representation, and hybrid statistical–temporal representation. In addition, the study evaluates both weighted and macro-level performance under highly imbalanced conditions to provide additional insight into model behavior across different attack categories, as well as feature importance analysis to investigate the dominance of different feature

representations during the classification process. The main contributions of this study can be summarized as follows:

- Conducting a systematic analysis of statistical, temporal, and hybrid representations for large-scale IoT DDoS detection through an ablation-driven evaluation framework.
- Evaluating weighted and macro-level performance under highly imbalanced traffic conditions to provide additional insight into model behavior across different attack categories.
- Providing feature importance analysis to investigate the relative contribution of statistical and temporal representations in IoT DDoS detection.

The remainder of this paper is organized as follows. Section 2 reviews previous studies related to IoT DDoS detection, statistical and temporal representations in IDS, and the research gap addressed in this work. Section 3 describes the proposed analytical framework, dataset preprocessing, feature construction process, experimental scenarios, and ablation design. Section 4 presents the experimental results and representation analysis across different scenarios. Finally, Section 5 concludes the paper and discusses potential directions for future research.

2. Literature Review

2.1. IoT DDoS Detection Approaches

Research on DDoS attack detection has developed rapidly over the past decades alongside the increasing complexity of modern networks and the growth of IoT ecosystems. Early studies by Mirkovic and Reiher [20] as well as Hussain et al. [21] investigated the characteristics and classification of DDoS attacks by categorizing them based on attack sources, vectors, and system impact. Subsequently, Zargar et al. [22] demonstrated that traditional rule-based and signature-matching approaches have become increasingly limited in handling the scale and dynamics of modern network traffic. These limitations encouraged the transition toward machine learning- and deep learning-based approaches to improve attack detection capabilities in increasingly complex network environments.

During the early development of modern Intrusion Detection Systems (IDS), various classical machine learning approaches such as SVM, Decision Tree, Random Forest, and XGBoost were widely adopted due to their ability to handle high-dimensional data with relatively low computational complexity [23], [24]. These approaches generally rely on statistical network traffic features such as packet rate, flow duration, protocol distribution, packet size, and other flow-based statistics to distinguish between benign and malicious traffic. Such characteristics make statistical feature-based machine learning approaches particularly effective for detecting flood-based attacks that generate significant traffic changes.

In the context of IoT security, Ntayagabiri et al. evaluated multiple machine learning algorithms using the CICIoT2023 dataset and reported that ensemble and tree-based approaches such as Random Forest and XGBoost still achieve highly competitive performance [25]. Their study also highlighted that the major challenges in modern IDS research are not limited to classification accuracy, but also include class imbalance and the relevance of feature representations used during classification.

Among various machine learning algorithms, XGBoost has consistently emerged as one of the most competitive methods in IDS-related studies. Chen and Guestrin introduced XGBoost as a scalable and efficient gradient boosting algorithm for large-scale data processing [26]. In addition to strong classification performance, XGBoost provides effective regularization mechanisms, computational optimization, and feature importance analysis, making it widely adopted in highly imbalanced and multiclass IoT classification scenarios [24].

With the advancement of deep learning, research attention gradually shifted toward automatic representation learning. Various architectures such as CNN, LSTM, GRU, and Transformer models have been applied to capture non-linear patterns and temporal dependencies in network traffic. Deep learning approaches are generally considered capable of learning more complex traffic behaviors than conventional statistical approaches, particularly for dynamic and sequential network traffic data. Several hybrid CNN-LSTM studies reported that temporal modeling can help improve classification performance and reduce false positive rates in IoT DDoS detection [17], [27], [28]. In addition, Abbas et al. [29] showed that deep

learning approaches such as DNN, CNN, and RNN can also achieve near-99% accuracy for multiclass IoT attack classification. Another Hybrid Stacking approach combining Logistic Regression, Gaussian Naive Bayes, and Random Forest reported 99.00% accuracy on the CICIoT2023 dataset [23].

The consistently high classification performance reported in previous studies indicates that modern IDS systems are capable of achieving very high benchmark performance on several public datasets. However, these findings also suggest that performance improvements have started to become increasingly marginal, particularly when machine learning, deep learning, and hybrid approaches all achieve near-perfect accuracy under similar benchmark settings. Consequently, recent research attention should not only focus on improving benchmark accuracy, but also on understanding how models construct classification decisions and which feature representations contribute most significantly to attack detection.

2.2. Statistical and Temporal Feature Representations in IDS

Feature representation plays an important role in determining the capability of an Intrusion Detection System (IDS) to distinguish between benign and malicious network traffic. In this study, statistical and temporal representations refer to two different approaches for modeling network traffic characteristics during the classification process [6], [30]. Statistical representations describe traffic behavior using engineered traffic statistics extracted from a particular traffic snapshot or interval, whereas temporal representations model traffic evolution over time through sequential patterns and temporal dependencies among network flows.

Statistical representations are commonly constructed using network traffic statistics such as packet rate, flow duration, protocol ratio, TCP flag distribution, packet density, and other flow-based features [15]. These representations characterize network conditions within a specific interval without explicitly modeling temporal order between traffic instances. In modern IDS research, classical machine learning approaches frequently rely on statistical representations because they can directly capture anomalous traffic behavior with relatively low computational complexity. In DDoS scenarios, statistical anomalies such as packet surges, abnormal protocol dominance, and irregular flag distributions are often sufficiently informative for distinguishing benign traffic from attack traffic.

In contrast, temporal representations focus on the dynamic evolution of network traffic over time. These representations model sequential traffic relationships, gradual traffic pattern changes, and temporal dependencies that may not be adequately captured through isolated statistical snapshots. Temporal approaches are widely adopted in deep learning because they are capable of learning non-linear traffic behavior and dynamically evolving attack patterns [17], [31]. An illustration of the differences between statistical and temporal representations is presented in Figure 1.

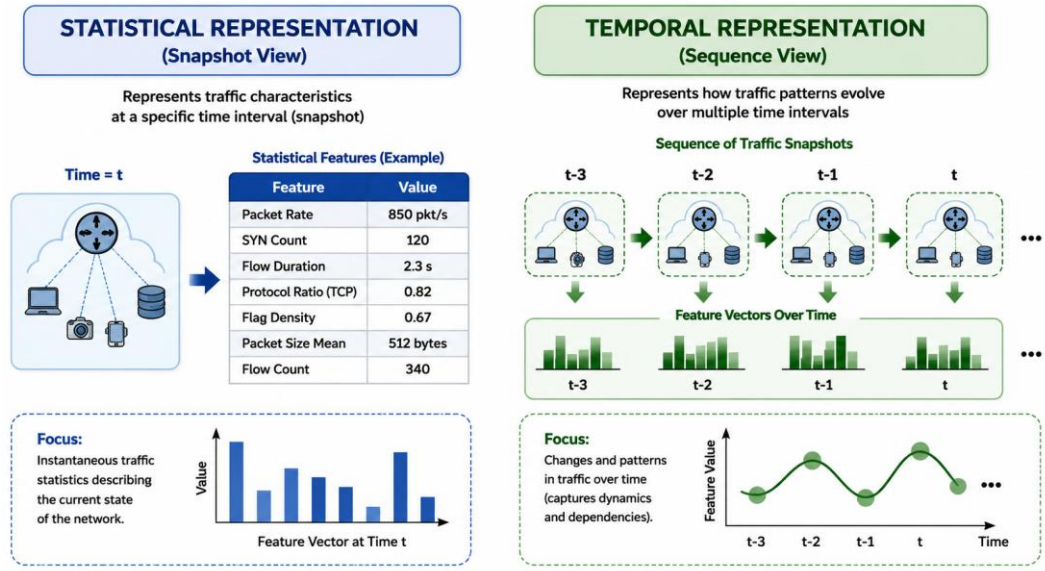

Figure 1. Illustration of the differences between statistical and temporal representation of features.

Several deep learning approaches, such as LSTM and GRU, are based on Recurrent Neural Network (RNN) architectures designed to learn short- and long-term temporal dependencies in sequential data. Beyond recurrent-based architectures, temporal modeling has also been increasingly explored using one-dimensional Convolutional Neural Networks (1D-CNN). Unlike two-dimensional CNNs in computer vision, which learn spatial locality in images, 1D-CNNs in IDS applications perform convolution operations over traffic sequences to capture local traffic patterns and temporal variations across consecutive intervals. This characteristic enables 1D-CNNs to learn latent traffic representations with lower training complexity than recurrent architectures, particularly for large-scale traffic data [32].

In modern IDS systems, statistical and temporal representations are generally considered complementary [33]. Statistical representations are effective for directly capturing traffic anomalies, while temporal representations are useful for modeling the sequential evolution of network behavior. Consequently, many recent studies have explored hybrid approaches that combine both representation types within a unified classification framework. Most recent hybrid IDS studies primarily report overall detection performance, while representation-level contribution analysis and feature dominance investigations remain relatively limited. Although many deep learning and hybrid models report very high weighted performance, this does not necessarily indicate that all feature representations contribute equally to the classification process. Furthermore, analyses related to feature dominance and representation-level importance remain relatively limited in modern IDS research. Understanding the contribution of different representations is important not only for identifying which features dominate attack detection, but also for evaluating the practical effectiveness of hybrid approaches in IoT-based IDS systems.

