Lightweight Flow-based Anomaly Detection for IoT Using HC-MTDNN: A Hierarchically Cascaded Multitask Deep Neural Network

Mohamed Amine Beghoura and Younes Belouche

University of Mohamed El Bachir El Ibrahimi, Bordj Bou Arreridj, Algeria

https://doi.org/10.26636/jtit.2025.4.2311

Abstract — In this article, we propose a lightweight, hierarchical multi-task learning framework designed for detecting both high-level and fine-grained threats in IoT traffic. The developed model focuses on anomalies detectable through flow-level metadata. The deliberate choice to prioritize computational efficiency by excluding content analysis scopes the approach to payload-independent threats, while still enabling robust detection of key attack classes. To further enhance efficiency within this metadata-driven paradigm, we introduce HC-MTDNN, a hierarchical multitask model that integrates a gated feature mechanism and feature reuse to significantly reduce redundancy and computational overhead, improving upon previous hierarchical architectures and achieving high performance while dealing with volumetric and protocol-based attacks. The model is evaluated on four benchmark datasets: CICIoT2023, N-BaIoT, Bot-IoT, and Edge-IIoTset. It demonstrates strong performance in both binary and multiclass classification tasks, with an average inference time of 122 µs per sample and a compact model size of 2.4 MB. The proposed framework effectively balances accuracy and computational efficiency, offering a practical and scalable solution for securing resource-constrained IoT environments.

Keywords — anomaly detection, deep neural network, IoT security, lightweight model, multitask learning, network traffic analysis

1. Introduction

The exponential growth of the Internet of Things (IoT) has transformed conventional networks into vast interconnected ecosystems spanning various domains such as smart homes, healthcare, industrial automation, and smart cities [1]. By 2025, more than 75 billion IoT devices are projected to be online, roughly 40% of them in smart home environments [2]. Although this expansion offers convenience and functionality, it also increases vulnerability to attacks. Many IoT devices operate under severe resource limitations and lack robust security mechanisms, making them attractive targets for cyber threats [3], [4]. In particular, high-profile attacks such as the Mirai botnet [5] have demonstrated the dangers associated with compromised IoT infrastructures.

IoT anomaly detection is particularly sophisticated due to the heterogeneity, volume, and nonstationarity of the data generated from such applications. Traditional detection methods, such as rule-based systems and fixed thresholds, cannot cope with the dynamic and multimodal characteristic of traffic in IoT networks [6]. As a result, these methods frequently misclassify benign traffic as malicious, leading to an excessive number of false alarms.

However, most of the existing approaches in IoT anomaly detection handle each classification task independently, employing separate single-task models for binary classification, attack-type categorization, and fine-grained subtype classification. The authors of [7], [8] each deploy distinct single-task models dedicated specifically to binary or multiclass scenarios without exploiting task interdependencies. Such isolated training can neglect beneficial shared representations across tasks, potentially limiting generalization and computational efficiency.

To address these challenges, we propose HC-MTDNN, a hierarchical multitask model designed to tackle three core IoT anomaly detection tasks in a single forward pass. Instead of treating each task independently, the model cascades predictions, refining the output at each level. HC-MTDNN performs anomaly detection in a staged, multitask manner, progressing from binary to coarse to fine classification. Each level of the network is responsible for different interdependent tasks, such as feature extraction, anomaly detection, and classification. By cascading these tasks, the model can progressively refine its analysis, leading to a more accurate identification of anomalies. This structure allows the model to efficiently process data and reduce false positives.

Incorporation of multitask learning (MTL) [9] enables the HC-MTDNN to simultaneously learn and optimize multiple related tasks. This approach leverages shared representations, improving generalization between tasks, and enhancing the model's ability to detect various types of anomalies.

The primary advantage of the proposed hierarchical multitask architecture lies in its ability to progressively refine classification through shared supervision and structured information flow. HC-MTDNN processes each IoT flow in three gated stages:

- benign vs. malicious screening,
- attack-family categorization,

- fine-grained subtype classification.

Intermediate logits from one task are concatenated or transformed before being passed to the next stage, allowing downstream classifiers to condition on earlier decisions. The shared encoder allows for generalization and reduces redundant computation.

Additionally, the architecture strengthens feature reuse, stabilizes rare subcategories, and diversifies error signals. The model also maintains hierarchical consistency by using gating and attention mechanisms to align predictions across levels, thus minimizing contradictions like predicting "mirai.scan" when the binary classifier identifies a sample as benign.

Finally, this multitask setup often leads to faster convergence and improved generalization, as the shared encoder can absorb additional tasks with minimal reconfiguration, making the model adaptable to evolving threat landscapes in complex IoT environments.

The proposed model is designed for flow-based anomaly detection, using network metadata such as packet counts, flow durations, and protocol attributes to identify threats. This approach excels at detecting volumetric attacks (e.g., DDoS) and protocol anomalies, but is inherently limited for payload-intensive threats (e.g., SQL injections or XSS), where content analysis is required. This trade-off ensures deployability on edge devices, prioritizing speed and low resource usage over comprehensive payload inspection.

With 245 249 total parameters, the model occupies just 2.4 MB of disk space. The average inference latency is 122 µs per sample, corresponding to roughly 8 200 predictions per second, which falls within the real-time requirements for gateway-level traffic inspection. This parameter sharing and conditional gating eliminate redundant computation, allowing HC-MTDNN to be deployable on memory and power constrained IoT edge devices.

We evaluate the proposed model across four datasets: CICIoT2023 [10], N-BaIoT [11], Bot-IoT [12], and EdgeIIoT [13]. These datasets cover diverse environments and traffic profiles. They span distinct deployment scenarios: CICIoT2023 imitates modern smarthome traffic, N-BaIoT isolates single device compromises typical of consumer gadgets, Bot-IoT replicates campus-scale probing and DDoS, while EdgeIIoT captures latency-sensitive industrial control flows.

Across all experiments, HC-MTDNN delivers accuracy and efficiency that match the practical constraints of edge hardware. On the binary task, it attains macro F1 between 97% and 100% on every dataset, while at the coarse-level it reaches 99.5% accuracy on CICIoT2023, 97% on EdgeIIoT and Bot-IoT, and a near-perfect 99.99% on N-BaIoT. Even at the most demanding fine-grained level (34 classes in CICIoT2023, 9 – 10 classes on the other sets) the network keeps the weighted F1 above 93% for Bot-IoT and the macro F1 above 83% for the heavily imbalanced N-BaIoT. The results demonstrate robust multitask performance across all classification hierarchies, highlighting the robustness and effectiveness in accurately detecting and classifying anomalies within complex IoT environments.

The remainder of this paper is structured as follows. Section 2 reviews related work and discusses the limitations of existing approaches. Section 3 presents, in detail, the proposed architecture of the HC-MTDNN model. Section 4 describes the datasets, experimental setup, and evaluation metrics used. Section 5 reports and analyzes the experimental results, comparing them with established baselines. Finally, Section 6 summarizes the findings, implications, and suggests directions for future research.