2.3. Research Gap and Motivation

Although machine learning, deep learning, and hybrid IDS approaches have achieved very high benchmark performance for IoT DDoS detection, most existing studies still primarily emphasize evaluation metrics such as accuracy and weighted F1-score. Consequently, relatively limited attention has been given to understanding how different feature representations contribute to the classification process, particularly in large-scale IoT environments characterized by highly imbalanced traffic distributions. In many studies, hybrid architectures combining statistical and temporal modeling are implicitly assumed to improve detection capability, yet the actual contribution of each representation type often remains insufficiently analyzed. As a result, it is still unclear whether the performance gains reported by hybrid and deep learning approaches genuinely originate from temporal representation learning or are still largely dominated by statistical traffic characteristics. Moreover, very high weighted performance does not necessarily indicate robust detection capability across all attack categories, especially for minority classes with substantially fewer samples than dominant traffic classes.

In addition to the limited analysis of representation contribution, studies examining feature dominance and feature importance in modern IDS systems also remain relatively scarce. Understanding the relative contribution of statistical and temporal representations is important not only for interpreting model behavior, but also for evaluating the practical effectiveness of hybrid approaches in IoT-based intrusion detection. Motivated by these limitations, this study investigates the contribution of statistical, temporal, and hybrid representations for large-scale IoT DDoS detection through an ablation-driven evaluation framework. The study evaluates multiple representation scenarios under highly imbalanced traffic conditions and further analyzes the relationship between weighted performance and macro-level robustness. In addition, feature importance analysis is conducted to examine the relative dominance of different feature representations and to provide a more detailed understanding of how representation learning contributes to the attack detection process in modern IoT IDS systems.

3. Proposed Framework and Experimental Design

3.1. Overall Framework of the Proposed Method

This study employs an ablation-driven analytical framework to investigate the contribution of statistical and temporal representations for large-scale IoT DDoS detection. Unlike many previous studies that primarily focus on improving final benchmark performance, this

work emphasizes representation-oriented analysis through a comparative evaluation of statistical representations, temporal sequence representations, and hybrid statistical–temporal representations. The overall framework consists of three main stages: (1) feature representation construction, (2) temporal representation learning and classification, and (3) scenario-based evaluation and ablation analysis. The complete analytical framework is illustrated in Figure 2.

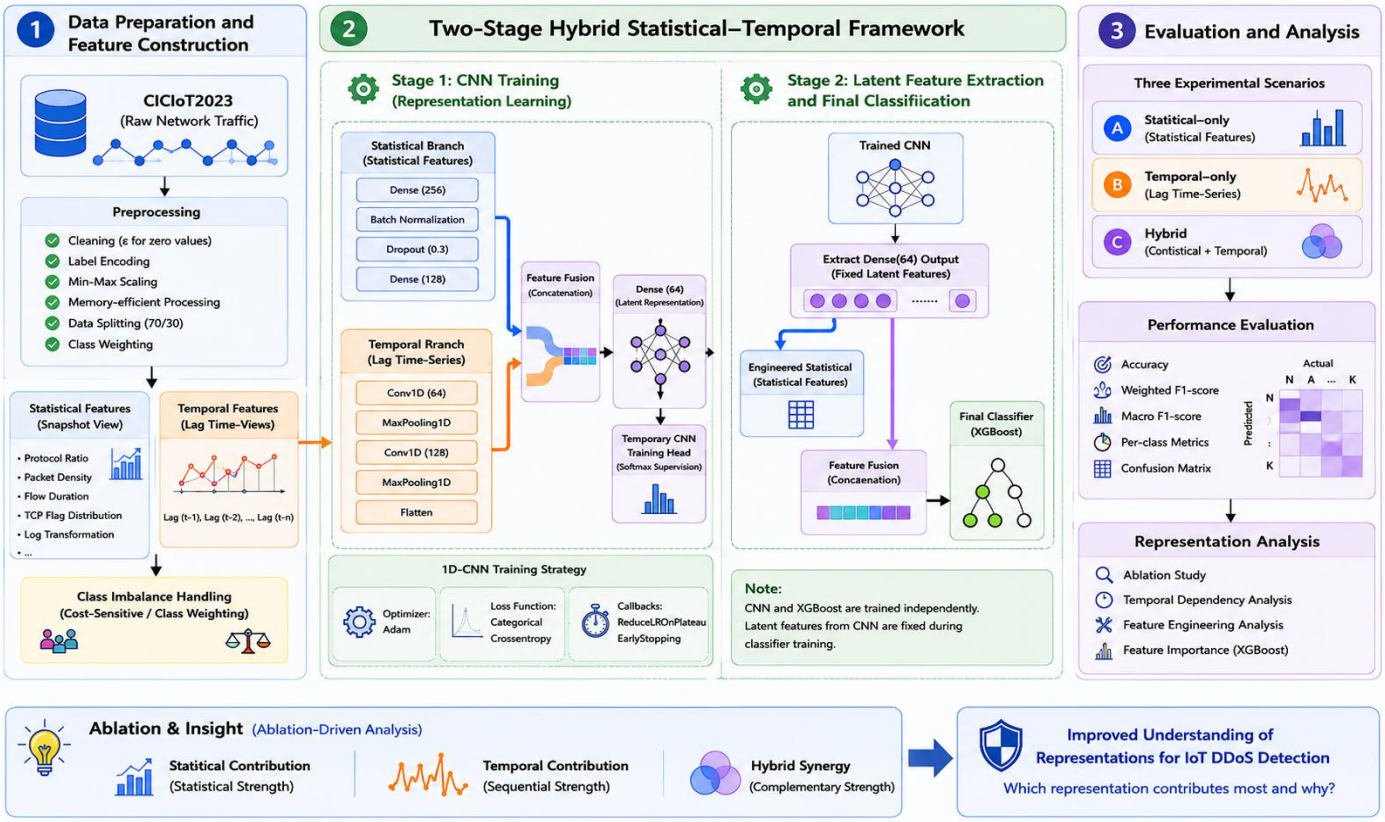

Figure 2. Overall framework of the proposed two-stage hybrid statistical–temporal representation learning and classification framework for IoT DDoS detection.

The first stage focuses on preprocessing and feature construction to generate statistical and temporal traffic representations. Statistical representations are derived from engineered traffic statistics, while temporal representations are constructed using lag-based sequential traffic features to preserve temporal dependencies across traffic intervals. Feature engineering is performed before data sampling and shuffling to maintain temporal consistency during sequence construction.

The second stage employs a two-stage statistical–temporal learning pipeline. A one-dimensional Convolutional Neural Network (1D-CNN) is first used to learn latent temporal representations from lag-based traffic sequences. The learned latent features are subsequently combined with engineered statistical features and used for ensemble tree-based classification using XGBoost. Within this framework, CNN is primarily used for temporal representation learning, whereas XGBoost is used for final classification and representation-oriented analysis.

The final stage focuses on experimental evaluation and ablation-driven analysis across three representation scenarios: statistical-only, temporal-only, and hybrid statistical–temporal representations. In addition to classification performance evaluation, feature importance and ablation analyses are conducted to investigate the relative contribution of different feature representations in IoT DDoS detection.

3.2. Dataset and Preprocessing

This study utilizes the CICIoT2023 dataset [34], which is one of the recent large-scale benchmark datasets designed for cybersecurity research in IoT environments. The dataset contains millions of network traffic samples consisting of benign traffic and multiple DDoS

attack categories with highly imbalanced class distributions. These characteristics make CICIoT2023 suitable for evaluating IDS performance under large-scale and heterogeneous network traffic conditions. To preserve computational efficiency and maintain the validity of temporal representations, this study adopts a feature-engineering-first preprocessing strategy, where feature construction is performed before data sampling and shuffling. This approach ensures that temporal dependencies between network traffic instances remain consistent during temporal feature generation and sequence construction.

3.2.1. Basic Cleaning and Feature Scaling

The preprocessing stage begins with basic data cleaning to handle extreme values and prevent computational instability during feature engineering. Zero values in several attributes, such as Duration and Number, are replaced using a small epsilon constant to avoid division-by-zero errors during ratio-based feature calculations. Categorical labels are subsequently transformed into numerical representations using Label Encoding to enable processing by machine learning and deep learning models. The encoder is fitted on the training data and consistently applied to the testing data to preserve label consistency across all experimental scenarios. After encoding, all numerical features are normalized using Min-Max Scaling within the range of [0,1]. This normalization process reduces feature scale variation and improves training stability, particularly for gradient-based learning models such as CNNs.

3.2.2. Memory-Efficient Data Processing

Considering the large scale of the CICIoT2023 dataset, several memory optimization strategies are implemented to maintain computational stability during data processing. Numerical data types such as float64 and int64 are converted into lower-memory formats such as float32 and int32 without significantly affecting classification precision. In addition, dataset loading is performed incrementally based on individual dataset partitions (*.csv) to avoid excessive memory consumption. Each file partition is processed independently, including during preprocessing and feature engineering, before being merged into the final dataset. This strategy enables large-scale traffic processing under limited computational resources while preserving consistency across all experimental stages.

3.2.3. Data Splitting and Imbalance Handling

The dataset is divided using a file-based splitting strategy with a ratio of 70% training data and 30% testing data to reduce the risk of temporal data leakage between closely related traffic samples. The same split configuration is consistently applied across all experimental scenarios and ablation studies to ensure that performance differences primarily reflect representation effects rather than variations in data distribution. To address the highly imbalanced class distribution in CICIoT2023, this study applies a cost-sensitive learning strategy through class weighting. The weight for each class is calculated using the following equation:

$$w_i = \frac{N}{C \times n_i} \tag{1}$$

where w_i denotes the weight of the i-th class, N represents the total number of training samples, C s the number of classes, and n_i represents the number of samples belonging to the i-th class.