2. Related Work

The extensive deployment of IoT devices across various sectors has significantly increased the prevalence of cyber attacks, underscoring the need for effective anomaly and intrusion detection mechanisms [14]. Traditional anomaly detection strategies, such as rule-based systems and fixed threshold methods, frequently encounter difficulties due to the heterogeneous, high-volume, and dynamically changing nature of IoT-generated data. Such conventional methods often produce false alarms or fail to identify subtle attacks, particularly in environments where multiple distinct data streams are processed simultaneously. A summary of recent intrusion detection studies is provided in Tab. 1 which categorizes the approaches by model type, dataset, key contributions, and gaps addressed by the proposed method.

2.1. Traditional Machine Learning Approaches

Several recent studies have applied traditional machine learning (ML) approaches using the CICIoT2023 dataset [10]. The authors of [8] introduced a random forest-based intrusion detection framework specifically addressing class imbalance, achieving notable performance improvements of 3.72% in precision, 3.75% in recall and 4.69% in F1 score compared to existing methodologies. Their method also showed an enhancement of 7.9% in the F1 score for underperforming classes. In another comparative analysis, multiple machine learning algorithms, including logistic regression, AdaBoost, perceptron, MLP, random forest (RF) and hist-gradient boosting, were evaluated for different classification scenarios (binary, eight class and 34-class), with RF outperforming others in accuracy, while hist-gradient boosting excelled in computational efficiency [15].

Addressing specific attack types, the researchers developed a specialized intrusion detection system (IDS) that employs hierarchical feature selection coupled with the CatBoost algorithm, targeting DoS, DDoS, and Mirai attack variants. This approach achieved fast prediction times and high accuracy, significantly improving cybersecurity defenses against sophisticated threats [16]. Similarly, a comprehensive evaluation, as presented in [7], emphasized the broad spectrum of threats encapsulated by the CICIoT2023 dataset, reinforcing its utility in benchmarking classification methods. Parallel, lightweight ML models – such as decision trees, closest neighbors k, RF,

and naive Bayes – were evaluated, demonstrating impressive precision and processing efficiency, notably the ability of decision trees to classify nearly three million instances per second [17].

In [18], refined preparation and feature selection phases are investigated through cooperative game theory, and RF achieves 99% accuracy on the original CICIoT2023 dataset. However, the accuracy decreased slightly with novel features, highlighting complexities in feature engineering for IoT intrusion detection. While these single-task machine learning approaches achieve high accuracy on balanced classes and specific attack types, they often treat classification tasks independently, overlooking intertask dependencies such as shared patterns between binary detection and multiclass categorization. This leads to redundant computations and limited generalization of diverse or evolving threats.

2.2. Multitask Learning and Lightweight Models

To overcome limitations inherent in single-task or parallel-output-head models, researchers have explored multitask learning frameworks, aiming to enhance anomaly detection performance by leveraging interrelated tasks [19]. The CICIoT2023 dataset, a comprehensive and realistic benchmark, has been widely utilized to evaluate the effectiveness of these advanced models, particularly emphasizing improvements in the detection of low-profile attacks [20]. Additionally, due to the resource-constrained nature of many IoT devices, significant research has focused on developing lightweight models optimized for efficient deployment in such environments.

In resource-sensitive IoT scenarios, the authors of [21] introduced DL-BiLSTM, integrating DNN and bi-LSTM networks, along with incremental PCA and dynamic quantization to optimize model performance for limited-resource environments. Furthermore, in [22], an innovative VGGIncepNet model was proposed that converts non-image network data into image format to leverage CNN feature extraction capabilities, significantly outperforming established NLP-based models such as BERT and XLNet in CICIoT2023.

In [23], edge-based deep learning models are presented that employ 1D-CNN architectures optimized by preprocessing techniques that address data imbalance and distribution discrepancies, achieving a robust F1 score of 93.8%. Furthermore, the authors of [24] proposed a cost-sensitive autoencoder (CSAE)-based ensemble approach, demonstrating exceptional accuracy rates for both binary and multiclass classifications.

In article [25], a DGConv-IDS was developed. It is a lightweight autoencoder and CNN-based model tailored for resource-limited IoT environments. The model used sliding-window techniques to manage computational overhead while providing real-time DDoS detection. Similarly, in [26], the convolutional Kolmogorov-Arnold network (CKAN) is introduced which integrates Kolmogorov-Arnold layers into convolutional neural networks, achieving high performance with fewer parameters. The authors of [27] proposed hybrid models, such as the autoencoder-CNN and transformer-DNN frameworks, emphasizing reshaping network traffic, handling class imbalance, and improving feature extraction capabilities across multiple datasets, including CICIoT2023.

Existing lightweight deep learning models prioritize computational efficiency and edge deployment, but often lack hierarchical structures for progressive refinement from binary to fine-grained classification, leading to potential error propagation in multiclass scenarios. Moreover, they often incorporate conditional mechanisms such as dynamic gating to adapt features based on prior task outputs.

2.3. Dataset-specific and Hybrid Models

The authors of [28] propose an efficient anomaly detection mechanism for IoT architectures using DNN, with a specific focus on feature selection through mutual information (MI). The study uses the Bot-IoT 2020 dataset and evaluates the performance of several deep learning models, including DNN, CNN, RNN, and RNN variants. The authors demonstrate that selecting the top 16 to 35 MI-based features, instead of using all 80 features, resulted in only negligible performance degradation while significantly reducing model complexity. The proposed DNN-based model achieves an accuracy of 99.01% with a false alarm rate (FAR) of 3.9%.

In [29], XAI-IoT, an explainable AI framework is introduced designed to enhance multi-class anomaly detection and defect type classification in IoT systems. The framework incorporates seven explainable AI (XAI) techniques, including SHAP, LIME, CEM, and LOCO, to evaluate the importance in model predictions. Experimental validation was performed on two datasets: one collected from IoT-based MEMS sensors and the other from IoT botnet attacks (N-BaIoT). The results indicate that single-model approaches delivered better performance on the MEMS dataset, while ensemble-based models outperformed on the N-BaIoT dataset. The use of XAI techniques allowed the identification of critical features that influenced model decisions in both contexts.

In [30], an IDS for detecting DoS attacks in IoT networks by relying on ML algorithms is described. The study compared four classifiers: decision tree (DT), RF, K-nearest neighbor (kNN), and support vector machine (SVM), to determine the most effective model for classifying DoS traffic. Feature selection was enhanced using correlation-based feature selection (CFS) and a genetic algorithm (GA), with the IoTID20 data set used for training, which includes real-time traffic data with simulated DoS attacks. The DT and RF classifiers, using GA-selected features (13 features), achieved 100% accuracy, precision, recall, and F1 scores. The DT model outperformed RF in terms of computational efficiency. The study emphasizes the effectiveness of the IoTID20 dataset and the chosen feature selection methods to improve the performance of the model.

The authors of [31] introduce DeepDetect, a hybrid deep learning model for anomaly detection in IoT networks which combines CNN, GRU, and Bi-LSTM to improve network traffic analysis. The hierarchical CNN structure captures spatial features, while the problem of GRU mitigates the vanishing

Tab. 1. Summary of recent intrusion detection studies in IoT environments.

Ref.	Model type	Dataset	Key contribution
[8]	RF	CICIoT2023	Improved class performance; tackled class imbalance
[7]	ML	CICIoT2023	Comprehensive benchmarking across attack categories
[18]	RF + game theory	CICIoT2023	Cooperative game theory feature selection
[21]	DL-BiLSTM+PCA	CICIoT2023	Resource-efficient BiLSTM with dynamic quantization
[22]	VGGIncepNet (CNN-based)	CICIoT2023	Converted traffic to images; outperformed BERT/XLNet
[23]	1D-CNN	CICIoT2023	Edge-based detection with preprocessing for imbalance
[24]	CSAE	CICIoT2023	High accuracy for binary and multiclass tasks
[25]	DGConv-IDS	CICIoT2023	Real-time DDoS detection via sliding windows
[26]	CKAN	CICIoT2023	Low-parameter model with high performance
[27]	AE-CNN, transformer-DNN	CICIoT2023	Multi-dataset approach; class imbalance handling
[28]	DNN, CNN, RNN variants	BoT-IoT 2020	Mutual-information feature selection (top 16 – 35 features)
[29]	Ensemble + SHAP, LIME	N-BaIoT, MEMS	XAI-IoT with comparative model/XAI analysis
[30]	DT, RF, kNN, SVM	IoTID20	GA-based feature selection; 100% metrics with DT
[31]	CNN-GRU-BiLSTM	NSL-KDD	High accuracy; temporal modeling with low FAR
[32]	XGBoost, RF	IoT-23 combined	PySpark-based scalable real-time IDS
[33]	TinyML + DT, RF, KNN	Custom + real IoT LAN data	Energy/memory efficient IDS using TinyML
[34]	CNN + LSTM/-GRU/BiLSTM	NSL-KDD, BoT-IoT, MQTTset	Addressed class imbalance with SMOTE and class weighting
[35]	RF vs. DNN	CICIoT2023	Multilevel classification and feature selection study
[36]	Transformer, CNN + LSTM, DNN	CICIoT2023	Multi-class top accuracy with transformer
[37]	PCA + expansion – compression NN	UNSW-NB15, Bot-IoT	Lightweight NN with NID loss; 99.99% binary accuracy
[38]	Multi-stage pipeline	CIC-IDS-2017, CSE-CIC-IDS-2018	Adjustable zero-day detection; low bandwidth/latency
[39]	MI + attention CNN	Edge-IoTset, IoTID20, ToN IoT, CIC-IDS2017	99.81% average accuracy; attention helps low-instance classes
[40]	Multitask LSTM + feature selection	IoT-23, EU CEF VAR-IoT, 18-device pcaps	Joint malware detection/identification; SMOTE-ENN + XGBoost-/SULOV

gradients and learns sequential dependencies. The Bi-LSTM captures long-term dependencies from both forward and backward contexts, improving temporal analysis. Based on the NSL-KDD dataset, DeepDetect achieved 99.12% accuracy for binary classification and 99.31% for multiclass classification, demonstrating superior performance with a lower false positive rate and higher detection rate compared to other deep learning-based IDS.

Paper [32] presents a real-time IDS for IoT networks using multiclass ML techniques. Using the IoT-23 combined dataset, which includes more than 1.4 million records of various types and benign traffic, the class imbalance with SMOTE and the applied SelectKBest is addressed. IDS was built on a PySpark architecture to support scalable training and inference. Five ML models were tested using a one-versus-rest approach, with XGBoost achieving the highest accuracy (98.89%) and RF delivering the fastest inference time (0.0311 s), demonstrating a strong balance between speed and accuracy.

In [33], an ML-based IDS is developed tailored for resource-constrained IoT devices, emphasizing energy and memory efficiency. By integrating TinyML with traditional ML models, the study addressed key challenges in limited resource environments. A major contribution was the creation of a rich validation dataset combining prior work, laboratory experiments, and real-world metrics across IoT layers (end devices, edge, cloud), including both normal and malicious traffic. The system was tested in a LAN based setup with extended edge/cloud components, using models such as DT

(99.5% accuracy), KNN (96.5%), Naive Bayes (97.3%) and RF (98.3%). The results highlighted a trade-off between accuracy and training time, with more accurate models requiring longer training.

The authors of [34] propose a DL-based anomaly detection framework for IoT networks, utilizing a combination of RNN and CNN. Their study developed lightweight models that employ LSTM, BiLSTM, and GRU architectures to perform binary and multiclass classification tasks. CNNs were also incorporated for feature selection to enhance detection performance. Models were trained and evaluated on several widely used datasets, including NSL-KDD, BoT-IoT, IoT-NI, IoT-23, MQTT, MQTTset, and IoT-DS2. To address the issue of class imbalance within these datasets, the authors applied class weighting techniques during training and employed the Borderline-SMOTE algorithm to generate synthetic samples and balance the training data distribution.

In [35], an anomaly detection study is conducted in IoT-based healthcare systems using the CICIoT2023 data set. Their investigation involved multilevel classification architectures, including 2-class (binary), 8-class, and 34-class models. The authors explored two training approaches: one using the full set of features and the other using a reduced feature subset. To address class imbalance, they applied SMOTE. Their evaluation demonstrated that, on both training paths and on the balanced CICIoT2023 dataset, the RF classifier consistently outperformed the DNN model.

These dataset-specific and hybrid models demonstrate strong performance on individual benchmarks but typically do not integrate multitask hierarchies for handling interdependent classification tasks, such as simultaneous binary and fine-grained detection. This results in missed opportunities for shared learning and efficiency in resource-constrained settings.

2.4. Advanced Hybrid and Attention-based Models

The authors of [36] propose a transformer-based IDS evaluated on the CICIoT2023 dataset. Their model leveraged self-attention mechanisms to effectively handle multi-class intrusion detection, achieving a high accuracy of 99.40%. After comparing seven neural network models, they found the transformer to be the most effective solution for multiclass tasks, while DNN and CNN+LSTM models performed best for binary classification.

In [37], a lightweight neural network-based IDS is presented that uses PCA for feature dimensionality reduction. It relies on an expansion compression classifier architecture with inverse residual blocks and channel shuffle operations to minimize computational cost, and a loss of NID to mitigate class imbalance. Evaluated on UNSW-NB15 and Bot-IoT, it achieves a precision of up to 99.99% (F1 98.81%) for binary detection and multiclass accuracies of 86.11% and 96.15%, respectively, without altering its core architecture.

Work [38] introduces a multi-stage approach for hierarchical IDS with a three-stage anomaly detection pipeline: fast filtering by anomaly score, confidence-based attack classification, and strict thresholding to separate unknown attacks from false positives, enabling efficient, adjustable detection of binary and multiclass intrusions, including zero-day attacks, in the CIC-IDS2017 and CSE-CIC-IDS-2018 datasets. Each stage can be deployed independently to minimize bandwidth usage and prediction latency without requiring retraining.