This weighting mechanism assigns larger classification penalties to minority classes during training, enabling the model to become more sensitive to attack categories with limited sample distributions. The class weighting strategy is applied only during the training stage without altering the natural distribution of the testing data.

3.3 Statistical and Temporal Feature Construction

This study constructs two primary feature representation types, namely statistical representations and temporal representations. Both representations are designed to capture network traffic characteristics from different yet complementary perspectives. Statistical representations focus on snapshot-based traffic statistics within a particular interval, whereas temporal representations emphasize traffic evolution over time through sequential dependencies among network traffic instances [17], [35].

3.3.1. Statistical Features

Statistical representations are constructed using network traffic statistics that describe traffic conditions within a specific interval without explicitly modeling temporal order. These features are intended to capture statistical anomalies commonly observed in DDoS attacks, such as packet rate surges, abnormal protocol dominance, and irregular TCP flag distributions [1]. In this study, statistical feature engineering includes several transformations, including protocol ratio features, packet density features, and logarithmic transformations [36]. One example of a protocol ratio representation is defined as follows:

$$Syn\_ratio = \frac{Syn_{count}}{Ack_{count} + \epsilon} \tag{2}$$

where ϵ denotes a small constant used to avoid division-by-zero errors. Network traffic density is further represented using packet rate, defined as:

$$Packet\_rate = \frac{Tot\_size}{Duartion} \tag{3}$$

In addition, logarithmic transformations are applied to several numerical attributes to reduce distribution skewness and improve model training stability:

$$Rate\_log = \log(1 + Rate) \tag{4}$$

The statistical features used in this study are summarized in Table 1.

Table 1. Statistical feature construction used in the proposed framework.

Category	Feature Name	Formula
Protocol Ratio	Syn_ratio	Syn_count / (ack_count + ε)
Protocol Ratio	Fin_ratio	Fin_count / (ack_count + ε)
Protocol Ratio	Rst_ratio	Rst_count / (ack_count + ε)
Packet Density	Packet_rate	Tot_size / Duration
Packet Density	Flag_density	(syn_flag + fin_flag + rst_flag) / (number + ε)
Log Transformation	Rate_log	Log(1 + rate)
Log Transformation	Srate_log	Log(1 + Srate)
Log Transformation	Drate_log	Log(1 + Drate)
Log Transformation	Tot_size_log	Log(1 + tot_size)

These statistical features are used as the primary representation in the statistical-only scenario and as part of the hybrid statistical–temporal representation in the hybrid scenario. This approach enables the model to directly capture changes in network traffic characteristics through statistical traffic snapshots without relying on explicit temporal dependencies.

3.3.2. Temporal Lag Representation

Temporal representations are constructed using a lag-based feature engineering approach to model temporal dependencies in network traffic. This approach is implemented by shifting feature values from previous time intervals into the current traffic instance, allowing the model to learn sequential relationships across traffic observations. In general, the lag-based temporal representation can be expressed as follows:

$$X_t^{lag} = [x_{t-1}, x_{t-2}, x_{t-3}] \tag{5}$$

where X_t^{lag} represents a collection of historical features used to characterize traffic conditions at time t.

As an illustration, feature values from time $t-1$ are incorporated as additional inputs for classification at time t. This strategy enables the model to learn gradual traffic behavior changes, including burst traffic patterns and local temporal dependencies commonly associated with DDoS attacks [37], [38]. The mechanism of temporal feature generation is illustrated in Figure 3.

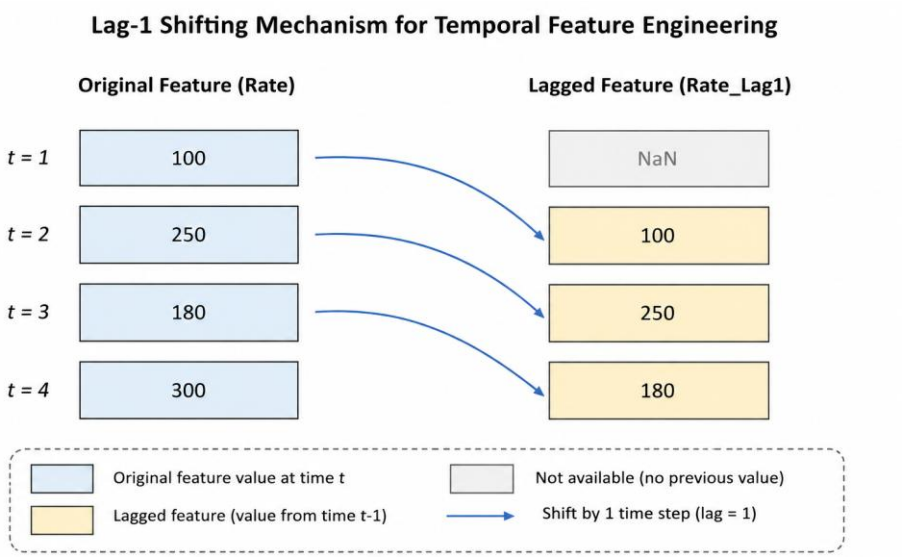

Figure 3. Illustration of temporal feature generation using lag-based shifting representation.

In practice, temporal features are constructed using several key attributes, including packet rate, inter-arrival time (IAT), and packet size, which are shifted across multiple previous intervals. Lag feature construction is performed before data shuffling and sampling to preserve the validity of temporal ordering among traffic instances. This procedure ensures that lag features genuinely represent traffic occurring prior to the current observation, thereby maintaining the historical context of attack behavior. The resulting temporal representations are subsequently used as inputs for the 1D-CNN model to learn sequential patterns and local temporal dependencies automatically [17], [39].

3.4. Hybrid Statistical–Temporal Architecture

This study employs a hybrid statistical–temporal learning pipeline to analyze the contribution of statistical and temporal representations for IoT DDoS detection. Unlike conventional end-to-end deep learning approaches, the framework follows a two-stage learning strategy in which CNN is used for temporal representation learning, while the final classification stage is performed using ensemble tree-based models.

Statistical representations are used to capture snapshot-based traffic characteristics such as packet rate, protocol ratios, flow density, and TCP flag distributions. In contrast, temporal representations are designed to model traffic evolution through sequential dependencies among network traffic observations. The combination of both representations is evaluated to examine whether integrating statistical and temporal information provides complementary benefits compared with using each representation independently. The framework consists of two main stages. The first stage employs a 1D-CNN architecture to learn latent temporal representations from lag-based traffic sequences. The second stage utilizes XGBoost as the final classifier by combining engineered statistical features with latent temporal features extracted from the CNN model.

3.4.1. CNN-Based Latent Representation Learning

Temporal representations in this study are learned using a dual-input 1D-CNN architecture that simultaneously processes lag-based temporal features and engineered statistical features. Unlike two-dimensional CNNs commonly used in computer vision, the 1D-CNN in this study operates on sequential traffic data to capture local traffic patterns and temporal dependencies across consecutive intervals.

The temporal branch receives lag-based sequential traffic features represented as ordered traffic sequences from several previous intervals. These sequential representations are processed through multiple Conv1D and MaxPooling1D layers to extract local temporal patterns and traffic variations. The output from the temporal branch is subsequently transformed into a one-dimensional representation using a Flatten layer. In parallel, the model also receives engineered statistical traffic features through a dense network branch to preserve discriminative statistical traffic characteristics associated with DDoS attacks. The outputs from both

branches are combined using feature concatenation to construct a unified representation. The fused representation is then processed through a Dense layer consisting of 64 neurons to produce a 64-dimensional latent representation. During CNN training, this latent representation is optimized using a softmax classification objective.

To improve training stability and reduce overfitting risk, the CNN training process incorporates ReduceLROnPlateau and EarlyStopping callback strategies. ReduceLROnPlateau adaptively decreases the learning rate when validation performance stagnates, whereas EarlyStopping terminates training when no further validation improvement is observed. These strategies improve training efficiency and help maintain model generalization stability for large-scale and highly imbalanced datasets. After CNN training is completed, the output from the Dense(64) layer is extracted as a fixed latent feature representation. The extracted latent features are subsequently used as additional inputs during ensemble tree-based classification. Therefore, the framework follows a two-stage learning strategy in which CNN is first trained for temporal representation learning, followed by downstream classification using the extracted latent features. The primary configuration of the CNN-based temporal representation learning architecture is summarized in Table 2.

Table 2. Configuration of the proposed CNN-based latent representation learning architecture.

Component	Configuration
Temporal Input	Lag-based sequential traffic features
Temporal Convolution Layer 1	Conv1D (64 filters) + MaxPooling1D
Temporal Convolution Layer 2	Conv1D (128 filters) + MaxPooling1D
Temporal Feature Flattening	Flatten Layer
Statistical Embedding Layers	Dense (256), Dense (128)
Regularization	Batch Normalization, Dropout = 0.3
Feature Fusion	Concatenation of statistical and temporal representations
Latent Representation Layer	Dense (64 neurons)
CNN Training Output	Softmax Layer
Learning Strategy	ReduceLROnPlateau, EarlyStopping
Deep Learning Framework	TensorFlow (Keras API)

3.4.2. Final Classification

In the final classification stage, this study employs XGBoost as the primary classifier due to its capability to handle large-scale data, non-linear feature relationships, and highly imbalanced class distributions. In the hybrid scenario, the classifier input consists of engineered statistical features combined with fixed latent representations extracted from the CNN model. These latent representations are obtained from the Dense(64) layer after CNN training and subsequently used as fixed features during ensemble tree-based classification.