The authors of [39] propose an attention-based CNN for IDS, using MI for feature selection and an attention mechanism to improve learning in low-instance classes. The proposal is evaluated in Edge-IoTset, IoTID20, ToN IoT, and CIC-IDS2017. It achieves an average accuracy of 99.81%, with 98.02% precision, 98.18% recall, and an F1 score of 98.08%.

In [40], a multitask LSTM is proposed for the detection and identification of IoT malware through behavioral traffic analysis, performing both benign/malicious classification and malware type prediction on 145 pcaps from 18 devices and on the IoT-23 and EU CEF VARIoT datasets. Features are organized into flow-, flag-, and payload-related modalities, each subjected to recursive XGBoost and SULOV feature selection before merging, with class imbalance addressed using SMOTE-ENN and extensive experiments on imbalance techniques, feature selection, and modality fusion.

Advanced hybrid and attention-based models enhance temporal and spatial feature analysis, but often miss conditional feature modulation, where prior task outputs dynamically influence subsequent processing. This can limit adaptability in hierarchical scenarios.

Although significant progress has been made in IoT anomaly detection, existing approaches typically address tasks individually, deploying separate single-task models, thus overlooking valuable shared information among related classification tasks. Moreover, hierarchical cascading, progressively refining anomaly classification from broad to detailed levels, has rarely been explicitly combined with computational efficiency strategies optimized for deployment in resource-constrained IoT environments. Most lightweight models neglect hierarchical classification or fail to incorporate dynamic feature gating and hierarchical feature reuse mechanisms necessary for efficiency in real-time scenarios.

Table 1 summarizes mentioned studies on intrusion detection.

3. Proposed Method

3.1. Model Architecture

The proposed architecture, as illustrated in Fig. 1, is a multi-stage neural network processing incoming IoT sensor streams using a shared feature encoder that learns a common representation. The data are then refined by three task-specific heads: a binary anomaly detector, a coarse (categorical) classifier, and a fine-grained (subcategorical) classifier.

This hierarchical structure enables the model to progressively refine the predictions: first, by identifying anomalies, then by categorizing broad attack types, and finally by distinguish specific subcategories.

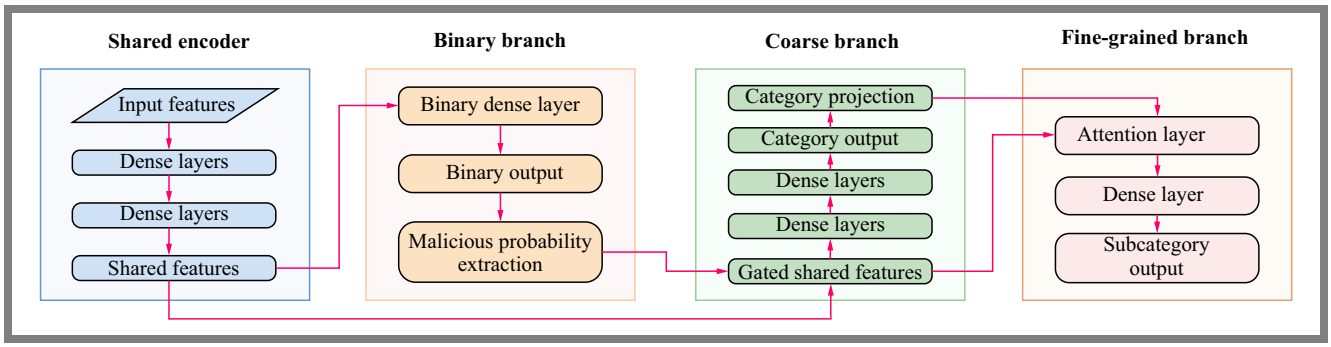

Fig. 1. Overall architecture of the proposed HC-MTDNN model.

The core of the architecture is a deep neural backbone that extracts general-purpose features from raw IoT data. This encoder is hard-shared across all tasks, ensuring that early layers capture patterns common to anomaly detection, while later layers refine these features for task-specific objectives. Shared encoders are a standard approach in multi-task learning to reduce redundancy and improve generalization.

The binary branch is the lightweight head that receives the shared features and outputs a scalar anomaly probability indicating whether the input is normal or anomalous. The output is also reused as a gating signal to modulate downstream processing. Gating mechanisms are commonly used in attention-based architectures to dynamically suppress irrelevant features.

The coarse branch performs coarse-grained anomaly classification. It applies an attention mechanism to the shared features, refining them using the binary gating signal and preliminary category logits. The attention mechanism computes feature-wise importance weights, focusing on dimensions most relevant to distinguishing broad anomaly categories.

The fine-grained branch identifies fine-grained anomaly subtypes. It takes as input a fusion of attention-refined features from the coarse branch, the binary gating signal, and the predicted category. This hierarchical design mirrors strategies used in multilevel classification tasks, where coarse predictions inform finer distinctions.

All components are jointly trained end-to-end. The shared encoder is updated by all three tasks, encouraging it to learn useful features in binary, categorical, and subcategorical decisions. Selective feature flow ensures that computational effort is focused where needed the most (e.g., suppressing processing for benign samples).

3.2. Input Representation and Shared Encoder

The model processes a static feature vector $\mathbf{x} \in \mathbb{R}^d$, which encapsulates critical IoT traffic characteristics.

These include network flow statistics (e.g., packet counts, byte rates), packet-level attributes (i.e., protocol types, payload sizes), and temporal dynamics such as traffic variations over sliding windows. The shared encoder consists of three dense layers with ReLU activation (256, 256, and 128 neurons, respectively), designed to extract foundational representations while minimizing redundancy between tasks. The final output of the shared encoder is defined as:

$$\mathbf{x}_{\text{shared}} = \text{ReLU}\left[\text{Dense}_{128}\left(\text{ReLU}(\text{Dense}_{256}(\mathbf{x}))\right)\right], \quad (1)$$

where nested ReLU activations ensure non-linearity at each layer.

This design aligns with multitask learning principles, enabling knowledge transfer by learning task-agnostic representations.

3.3. Binary Classification Branch

The binary classification branch employs a shallow structure comprising a single dense layer (128 neurons, ReLU activation) followed by a softmax output:

$$\hat{\mathbf{y}}_{\text{bin}} = \text{softmax}\left[\text{Dense}_2\left(\text{Dense}_{128}(\mathbf{x}_{\text{shared}})\right)\right]. \quad (2)$$

This prioritizes computational efficiency for edge deployment, balancing accuracy and inference speed. The output $\hat{\mathbf{y}}_{\text{bin}}$ serves dual purposes: direct binary anomaly detection (normal vs. anomalous) and generating a gating signal $p_{\text{malicious}}$ to modulate downstream processing.

3.4. Gated Coarse Classification Branch

To refine predictions, the malicious probability $p_{\text{malicious}}$ is extracted from $\hat{\mathbf{y}}_{\text{bin}}$ via a lambda layer:

$$p_{\text{malicious}} = \Lambda\left(z \mapsto z[:,1]\right)(\hat{\mathbf{y}}_{\text{bin}}), \quad (3)$$

where $z[:,1]$ isolates the probability of the anomalous class.

A sigmoid-activated dense layer then generates gating weights $\mathbf{g} \in [0,1]^{128}$:

$$\mathbf{g} = \sigma\left(\text{Dense}_{128}(p_{\text{malicious}})\right). \quad (4)$$

These gating weights dynamically modulate shared features by selectively emphasizing relevant dimensions and suppressing irrelevant ones, particularly for benign samples. Formally, this modulation is implemented as element-wise multiplication between gating weights and shared features.

$$\mathbf{x}_{\text{gated}} = \mathbf{x}_{\text{shared}} \odot \mathbf{g}. \quad (5)$$

Dynamic gating significantly reduces unnecessary computations by minimizing redundant feature processing for benign inputs, enhancing computational efficiency crucial for resource-constrained IoT environments. The gated features are then processed by two dense layers (128 neurons, ReLU)

to produce coarse-grained classification outputs:

$$\hat{\mathbf{y}}_{\text{coarse}} = \text{softmax}\left[\text{Dense}_{\text{coarse}}\left(\text{Dense}_{128}(\mathbf{x}_{\text{gated}})\right)\right] . \quad (6)$$

3.5. Fine-grained Classification Branch

The fine-grained classification branch incorporates two components:

Semantic projection is the 8-class output, where $\hat{\mathbf{y}}_{\text{coarse}}$ is projected into the shared feature space to embed coarse-grained priors:

$$\mathbf{x}_{\text{proj}} = \text{ReLU}\left(\text{Dense}_{256}(\hat{\mathbf{y}}_{\text{coarse}})\right) . \quad (7)$$

This projection aligns the semantic context with latent features, enhancing cross-task knowledge transfer.

The **cross-task attention** mechanism fuses $\mathbf{x}_{\text{shared}}$ and $\mathbf{x}_{\text{proj}}$:

$$\mathbf{x}_{\text{att}} = \text{Attention}\left(\mathbf{x}_{\text{shared}}, \mathbf{x}_{\text{proj}}, \mathbf{x}_{\text{proj}}\right) , \quad (8)$$

where queries $\mathbf{x}_{\text{shared}}$ and keys/values $\mathbf{x}_{\text{proj}}$ compute feature-wise importance weights.

The attended vector $\mathbf{x}_{\text{att}}$ is concatenated with $\mathbf{x}_{\text{gated}}$:

$$\mathbf{x}_{\text{concat}} = \text{Concat}(\mathbf{x}_{\text{gated}}, \mathbf{x}_{\text{att}}) . \quad (9)$$

This combined representation is passed through a dense layer (ReLU) and batch normalization:

$$\mathbf{x}_{\text{norm}} = \text{BatchNorm}\left[\text{ReLU}\left(\text{Dense}_{128}(\mathbf{x}_{\text{concat}})\right)\right] , \quad (10)$$

before yielding the final fine-grained prediction:

$$\hat{\mathbf{y}}_{\text{fine}} = \text{softmax}\left[\text{Dense}_{\text{fine}}(\mathbf{x}_{\text{norm}})\right] . \quad (11)$$

This hierarchical fusion takes advantage of coarse-level context to constrain fine-grained predictions, improving robustness for closely related subtypes.

4. Experimental Setup and Evaluation

4.1. Dataset Overview

CICIoT2023 [10] is a comprehensive benchmark data set that captures network traffic from 105 real IoT devices in a laboratory environment. It includes 33 distinct attacks across seven categories (DDoS, DoS, Reconnaissance, Web-based attacks, BruteForce, Spoofing, Mirai botnet) and benign traffic (e.g., video streaming, sensor data). Features such as flow duration, packet length, and protocol types are extracted from pcap files and stored in CSV format. Baseline models (LR, RF) have been evaluated on binary, 8-class, and 34-class tasks, making it ideal for comparative studies.

Bot-IoT [12] combines real and simulated IoT traffic with various cyberattack scenarios. Developed using a realistic testbed, it addresses limitations of older datasets (e.g., outdated attack patterns, poor labeling). Its validity has been confirmed through statistical analysis and ML experiments.

N-BaIoT [11] focuses on botnet detection, containing traffic from nine commercial IoT devices infected with Mirai and Bashlite malware. It includes over 7 million records with 115 features, classified into ten categories (primarily DDoS and remote access attacks).

Edge-IIoTset [13] is a cybersecurity data set for IoT/IIoT applications, supporting centralized and federated learning. Generated using a custom testbed, it includes 14 attack types across five categories (DoS/DDoS, information gathering, MITM, injection, malware). Of 1 176 initial features, 61 were selected based on correlation and domain knowledge, ensuring efficient model training.

4.2. Evaluation Metrics

The performance of the hierarchical multitask DNN is assessed using accuracy, precision, recall, F1 score, and AUC-ROC. The definitions of the aforementioned terms are provided below:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} , \quad (12)$$

$$\text{Precision} = \frac{TP}{TP + FP} , \quad (13)$$

$$\text{Recall} = \frac{TP}{TP + FN} , \quad (14)$$

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} , \quad (15)$$

where TP, TN, FP, and FN denote true/false positives/negatives.

The F1 score is prioritized due to class imbalance in IoT security datasets.

4.3. Model Training

The model is trained on 80% of the data (64% training, 16% validation), with 20% held for testing. Pre-processing includes label mapping and feature normalization. Hyperparameters are tuned iteratively: learning rate 10^{-3} (Adam optimizer), batch size 512, epochs 127 (one CSV file per epoch to manage memory).

The loss function is:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{bin}}\,\mathcal{L}_{\text{bin}} + \lambda_{\text{int}}\,\mathcal{L}_{\text{int}} + \lambda_{\text{fine}}\,\mathcal{L}_{\text{fine}} , \quad (16)$$

with λ_{bin}, λ_{int}, and λ_{fine} as task-specific weights.

Figure 2 shows training/validation curves. The binary head converges faster than multiclass heads, reflecting its simplicity. The validation accuracy plateaus earlier for coarse tasks, suggesting diminishing returns beyond 80 epochs.

5. Experimental Results

The proposed lightweight multitask DNN demonstrates robust performance across the IoT datasets used. With 245 249 parameters (2.4 MB in size) and an average inference time of 122 µs per flow, the model is optimized for real-time deployment on resource-constrained devices. Hierarchical classification tasks are evaluated, with macro and weighted average metrics summarized in Tab. 2.