In the statistical-only scenario, the classifier is trained solely using engineered statistical features without incorporating temporal representations. Meanwhile, in the temporal-only scenario, classification is performed using temporal representations obtained through CNN-based sequence learning. This experimental design enables the study to evaluate the relative contribution of statistical, temporal, and hybrid representations for IoT DDoS detection. In addition to XGBoost, Random Forest and CatBoost are also evaluated as comparative ensemble tree-based classifiers to examine performance consistency across different classification models. Nevertheless, the primary focus of this study remains on representation-oriented analysis rather than exploring increasingly complex classification architectures. The main classifier configurations used in this study are summarized in Table 3.

Table 3. Configuration of ensemble tree-based classifiers.

Model	Main Parameters
XGBoost	n_estimators = 100, max_depth = 6, tree_method = hist
CatBoost	iterations = 100, depth = 6
Random Forest	n_estimators = 100, class_weight = balanced

3.5 Experimental Scenarios and Ablation Design

The experimental evaluation in this study is designed to analyze the contribution of statistical and temporal representations to the IoT DDoS detection process. Therefore, the experiments are not solely focused on final classification performance, but also on understanding how different feature representations influence the model's ability to recognize attack patterns under highly imbalanced network traffic conditions. To achieve this objective, the evaluation framework combines representation-based experimental scenarios with ablation-driven analysis to examine the relative contribution of statistical features, temporal dependencies, and feature engineering strategies. The overall experimental scenario design and ablation configuration are illustrated in Figure 4.

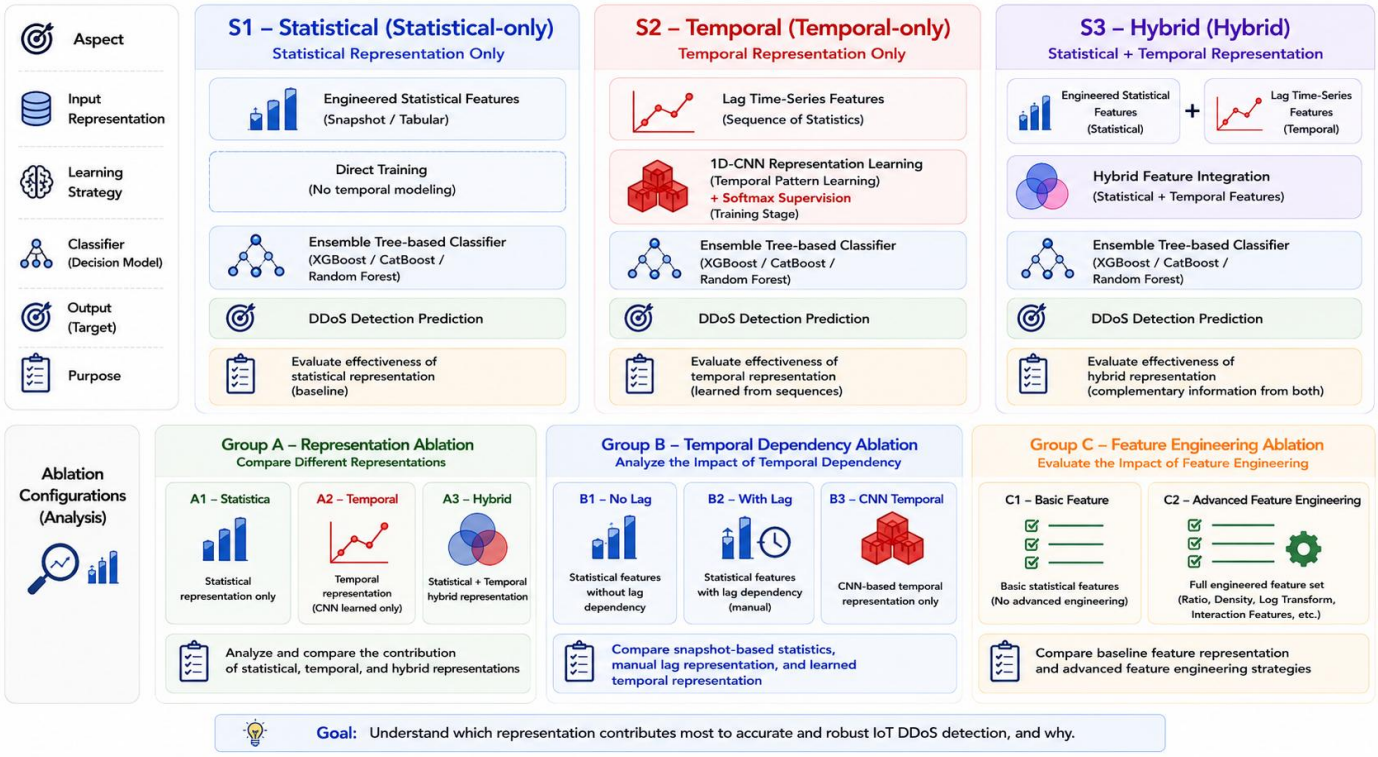

Figure 4. Experimental scenario design and ablation configurations for representation analysis in IoT DDoS detection.

3.5.1. Representation-Based Scenarios

This study evaluates three primary experimental scenarios designed to isolate the contribution of statistical representations, temporal representations, and their hybrid integration during the classification process. The first scenario, namely Scenario A (S1-Statistical), utilizes statistical traffic representations without incorporating explicit temporal dependencies. In this scenario, machine learning classifiers are trained using engineered statistical traffic features to evaluate the effectiveness of snapshot-based traffic representations for distinguishing benign traffic from DDoS attacks. This scenario serves as the baseline representation setting for evaluating the contribution of statistical feature engineering independently.

The second scenario, Scenario B (S2-Temporal), focuses on temporal representations constructed using lag-based traffic sequences and learned through 1D-CNN-based temporal modeling. In this configuration, the model receives only temporal sequence representations without explicitly incorporating engineered statistical traffic features. The objective of this scenario is to evaluate the capability of temporal representations in capturing traffic evolution patterns and local sequential dependencies across network traffic intervals.

The third scenario, Scenario C (S3-Hybrid), combines engineered statistical features with latent temporal representations extracted from the CNN model. In this hybrid configuration, the latent temporal features are concatenated with statistical traffic features before final classification using XGBoost. This scenario is designed to evaluate whether integrating statistical

and temporal representations provides complementary benefits and produces a more comprehensive traffic representation compared with using each representation independently

3.5.2. Ablation Analysis Design

In addition to the three primary experimental scenarios, this study also conducts ablation analysis to examine the contribution of feature representations, temporal dependencies, and feature engineering strategies to classification performance. All ablation experiments are performed using identical preprocessing configurations, dataset splitting strategies, and evaluation protocols to ensure that performance differences primarily reflect the impact of representation configurations rather than variations in data distribution or training procedures.

The ablation study is divided into three main groups. Group A focuses on representation-level analysis by comparing statistical-only, temporal-only, and hybrid representations. Group B investigates the contribution of temporal dependency modeling through lag-based features and CNN-based temporal learning. Meanwhile, Group C evaluates the impact of advanced feature engineering strategies on classification performance. The complete ablation configurations used in this study are summarized in Table 4.

Table 4. Ablation configurations for representation, temporal dependency, and feature engineering analysis.

Ablation Group	Ablation Code	Configuration	Analysis Objective
Representation Ablation	A1 – Statistical	Statistical representation only	Evaluate the effectiveness of snapshot-based statistical representation
	A2 – Temporal	CNN-based latent representation only	Evaluate the effectiveness of learned temporal representation
	A3 – Hybrid	Statistical and temporal hybrid representation	Evaluate the contribution of hybrid representation fusion
Temporal Dependency Ablation	B1 – Statistical-NoLag	Statistical features without lag dependency	Analyze statistical representation without temporal dependency
	B2 – Statistical +Lag	Statistical features with lag features	Analyze the contribution of manual temporal dependency
	B3 – CNN-TemporalOnly	CNN-based temporal representation	Compare manual lag representation and learned temporal representation
Feature Engineering Ablation	C1 – BasicFeature	Basic statistical feature set	Evaluate baseline representation without advanced feature engineering
	C2 – FullFeatureEngineering	Full engineered feature set	Evaluate the contribution of advanced feature engineering strategies

This ablation-driven design enables a more detailed analysis of the relationship between feature representations, temporal dependency modeling, and classification performance compared with conventional benchmark-oriented evaluation alone.

3.6 Evaluation Metrics and Representation Analysis

Performance evaluation in this study is conducted using multiple classification metrics to analyze model behavior under highly imbalanced network traffic conditions. Since the class distribution in the CICIoT2023 dataset is dominated by several majority classes, relying on a single evaluation metric is considered insufficient for representing overall model performance comprehensively. Therefore, the primary evaluation metrics used in this study include accuracy, weighted F1-score, and macro-level evaluation metrics.

Accuracy is used to measure the proportion of correct predictions over the entire testing dataset, while weighted F1-score evaluates classification performance by considering the sample distribution of each class. However, weighted metrics in highly imbalanced datasets may produce seemingly high performance even when the model still struggles to recognize minority attack categories consistently. To address this limitation, macro-level evaluation metrics are also incorporated to assess model robustness across all attack categories more evenly, regardless of class distribution size.

Beyond classification performance evaluation, this study also conducts feature importance analysis to investigate the dominance of different feature representations during the decision-making process. The analysis is performed using feature importance scores obtained from the XGBoost classifier in the hybrid representation scenario. Feature importance analysis is used to evaluate the relative contribution of engineered statistical features and latent temporal representations extracted from the CNN model. This approach enables a more detailed understanding of which feature representations contribute most significantly to IoT DDoS detection under large-scale and highly imbalanced traffic conditions.

4. Results and Representation Analysis

4.1. Experimental Setup and Dataset Characteristics

4.1.1. Dataset Characteristics

This study utilizes the CICIoT2023 dataset developed by the Canadian Institute for Cybersecurity (CIC), which is considered one of the recent benchmark datasets for cybersecurity research in IoT environments. The dataset is publicly available through the CIC research repository at: https://www.unb.ca/cic/datasets/iotdataset-2023.html. The dataset was constructed using more than 100 real IoT devices and contains 33 attack types grouped into several major categories, including Distributed Denial of Service (DDoS), Denial of Service (DoS), Reconnaissance, Spoofing, Web-based attacks, and Mirai botnet traffic. Compared with earlier IDS benchmark datasets, CICIoT2023 provides a substantially larger traffic scale with tens of millions of network traffic samples and highly imbalanced class distributions. These characteristics make the dataset more representative for evaluating modern IDS under large-scale and heterogeneous IoT traffic conditions. To maintain computational efficiency while preserving traffic diversity, this study utilizes a subset consisting of 50 CSV file partitions from the original dataset.

The class distribution in CICIoT2023 is heavily dominated by several DDoS and DoS categories, while other attack classes contain considerably fewer samples. This imbalance introduces a major challenge during classification because learning models may become biased toward majority traffic categories. A representative summary of several dominant classes in CICIoT2023 is presented in Table 5.

Table 5. Representative class distribution of the CICIoT2023 dataset.

Class	Samples	%	Class	Samples	%	Class	Samples	%
DDoS-ICMP-Flood	6.89M	15.31	DDoS-SYN-Flood	3.88M	8.63	Benign	1.05M	2.34
DDoS-UDP-Flood	5.18M	11.51	DDoS-RSTFINFlood	3.87M	8.60	Mirai-GREETH	0.95M	2.11
DDoS-TCP-Flood	4.31M	9.57	DDoS-SynonymousIP	3.45M	7.65	VulnerabilityScan	0.36M	0.79
DDoS-PSHACK-Flood	3.92M	8.71	DoS-UDP-Flood	3.18M	7.06	DNS-Spoofing	0.17M	0.38
DoS-TCP-Flood	2.56M	5.68	DoS-SYN-Flood	1.94M	4.31	Recon-PortScan	0.08M	0.18
Mirai-UDPPlain	0.74M	1.64	Recon-HostDiscovery	0.13M	0.29	DDoS-HTTP-Flood	0.03M	0.06
Mirai-GREIP	0.69M	1.53	MITM-ArpSpoofing	0.11M	0.24	SqlInjection	0.02M	0.04

The distribution shown in Table 5 indicates that most traffic samples are dominated by flooding-based DDoS categories, whereas several attack types such as reconnaissance attacks and HTTP flooding contain substantially smaller proportions. Under such highly imbalanced conditions, evaluation metrics based solely on accuracy may become less representative for measuring model robustness across all attack categories.

To address this imbalance, this study applies a cost-sensitive learning strategy through adaptive class weighting during model training. This approach allows minority classes to receive larger classification penalties without modifying the original distribution of the testing data. The class distribution imbalance and adaptive class weighting strategy used during training are illustrated in Figure 6. This strategy is intended to improve the model's sensitivity toward minority attack categories while preserving the natural traffic distribution during evaluation. These dataset characteristics also motivate the use of weighted F1-score and macro-level evaluation metrics in the subsequent representation analysis.

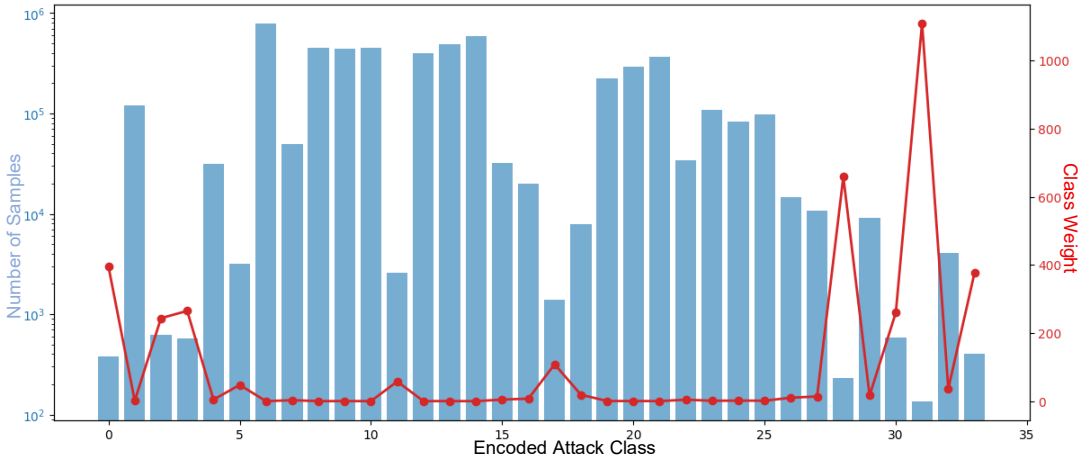

Figure 6. Class distribution imbalance and adaptive class weighting strategy applied during model training.

4.1.2. Experimental Configuration

All experiments are conducted in a local computing environment utilizing a combination of CPU- and GPU-based processing. Deep learning experiments are performed using an NVIDIA GeForce RTX 5070 Ti GPU with 16 GB VRAM to accelerate CNN-based temporal representation learning, whereas ensemble tree-based models such as XGBoost, CatBoost, and Random Forest are executed using CPU-based parallel processing. The complete hardware and software configuration used in this study is summarized in Table 6.

Table 6. Experimental environment and hardware configuration.

Component	Specification
Processor (CPU)	AMD Ryzen 7 9800X3D
Memory (RAM)	64 GB DDR5 6000 MT/s
Storage	NVMe SSD 4 TB
Graphics Processing Unit (GPU)	NVIDIA GeForce RTX 5070 Ti (16 GB VRAM)
Operating System	Windows 11 64-bit
Programming Language	Python
Data Processing Libraries	Pandas, NumPy
Machine Learning Libraries	Scikit-learn, XGBoost, CatBoost
Deep Learning Framework	TensorFlow (Keras API)

To maintain the validity of the ablation study and representation-oriented analysis, all experimental scenarios use identical preprocessing procedures, dataset splitting configurations, and evaluation protocols. Consequently, performance differences across scenarios can be interpreted primarily as the effect of representation configurations and model settings rather than variations in data distribution or training procedures.

4.2. Performance Across Representation Scenarios

4.2.1. Statistical Representation Performance

The experimental results indicate that statistical traffic representations provide very strong detection capability for IoT DDoS attacks on the CICIoT2023 dataset. Under the statistical-only scenario (S1-Statistical), XGBoost achieved the highest overall performance with an accuracy of 99.36% and a weighted F1-score of 99.31%, outperforming all other experimental configurations. The complete performance comparison across representation scenarios and classifiers is summarized in Table 7.

One of the most notable findings is that statistical representations alone already produce near-saturated classification performance across multiple classifiers. This result suggests that the traffic statistics available in CICIoT2023 contain highly discriminative patterns capable of separating benign and malicious traffic without requiring complex temporal modeling.

Features such as TCP flag distributions, packet density, protocol ratios, and packet rate appear sufficient to capture dominant traffic irregularities associated with large-scale DDoS attacks. Similar observations have also been reported in recent IDS studies, where engineered statistical traffic features and flow-based representations demonstrated strong effectiveness for detecting volumetric DDoS behavior and abnormal traffic surges [36], [40]. These findings indicate that engineered statistical traffic features remain highly effective for modern IDS systems, particularly for flood-based attack scenarios where abnormal traffic behavior can already be observed clearly at the snapshot level.

Table 7. Performance comparison across statistical, temporal, and hybrid representation scenarios.

Scenario	Model	Accuracy	Precision	Recall	Weighted F1
S1- Statistical	CatBoost	0.97874	0.97805	0.97874	0.97838
S1- Statistical	Random Forest	0.97656	0.98670	0.97656	0.98055
S1- Statistical	XGBoost	0.99357	0.99315	0.99357	0.99312
S2-Temporal	CatBoost	0.81505	0.81452	0.81505	0.81475
S2-Temporal	Random Forest	0.86564	0.88523	0.86564	0.86746
S2-Temporal	XGBoost	0.92905	0.92936	0.92905	0.92716
S3-Hybrid	CatBoost	0.97982	0.97958	0.97982	0.97970
S3-Hybrid	Random Forest	0.95820	0.97143	0.95820	0.96197
S3-Hybrid	XGBoost	0.99358	0.99316	0.99358	0.99313

4.2.2. CNN-Based Latent Representation Performance

The temporal representation scenario (S2-Temporal) produced substantially lower performance compared with the statistical representation scenario across all classifiers. Although the CNN-based temporal representation was able to learn sequential traffic patterns and local temporal dependencies, its discriminative capability remained below that of engineered statistical traffic features. Under the temporal-only configuration, XGBoost achieved the best performance with an accuracy of 92.90% and a weighted F1-score of 92.72%, while CatBoost and Random Forest experienced larger performance degradation.