5.1. Binary Classification

The binary classification results (Tab. 3, Fig. 3) show near-perfect or perfect separation between benign and malicious

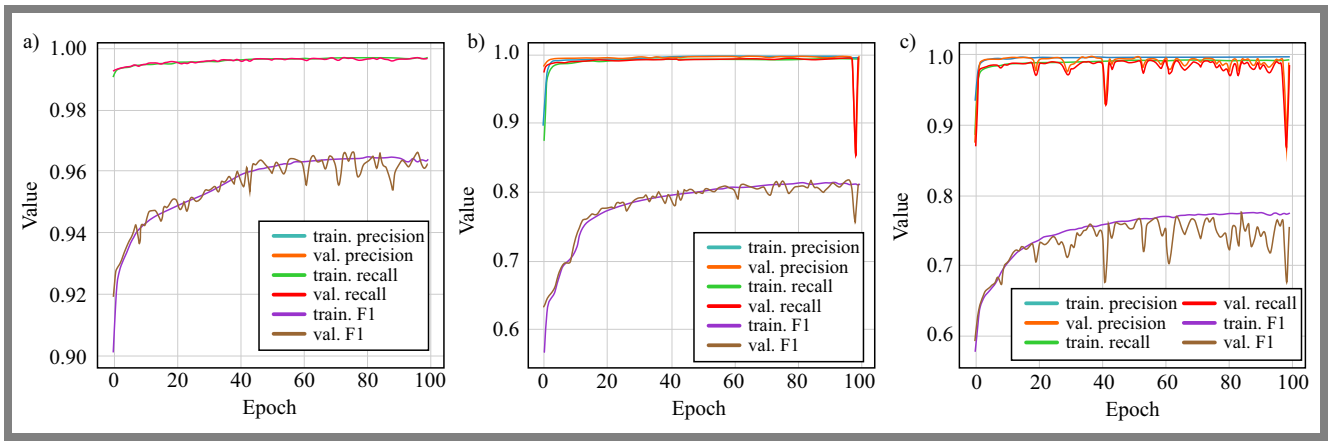

Fig. 2. Performance of training and validation of the proposed model on the CICIoT2023 data set: a) binary classification b) 8-class category classification, and c) 34-class fine-grained subtype classification.

Tab. 2. Summary of the macro- and weighted-average metrics results on the test splits of the four datasets.

Dataset	Level	Macro			Weighted		
		P	R	F_1	P	R	F_1
CICIoT 2023	Binary	96.0	97.0	97.0	100.0	100.0	100.0
	Category	92.0	78.0	82.0	99.0	99.0	99.0
	Subtype	83.0	75.6	77.2	99.2	99.2	99.2
N-BaIoT	Binary	100.0	100.0	100.0	100.0	100.0	100.0
	Category	100.0	100.0	100.0	100.0	100.0	100.0
	Subtype	97.0	89.0	87.0	91.0	88.0	83.0
Bot-IoT	Binary	98.5	96.7	97.6	99.3	98.4	98.8
	Category	99.2	95.1	97.0	99.2	97.1	98.1
	Subtype	90.4	87.9	87.7	95.0	93.7	93.6
EdgeIIoT	Binary	100.0	100.0	100.0	100.0	100.0	100.0
	Category	89.0	87.0	87.0	98.0	97.0	97.0

traffic across all data sets. N-BaIoT, Bot-IoT, and Edge-IIoT report 100% precision, recall, and F1 score, confirmed by diagonal dominance in confusion matrices. For CICIoT2023, 4% of benign flows are misclassified as malicious. This occurs because low-level attacks (i.e., reconnaissance scans) mimic benign behaviors, creating subtle overlaps in header-level features like packet size distributions and interarrival times. These patterns suggest that the model struggles to distinguish benign traffic from low-intensity adversarial activities that exploit normal protocol behaviors, such as slow port scans or HTTP GET requests.

5.2. Coarse-grained Classification

Coarse-grained classification (Tab. 4 and Fig. 4) identifies broader attack families (e.g., DDoS, DoS, Mirai, Reconnaissance). On CICIoT2023, the model attains 99.5% weighted accuracy, but struggles with underrepresented classes like web-based (40% recall) and BruteForce (24% recall). These errors arise from feature overlap in HTTP methods and port usage, where web-based attacks share characteristics with reconnaissance activities (e.g., POST requests, standard ports 80/443). Edge IIoT achieves 97.13% accuracy, though password attacks and SQL injections exhibit sub-80% precision.

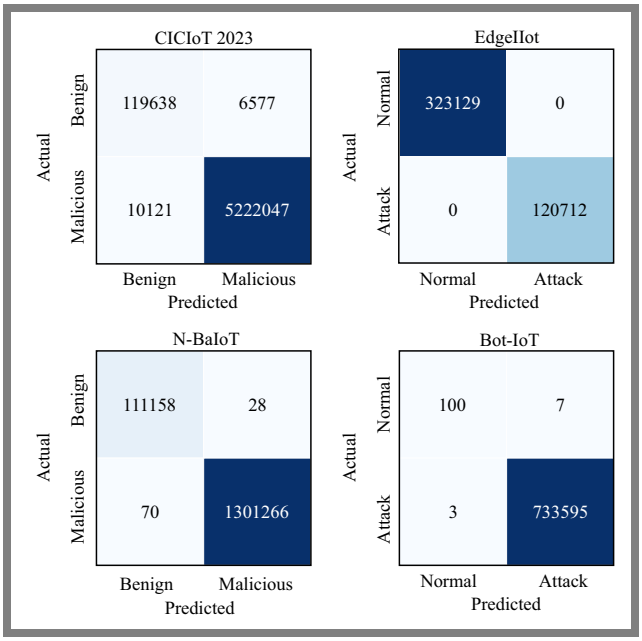

Fig. 3. Confusion matrices for binary anomaly detection across the four benchmark datasets. Each matrix shows the distribution of predicted versus actual classes (benign vs. malicious).

This reflects structural similarity in authentication-related behaviors, such as repeated log-in attempts over TCP, which the model conflates with other high-frequency traffic.

Bot-IoT and N-BaIoT maintain near-perfect scores across all categories, with dominant classes (DDoS, DoS, Mirai) classified with over 99% precision and recall. Confusion matrices highlight diagonal dominance for major categories, confirming the strength in distinguishing broad attack types through rate-driven features, e.g., packet-per-second rates, and flow duration.

5.3. Fine-grained Classification

Fine-grained classification (Tab. 5, Fig. 5) targets specific subtypes (DDoS UDP Flood, OS Fingerprinting). Although the weighted accuracy remains high (99.24% on CICIoT2023), the performance varies significantly for rare or overlapping

Tab. 3. Performance of the binary classification of the proposed model across four benchmark data sets.