These findings suggest that temporal sequence representations alone may not sufficiently capture the full variability of IoT DDoS traffic patterns under highly imbalanced traffic conditions. In large-scale DDoS traffic, many attack characteristics still appear to be strongly associated with observable statistical anomalies such as packet bursts, abnormal protocol distributions, and traffic density irregularities, which can already be effectively represented through snapshot-based traffic statistics [13], [36]. Consequently, the latent representations learned by CNN appear to provide complementary rather than dominant discriminative contributions during classification.

Another important observation is the noticeable precision–recall gap in several temporal configurations, particularly for Random Forest. In the temporal-only scenario, Random Forest achieved 88.52% precision but only 86.56% recall, indicating that the model tends to produce more conservative predictions under highly imbalanced conditions. While this behavior helps reduce false positive predictions, it also suggests that certain attack instances, especially minority traffic categories, remain difficult to detect consistently using temporal representations alone. Similar challenges related to minority attack detection under imbalanced traffic distributions have also been discussed in recent IDS imbalance studies [13].

4.2.3. Hybrid Representation Performance

The hybrid statistical–temporal scenario (S3-Hybrid) achieved the highest overall performance, particularly when combined with XGBoost. However, the numerical improvement over the statistical-only scenario remained relatively small. For example, the weighted F1-score of XGBoost increased only marginally from 99.312% to 99.313% after integrating latent temporal representations. This finding suggests that the primary source of discriminative information still originates from engineered statistical traffic features, while temporal representations contribute mainly as complementary sequential information.

Despite the relatively small numerical improvement, the hybrid configuration demonstrated more stable performance across multiple classifiers and evaluation metrics. The integration of latent temporal representations appears to provide additional contextual

information regarding traffic evolution patterns that may not be fully captured through isolated statistical snapshots. Similar hybrid IDS studies have also reported that temporal deep learning representations can complement statistical traffic features by capturing sequential dependencies and traffic evolution behavior [10], [13].

Differences in classifier behavior also reveal important relationships between feature characteristics and classification mechanisms. XGBoost consistently achieved the most stable performance across all scenarios with relatively balanced precision and recall values, indicating stronger robustness under highly imbalanced traffic distributions. This behavior is likely associated with the boosting mechanism of XGBoost, which incrementally focuses on difficult samples and complex feature interactions. In contrast, Random Forest exhibited larger precision–recall gaps, particularly in temporal and hybrid scenarios, suggesting lower sensitivity toward certain minority attack categories. Meanwhile, CatBoost produced stable but consistently lower performance, likely because the representations used in this study are dominated by engineered numerical traffic features and latent continuous representations rather than categorical feature relationships.

To further analyze classification behavior, confusion matrix evaluation was conducted using the best-performing model, namely the S3-Hybrid configuration with XGBoost. The confusion matrix shows a strong concentration along the main diagonal, indicating that most traffic samples across attack categories were classified correctly. Misclassification patterns remained relatively limited and did not exhibit dominant confusion between specific attack categories, suggesting good generalization capability under large-scale and highly imbalanced IoT traffic conditions.

4.3. Ablation and Representation Contribution Analysis

To better understand the contribution of feature representations and temporal dependencies to classification performance, this study conducts a series of ablation experiments across multiple representation and feature engineering configurations. Unlike conventional benchmark-oriented evaluation that primarily emphasizes final accuracy, the ablation framework in this study is designed to analyze the relationship between statistical representations, temporal representations, and classification robustness under highly imbalanced IoT traffic conditions. The complete ablation results are presented in Table 8.

Table 8. Ablation study results across different feature representation configurations.

Ablation Scenario	Accuracy	Weighted F1	Macro Precision	Macro Recall
A1 – Statistical	0.99359	0.99314	0.85374	0.74875
A2 – Temporal	0.92905	0.92716	0.76341	0.65442
A3 – Hybrid	0.99358	0.99313	0.84975	0.74552
B1 – Statistical-NoLag	0.99359	0.99314	0.85374	0.74875
B2 – Statistical+Lag	0.99353	0.99305	0.85021	0.74381
B3 – CNN-TemporalOnly	0.92905	0.92716	0.76341	0.65442
C1 – BasicFeature	0.99299	0.99239	0.84188	0.73646
C2 – FullFeatureEngineering	0.99360	0.99314	0.85171	0.74808

4.3.1. Representation Contribution Analysis

The ablation results suggest that statistical traffic representations provide the strongest contribution to IoT DDoS detection performance on CICIoT2023. The A1-Statistical scenario achieved a weighted F1-score of 99.31%, which is nearly identical to the hybrid configuration. This finding suggests that engineered statistical traffic features are already highly discriminative for separating benign and malicious traffic in large-scale IoT environments [36], [40]. In particular, snapshot-based traffic characteristics such as packet density, protocol imbalance, and TCP flag distributions appear sufficient to capture the dominant behavioral anomalies associated with flooding-based attacks.

In contrast, the A2-Temporal and B3-CNN-TemporalOnly configurations experienced a noticeable performance decline compared with the statistical representation scenario. The temporal representation achieved a weighted F1-score of approximately 92.7%, while macro recall decreased to 65.4%, indicating lower robustness across minority attack categories.

These results suggest that temporal sequence representations alone are not sufficient to capture the full variability of IoT DDoS traffic patterns when statistical traffic information is excluded. An interesting observation is that the performance gap between statistical and temporal representations remains substantially larger than the improvement obtained from hybrid fusion. This behavior implies that the discriminative capability of the dataset appears to rely more strongly on observable traffic statistics than on long-range temporal dependencies.

Nevertheless, temporal representations still provide complementary benefits within the hybrid configuration. Although the numerical improvement in weighted performance is relatively small, the hybrid representation demonstrates slightly more stable macro-level performance under multiclass and highly imbalanced traffic conditions. This finding indicates that CNN-based temporal learning functions more as a representation enhancer that supplements statistical traffic information rather than replacing it as the primary discriminative component.

The noticeable gap between weighted performance and macro-level evaluation metrics further indicates that extremely high weighted scores may still mask lower sensitivity toward minority attack categories [18], [19]. Although the statistical and hybrid scenarios achieved near-saturated weighted F1-score performance, the macro recall values remained substantially lower, suggesting that several minority attack classes were still more difficult to recognize consistently. This behavior highlights the importance of representation-level and class-level evaluation in highly imbalanced IDS benchmarks, where overall benchmark accuracy alone may not fully reflect model robustness across heterogeneous attack distributions.

4.3.2. Temporal Dependency Analysis

The temporal dependency ablation further shows that short-term lag dependencies have relatively limited influence on the final classification performance. The comparison between B1-Statistical-NoLag and B2-Statistical+Lag reveals only marginal differences in both weighted F1-score and macro-level evaluation metrics. This result suggests that the snapshot-based traffic statistics available in CICIoT2023 are already sufficiently informative for representing DDoS attack behavior without requiring complex temporal dependency modeling.

These findings imply that many attack patterns in the dataset appear to exhibit characteristics more consistent with instantaneous traffic anomalies than long sequential behavioral patterns. In other words, abnormal traffic intensity and protocol behavior can already be identified effectively from isolated traffic snapshots. This observation suggests that increasingly complex temporal modeling may not always produce substantial performance improvements, particularly when statistical traffic anomalies are already highly discriminative [10], [13]. In the context of CICIoT2023, the additional computational cost introduced by temporal sequence modeling appears to provide only limited practical gains relative to the strength of statistical traffic representations.

4.3.3. Feature Engineering Contribution Analysis

The comparison between C1-BasicFeature and C2-FullFeatureEngineering demonstrates that feature engineering still contributes meaningfully to classification performance, even when using powerful ensemble classifiers such as XGBoost. The inclusion of ratio-based features, traffic density features, and logarithmic transformations consistently improved both weighted F1-score and macro-level metrics. These transformations help stabilize traffic representations under highly skewed and heterogeneous traffic distributions, enabling the classifier to separate attack categories more effectively [36].

An important insight from this analysis is that high IDS performance is not solely determined by the complexity of deep learning architectures. Instead, the quality of traffic representation and the effectiveness of feature engineering remain critical factors in modern IoT intrusion detection systems. The ablation results further suggest that the strong detection performance achieved in this study originates primarily from the discriminative capability of engineered statistical traffic representations, while temporal modeling and advanced feature engineering mainly contribute by enhancing representation robustness and classification stability under highly imbalanced traffic conditions.

4.4. Feature Importance and Representation Dominance Analysis

In addition to classification performance evaluation, this study also conducts feature importance analysis to examine the dominance of different feature representations during the model decision-making process. The analysis is performed using feature importance scores obtained from the XGBoost classifier in the hybrid representation scenario to evaluate the

relative contribution of engineered statistical traffic features and CNN-based latent temporal representations. The visualization in Figure 7 shows that the classification process is still strongly dominated by statistical traffic features. The two most influential features, namely psh_flag_number and flag_density, exhibit substantially higher importance scores than the remaining features. The dominance of these features indicates that the model relies heavily on TCP flag distribution patterns and traffic intensity characteristics for detecting DDoS attacks. This observation is consistent with the flooding-oriented behavior of CICIoT2023, where attacks generate abnormal connection bursts, increased flag activity, and extremely high packet density within short traffic intervals.