Dataset	Class/metrics	Precision	Recall	F1 score	Support
CICIoT2023	Benign	92.00%	95.00%	93.00%	126 215
	Malicious	100.00%	100.00%	100.00%	5 232 168
	Accuracy			100.00%	
	Macro avg	96.00%	97.00%	97.00%	5 358 383
	Weighted avg	100.00%	100.00%	100.00%	5 358 383
N-BaIoT	Benign	100.00%	100.00%	100.00%	111 186
	Malicious	100.00%	100.00%	100.00%	1 301 336
	Accuracy			100.00%	
	Macro avg	100.00%	100.00%	100.00%	1 412 522
	Weighted avg	100.00%	100.00%	100.00%	1 412 522
Bot-IoT	Normal	97.1%	93.5%	95.2%	107
	Attack	100.00%	100.00%	100.00%	733 598
	Accuracy			100.00%	
	Macro avg	98.5%	96.7%	97.6%	733 705
	Weighted avg	99.3%	98.4%	98.8%	733 705
EdgeIIoT	Normal	100.00%	100.00%	100.00%	323 129
	Attack	100.00%	100.00%	100.00%	120 712
	Accuracy			100.00%	
	Macro avg	100.00%	100.00%	100.00%	443 841
	Weighted avg	100.00%	100.00%	100.00%	443 841

subcategories. Dominant subtypes like DDoS TCP Flood and Mirai achieve an F1 score, driven by distinctive volumetric patterns (sustained high packet rates, unique combinations of TCP flags).

However, minority classes such as XSS (12.18% recall) and Uploading attack (0% precision/recall) are systematically misclassified. In CICIoT2023, SqlInjection, CommandInjection, and BrowserHijacking are frequently conflated due to shared HTTP methods (POST) and common port usage, which the flow-level metadata cannot disentangle. The Bot-IoT OS fingerprint class is largely absorbed by the service scan and TCP categories, likely because the aggregate statistics do not capture TTL variations critical to fingerprinting. N-BaIoT TCP flag variants (ACK flood) exhibit 0% recall, indicating insufficient feature representation for flag-based distinctions, such as ACK-ratio thresholds.

5.4. Ablation Study

An ablation study in CICIoT2023 (Tab. 6) underscores the importance of architectural components in ensuring good performance. Removal of the shared encoder, a core element that enables MTL, degrades category classification by 10.41 percentage points and subcategory classification by 5.8 points. This highlights the necessity of shared representations to propagate discriminative features across hierarchical tasks.

The gating mechanism, which propagates features from binary tasks to category tasks, improves the category F1 score by approx. 4 points, while attention and batch normalization contribute approx. 3-point gains across multiclass tasks. These findings emphasize the interdependence of architectural elements in maintaining hierarchical consistency and mitigating the propagation of errors.

5.5. Baseline Comparison

Baseline comparisons (Tab. 7) further validate the model's superiority. Against classical methods like RF and Adaboost [10], multitask DNN achieves higher recall and F1 score, particularly in high-granularity settings. For example, in the 34-class classification on CICIoT2023, the model outperforms RF by 10 percentage points in the recall and 4 points in F1 score.

This gap widens with class imbalance and feature overlap, demonstrating the advantage in leveraging shared patterns across tasks to mitigate data scarcity in minority classes.

6. Discussion

The proposed lightweight multitask deep neural network (DNN) demonstrates robust performance across diverse IoT intrusion detection tasks, validating its suitability for real-time deployment in resource-constrained environments. Using shared representations across hierarchical classification levels, the model achieves high accuracy (up to 100% weighted F1 score) while maintaining computational efficiency. These results affirm the advantages of multitask learning in balancing generalization and specificity. However, limitations emerge in distinguishing rare or structurally similar subcategories, highlighting critical areas for improvement.

The model excels at identifying dominant attack patterns, particularly volumetric floods such as DDoS UDP_Flood, Mirai and protocol-driven anomalies, e.g., SYN floods. Across all datasets, these classes achieve near-perfect precision (over 99%) and recall ($> 99\%$), driven by rate-based features (flow duration, packet-per-second rates) and distinct TCP/UDP flag patterns. For example, N-BaIoT and Bot-IoT exhibit 100%

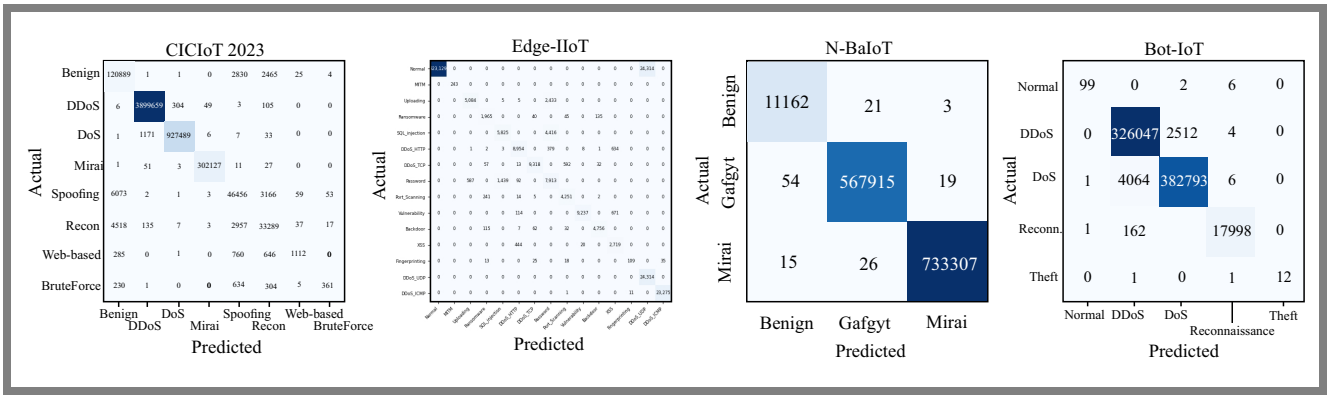

Fig. 4. Confusion matrices for coarse-grained attack classification across the four datasets: CICIoT2023, N-BaIoT, Bot-IoT, and Edge-IIoT.

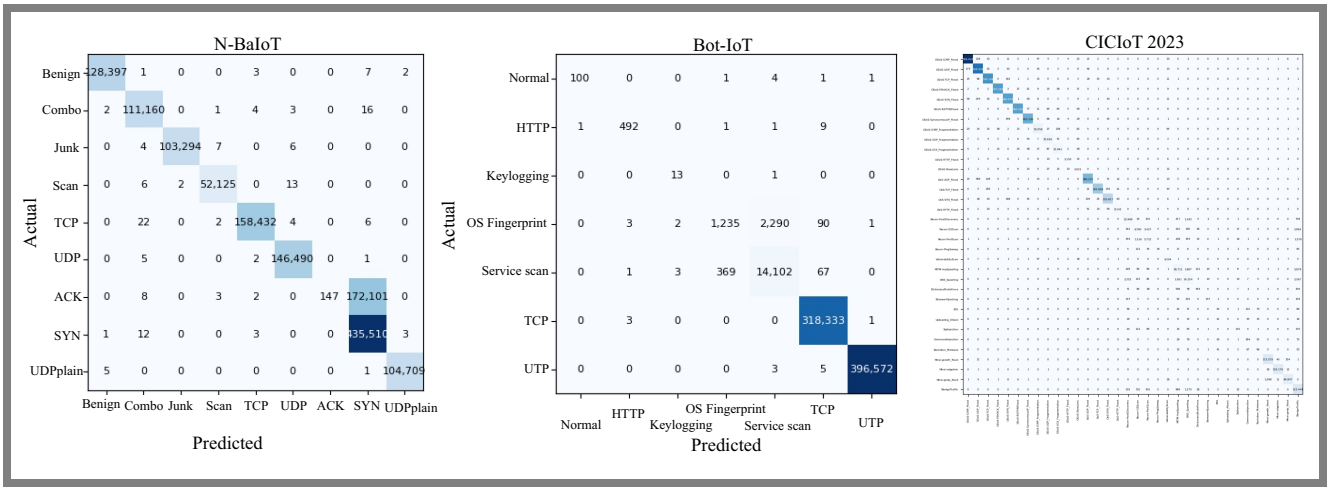

Fig. 5. Confusion matrices for fine-grained attack subtype classification across the CICIoT2023, N-BaIoT and Bot-IoT datasets.