Another important observation is the highly uneven contribution distribution across features. Most importance scores are concentrated on a small number of dominant statistical indicators, while many other features contribute only marginally to the final decision process. This finding suggests that the classification behavior on CICIoT2023 is driven primarily by a limited set of highly discriminative traffic characteristics rather than by uniform utilization of all available features. In other words, the model appears to construct its decision boundaries mainly around several dominant statistical traffic indicators.

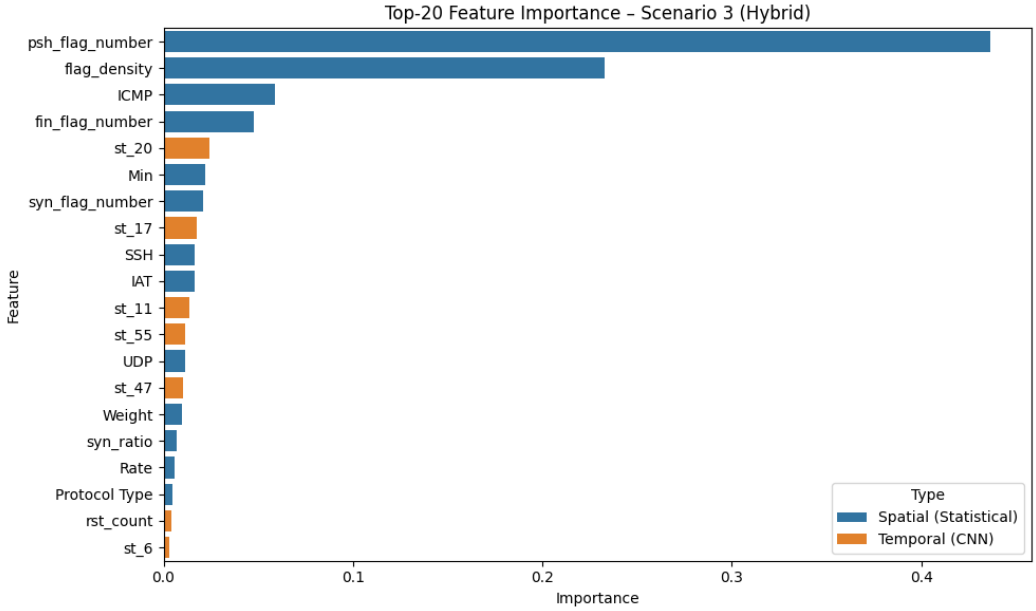

Figure 7. Feature importance distribution between statistical features and CNN-based temporal representations in the hybrid model.

CNN-based temporal features are still present within the Top-20 feature importance rankings through several latent representations such as *st_20*, *st_17*, *st_11*, and *st_55*. However, their contribution remains noticeably lower than that of the dominant statistical traffic features. This result indicates that temporal representations are still capable of capturing additional relevant traffic patterns, but their role is primarily complementary rather than dominant in the overall detection process [41].

An interesting implication of these findings is that the superior performance achieved by the hybrid configuration does not originate from temporal representation dominance, but rather from the ability of temporal representations to refine classification stability for certain traffic patterns that may not be fully represented through isolated statistical snapshots. This behavior aligns closely with the previous ablation results, where the statistical-only configuration already achieved weighted F1-score performance nearly identical to the hybrid configuration.

More importantly, these results highlight that very high benchmark performance on CICIoT2023 does not necessarily imply balanced contribution across all model components. Although the hybrid framework combines CNN-based temporal learning with XGBoost classification, the overall decision-making process remains substantially more influenced by engineered statistical traffic representations than by latent temporal representations. Consequently, the findings suggest that increasing deep learning complexity does not always

correspond to proportional representation dominance in modern IDS systems. Instead, the effectiveness of statistical traffic representation and feature engineering remains the primary factor driving classification performance in large-scale IoT DDoS detection.

5. Conclusions

This study presented an ablation-driven analytical framework to investigate the contribution of statistical, temporal, and hybrid representations for large-scale IoT DDoS detection using the CICIoT2023 dataset. Unlike conventional intrusion detection studies that primarily emphasize benchmark performance improvement, this work focused on understanding how different feature representations contribute to the detection process under highly imbalanced traffic conditions. The proposed framework combined CNN-based temporal representation learning with ensemble tree-based classification using XGBoost to evaluate the relative effectiveness of different representation configurations.

Experimental results consistently showed that engineered statistical traffic representations remain the dominant source of discriminative information for IoT DDoS detection. The statistical representation scenario achieved the highest overall performance with 99.36% accuracy and 99.31% weighted F1-score, while feature importance analysis further confirmed the strong dominance of statistical traffic features in the classification process. In contrast, CNN-based temporal representations provided comparatively limited contributions and primarily acted as complementary representations that improved classification stability under multiclass and highly imbalanced traffic conditions.

The ablation analysis further demonstrated that advanced statistical feature engineering contributed more substantially than temporal dependency modeling in the evaluated experimental settings. Although temporal sequence learning was able to capture additional sequential traffic behavior, the hybrid configuration produced only marginal improvements over the statistical representation alone. These findings suggest that, for CICIoT2023, snapshot-based traffic characteristics remain substantially more informative than learned temporal dependencies for distinguishing benign and malicious traffic. Another important finding is that very high weighted benchmark performance does not necessarily imply balanced robustness across all attack categories. The comparison between weighted and macro-level evaluation metrics showed that representation-level analysis remains important for understanding classifier behavior under highly imbalanced traffic distributions, particularly for minority attack classes.

Overall, this study provides empirical evidence that increasing deep learning complexity does not always lead to proportional representational gains in IoT intrusion detection systems. Instead, the effectiveness of engineered statistical representations remains a critical factor in achieving robust IDS performance. These findings contribute to a deeper understanding of representation behavior in modern IoT IDS research and provide practical insights for future representation-aware intrusion detection design. Nevertheless, this study has several limitations. The experiments were conducted only on the CICIoT2023 dataset, and the temporal modeling approach relied on lag-based sequence construction using a relatively lightweight 1D-CNN architecture. Future work may explore cross-dataset representation robustness, adaptive representation learning under evolving attack patterns, and lightweight representation-aware IDS frameworks for real-time IoT deployment.

Author Contributions: Conceptualization: D.N.W. and D.R.I.M.S.; Methodology: D.N.W. and D.R.I.M.S.; Software: D.N.W.; Validation: Aj.S., M.A.M. and Ac.S.; Formal analysis: D.R.I.M.S.; Investigation: Aj.S.,; Resources: D.N.W.; Data curation: D.N.W.; Writing—original draft preparation: D.N.W.; Writing—review and editing: D.R.I.M.S., Aj.S., M.A.M. and Ac.S.; Visualization: D.N.W. and D.R.I.M.S.; Supervision: D.R.I.M.S.; Project administration: D.N.W.; Funding acquisition: All. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors would like to acknowledge the use of artificial intelligence (AI)-assisted tools during the preparation of this manuscript. AI tools were utilized to support language refinement, grammatical improvement, and the development of conceptual framework illustrations and schematic figures. All technical content, experimental design, analysis,

interpretation of results, and final manuscript validation were conducted and reviewed entirely by the authors.

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

References

[1] A. Ahmim, F. Maazouzi, M. Ahmim, S. Namane, and I. Ben Dhaou, "Distributed Denial of Service Attack Detection for the Internet of Things Using Hybrid Deep Learning Model," *IEEE Access*, vol. 11, pp. 119862–119875, Aug. 2023, doi: 10.1109/ACCESS.2023.3327620.

[2] S. A. Wahab, S. Sultana, N. Tariq, M. Mujahid, J. A. Khan, and A. Mylonas, "A Multi-Class Intrusion Detection System for DDoS Attacks in IoT Networks Using Deep Learning and Transformers," *Sensors*, vol. 25, no. 15, p. 4845, Aug. 2025, doi: 10.3390/s25154845.

[3] S. Zubair, H. Abdulazeez, B. A. Salihu, M. Umar, and P. I. Ojo-Arome, "An Edge-Enabled Multimodal Cyber-Physical System for Near-Real-Time Intrusion Detection in Fiber-Optic Networks," *J. Futur. Artif. Intell. Technol.*, vol. 3, no. 1, pp. 84–98, May 2026, doi: 10.62411/faith.3048-3719-363.

[4] S. H. Rafique, A. Abdallah, N. S. Musa, and T. Murugan, "Machine Learning and Deep Learning Techniques for Internet of Things Network Anomaly Detection—Current Research Trends," *Sensors*, vol. 24, no. 6, p. 1968, Mar. 2024, doi: 10.3390/s24061968.

[5] M. Alharby, "Evaluating machine learning approaches for multiple attack classification with improved computational efficiency in IoT networks," *Sci. Rep.*, vol. 15, no. 1, p. 39914, Nov. 2025, doi: 10.1038/s41598-025-23711-7.

[6] E. Altulaihan, M. A. Almaiah, and A. Aljughaiman, "Anomaly Detection IDS for Detecting DoS Attacks in IoT Networks Based on Machine Learning Algorithms," *Sensors*, vol. 24, no. 2, p. 713, Jan. 2024, doi: 10.3390/s24020713.

[7] Z. S. Dhahir, "A Hybrid Approach for Efficient DDoS Detection in Network Traffic Using CBLOF-Based Feature Engineering and XGBoost," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 2, pp. 174–190, Sep. 2024, doi: 10.62411/faith.2024-33.

[8] W. Sarasjati, S. Rustad, Purwanto, H. A. Santoso, and D. R. I. M. Setiadi, "Phishing Detection Using Random Forest-Based Weighted Bootstrap Sampling and LASSO+ Feature Selection," *Int. J. Saf. Secur. Eng.*, vol. 14, no. 6, pp. 1783–1794, Dec. 2024, doi: 10.18280/ijsse.140613.

[9] O. D. Okey, D. Z. Rodriguez, and J. H. Kleinschmidt, "Enhancing IoT Intrusion Detection with Federated Learning-Based CNN-GRU and LSTM-GRU Ensembles," in *2024 19th International Symposium on Wireless Communication Systems (ISWCS)*, Jul. 2024, pp. 1–6. doi: 10.1109/ISWCS61526.2024.10639159.

[10] C. Zhang, J. Li, N. Wang, and D. Zhang, "Research on Intrusion Detection Method Based on Transformer and CNN-BiLSTM in Internet of Things," *Sensors*, vol. 25, no. 9, p. 2725, Apr. 2025, doi: 10.3390/s25092725.

[11] F. Kabura and T. Nsabimana, "An Attention-Enhanced CNN–RBF Framework for Network Intrusion Detection in Imbalanced Traffic," *J. Comput. Theor. Appl.*, vol. 3, no. 3, pp. 349–368, Jan. 2026, doi: 10.62411/jcta.15419.

[12] D. R. I. M. Setiadi, S. Widiono, A. N. Safriandono, and S. Budi, "Phishing Website Detection Using Bidirectional Gated Recurrent Unit Model and Feature Selection," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 2, pp. 75–83, Jul. 2024, doi: 10.62411/faith.2024-15.

[13] M. Sajid *et al.*, "Enhancing intrusion detection: a hybrid machine and deep learning approach," *J. Cloud Comput.*, vol. 13, no. 1, p. 123, Jul. 2024, doi: 10.1186/s13677-024-00685-x.

[14] S. Neupane *et al.*, "Explainable Intrusion Detection Systems (X-IDS): A Survey of Current Methods, Challenges, and Opportunities," *IEEE Access*, vol. 10, pp. 112392–112415, 2022, doi: 10.1109/ACCESS.2022.3216617.

[15] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique," *Cybersecurity*, vol. 5, no. 1, p. 1, Dec. 2022, doi: 10.1186/s42400-021-00103-8.

[16] Z. Wang, H. Huang, R. Du, X. Li, and G. Yuan, "IoT Intrusion Detection Model based on CNN-GRU," *Front. Comput. Intell. Syst.*, vol. 4, no. 2, pp. 90–95, Jun. 2023, doi: 10.54097/fcis.v4i2.10302.

[17] P. Sinha, D. Sahu, S. Prakash, T. Yang, R. S. Rathore, and V. K. Pandey, "A high performance hybrid LSTM CNN secure architecture for IoT environments using deep learning," *Sci. Rep.*, vol. 15, no. 1, p. 9684, Mar. 2025, doi: 10.1038/s41598-025-94500-5.

[18] V. Shanmugam, R. Razavi-Far, and E. Hallaji, "Addressing Class Imbalance in Intrusion Detection: A Comprehensive Evaluation of Machine Learning Approaches," *Electronics*, vol. 14, no. 1, p. 69, Dec. 2024, doi: 10.3390/electronics14010069.

[19] S. Farhadpour, T. A. Warner, and A. E. Maxwell, "Selecting and Interpreting Multiclass Loss and Accuracy Assessment Metrics for Classifications with Class Imbalance: Guidance and Best Practices," *Remote Sens.*, vol. 16, no. 3, p. 533, Jan. 2024, doi: 10.3390/rs16030533.

[20] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 2, pp. 39–53, Apr. 2004, doi: 10.1145/997150.997156.

[21] A. Hussain, J. Heidemann, and C. Papadopoulos, "A framework for classifying denial of service attacks," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, Aug. 2003, pp. 99–110. doi: 10.1145/863955.863968.

[22] S. T. Zargar, J. Joshi, and D. Tipper, "A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks," *IEEE Commun. Surv. Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013, doi: 10.1109/SURV.2013.031413.00127.

[23] A. Çetin and S. Öztürk, "Comprehensive Exploration of Ensemble Machine Learning Techniques for IoT Cybersecurity Across Multi-Class and Binary Classification Tasks," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 371–384, Feb. 2025, doi: 10.62411/faith.3048-3719-51.

[24] J. P. Ntayagabiri, Y. Bentaleb, J. Ndikumagenge, and H. El Makhtoum, "OMIC: A Bagging-Based Ensemble Learning Framework for Large-Scale IoT Intrusion Detection," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 401–416, Feb. 2025, doi: 10.62411/faith.3048-3719-63.

[25] J. P. Ntayagabiri, Y. Bentaleb, J. Ndikumagenge, and H. El Makhtoum, "A Comparative Analysis of Supervised Machine Learning Algorithms for IoT Attack Detection and Classification," *J. Comput. Theor. Appl.*, vol. 2, no. 3, pp. 395–409, Feb. 2025, doi: 10.62411/jcta.11901.

[26] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[27] Piyush M. Prajapati, Dr. Priyesh P. Gandhi, and Dr. Sheshang Degadwala, "Deep Learning-Based Classification of IoT DDoS Attacks Using CNN-LSTM," *Int. J. Sci. Res. Sci. Technol.*, vol. 12, no. 5, pp. 389–397, Oct. 2025, doi: 10.32628/IJSRST25125248.

[28] N. U. Ain, M. Sardaraz, M. Tahir, M. W. Abo Elsoud, and A. Alourani, "Securing IoT Networks Against DDoS Attacks: A Hybrid Deep Learning Approach," *Sensors*, vol. 25, no. 5, p. 1346, Feb. 2025, doi: 10.3390/s25051346.

[29] S. Abbas *et al.*, "Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks," *PeerJ Comput. Sci.*, vol. 10, p. e1793, Jan. 2024, doi: 10.7717/peerj-cs.1793.

[30] R. Jablaoui, O. Cheikhrouhou, M. Hamdi, and N. Liouane, "Deep learning enabled intrusion detection system for IoT security," *EURASIP J. Wirel. Commun. Netw.*, vol. 2025, no. 1, p. 66, Aug. 2025, doi: 10.1186/s13638-025-02477-6.

[31] O. A. Hussain, Z. Chen, and H. Zhu, "sSecure Net: A Hybrid CNN-LSTM-based Intrusion Detection System for Securing IoT Networks," in *Proceedings of the 4th International Conference on Computer, Artificial Intelligence and Control Engineering*, Jan. 2025, pp. 537–544. doi: 10.1145/3727648.3727736.

[32] C. Asuai *et al.*, "Enhancing DDoS Detection via 3ConFA Feature Fusion and 1D Convolutional Neural Networks," *J. Futur. Artif. Intell. Technol.*, vol. 2, no. 1, pp. 145–162, Jun. 2025, doi: 10.62411/faith.3048-3719-105.

[33] D. M. A. A. Afraji, J. Lloret, and L. Peñalver, "An Integrated Hybrid Deep Learning Framework for Intrusion Detection in IoT and IIoT Networks Using CNN-LSTM-GRU Architecture," *Computation*, vol. 13, no. 9, p. 222, Sep. 2025, doi: 10.3390/computation13090222.

[34] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, p. 5941, Jun. 2023, doi: 10.3390/s23135941.

[35] M. Luay *et al.*, "Time Matters: Temporal NetFlow Features for ML-Based Network Intrusion Detection," *IEEE Access*, vol. 14, pp. 66899–66913, 2026, doi: 10.1109/ACCESS.2026.3688204.

[36] Z. Liu, Y. Wang, F. Feng, Y. Liu, Z. Li, and Y. Shan, "A DDoS Detection Method Based on Feature Engineering and Machine Learning in Software-Defined Networks," *Sensors*, vol. 23, no. 13, p. 6176, Jul. 2023, doi: 10.3390/s23136176.

[37] W. Dai, X. Li, W. Ji, and S. He, "Network Intrusion Detection Method Based on CNN-BiLSTM-Attention Model," *IEEE Access*, vol. 12, pp. 53099–53111, 2024, doi: 10.1109/ACCESS.2024.3384528.

[38] J. Zhao, Y. Liu, Q. Zhang, and X. Zheng, "CNN-AttBiLSTM Mechanism: A DDoS Attack Detection Method Based on Attention Mechanism and CNN-BiLSTM," *IEEE Access*, vol. 11, pp. 136308–136317, 2023, doi: 10.1109/ACCESS.2023.3334916.

[39] A. M. Alashjaee, "Deep learning for network security: an Attention-CNN-LSTM model for accurate intrusion detection," *Sci. Rep.*, vol. 15, no. 1, p. 21856, Jul. 2025, doi: 10.1038/s41598-025-07706-y.

[40] O. D. Okey *et al.*, "Correction: Okey et al. BoostedEnML: Efficient Technique for Detecting Cyberattacks in IoT Systems Using Boosted Ensemble Machine Learning. Sensors 2022, 22, 7409," *Sensors*, vol. 25, no. 19, p. 6125, Oct. 2025, doi: 10.3390/s25196125.

[41] A. I. Sourav, M. S. Islam, U. S. Nahar, M. I. Nayon, and M. T. Ahmed, "Hybrid Framework with Feature Selection and Explainable AI for IoT Intrusion Detection," in *2025 IEEE 4th International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things (RAAICON)*, Nov. 2025, pp. 571–575. doi: 10.1109/RAAICON69033.2025.11502208.