binary classification accuracy, reflecting a flawless separation between benign traffic and large-scale attacks. Similarly, CICIoT2023 achieves 99.5% weighted accuracy in coarse-grained classification, with DDoS, DoS, and Mirai categories showing diagonal dominance in confusion matrices.

These successes stem from the model's ability to exploit temporal and volumetric signals, eg, burst traffic patterns, high packet rates, that distinguish dominant attacks from normal behavior. The shared encoder further enhances generalization by propagating discriminative features across tasks, as evidenced by the ablation study: removing the shared encoder degraded category classification by 10.41 percentage points.

Minority classes (XSS, BruteForce, SQL injection) suffer from poor recall ($< 40\%$ in CICIoT2023), exacerbated by two interrelated factors. First, severe data imbalance plagues these categories, with minority classes such as XSS (427 samples in CICIoT2023) outnumbered by dominant attacks by $1-3$ orders of magnitude. Second, feature ambiguity arises from shared protocol fields (e.g., HTTP POST methods, standard ports 80/443) and negligible inter-packet gaps, creating overlaps that obscure distinctions between classes.

For example, CICIoT2023's Web-based and BruteForce classes are frequently misclassified due to indistinguishable header-level statistics, even though precision remains above 90%.

In N-BaIoT, TCP flag variants (e.g. ACK floods) exhibit 0% recall, revealing a lack of explicit flag-based features, e.g. ACK-ratio thresholds. Similarly, Bot-IoT's OS Fingerprinting class (3 621 samples) is misclassified as Service Scan due to aggregate statistics failing to capture TTL and window-size variations. These errors underscore the model's inability to isolate subtle protocol behaviors when critical discriminative features are absent from the input representation.

Flow-level metadata lacks critical granular cues (e.g., payload entropy, token sequences) for low-volume attacks. In CICIoT2023, SQL Injection, Command Injection, and Browser Hijacking are conflated due to shared HTTP methods and port usage, achieving only $19-43\%$ recall. This limitation highlights the inherent constraints of header-only analysis in distinguishing attacks that rely on nuanced payload content or application-layer logic.

The ablation study underscores the importance of key architectural components. The gating mechanism, which propagates binary-to-category features, improves F1 scores by 4 points, while attention contribute 3-point gains in multiclass tasks by enhancing feature adaptability.

In the 34-class CICIoT2023 classification, DNN outperforms RF by 10 percentage points in recall and 4 points in F1 score, demonstrating the advantage in mitigating data scarcity through shared representations.

Tab. 4. Coarse-grained classification performance on datasets.

Class/metrics	Precision	Recall	F1 score	Support
CICIoT2023				
Benign	92.00%	96.00%	94.00%	126 215
DDoS	100.00%	100.00%	100.00%	3 900 126
DoS	100.00%	100.00%	100.00%	928 707
Mirai	100.00%	100.00%	100.00%	302 220
Spoofing	87.00%	83.00%	85.00%	55 813
Recon	83.00%	81.00%	82.00%	40 963
Web-based	90.00%	40.00%	55.00%	2804
BruteForce	83.00%	24.00%	37.00%	1535
Accuracy		99.50%		
Macro avg	92.00%	78.00%	82.00%	5 358 383
Weighted avg	99.00%	99.00%	99.00%	5 358 383
EdgeIIoT				
Normal	100%	100%	100%	323 129
MITM	100%	100%	100%	243
Uploading	90%	68%	77%	7527
Ransomware	82%	90%	86%	2185
SQL_injection	80%	57%	67%	10 241
DDoS_HTTP	93%	90%	91%	9982
DDoS_TCP	99%	93%	96%	10 012
Password	52%	79%	63%	10 031
Port_Scanning	86%	94%	90%	4513
Vulnerability_scanner	100%	92%	96%	10 022
Backdoor	97%	96%	96%	4972
XSS	68%	85%	75%	3183
Fingerprinting	91%	55%	68%	200
DDoS_UDP	100%	100%	100%	24 314
DDoS_ICMP	100%	100%	100%	23 287
Accuracy		97.13%		
Macro avg	89%	87%	87%	443 841
Weighted avg	98%	97%	97%	443 841
Bot-IoT				
Normal	98.0%	92.5%	95.2%	107
DDoS	98.9%	99.3%	99.1%	385 309
DoS	99.2%	98.8%	99.0%	330 112
Reconnaissance	99.9%	99.1%	99.5%	18 163
Theft	100.0%	85.7%	92.3%	14
Macro avg	99.2%	95.1%	97.0%	733 705
Weighted avg	99.2%	97.1%	98.1%	1 467 410

7. Conclusions

Comprehensive experiments on four benchmark data sets demonstrate the robustness of the proposed model across multiple classification levels. Despite its strengths, HC-MTDNN encounters challenges with fine-grained detection of structurally similar or low-prevalence attacks, such as XSS and SQL injection. This low effectiveness is not merely a limitation in feature discriminability or class imbalance, but a direct consequence of the model's reliance on flow-level metadata, excluding payload analysis. This is a conscious design trade-off to maintain the lightweight nature, enabling deployment in resource-constrained IoT settings where full packet inspection may be infeasible due to encryption, privacy concerns, or computational overhead.

Future work will be focused on augmenting the feature space with lightweight payload-derived statistics (e.g., entropy metrics, token frequencies), temporal behavior modeling, and

Tab. 5. Coarse-grained classification performance.

Class/metrics	Precision	Recall	F1 score	Support
CICIoT2023				
Benign	92.00%	96.00%	94.00%	126 215
DDoS-ICMP_Flood	99.96%	99.96%	99.96%	826914
DDoS-UDP_Flood	99.85%	99.94%	99.90%	618833
DDoS-TCP_Flood	99.89%	99.92%	99.91%	516498
DDoS-PSHACK_Flood	99.98%	99.96%	99.97%	471782
DDoS-SYN_Flood	99.83%	99.90%	99.87%	466143
DDoS-RSTFINFlood	99.98%	99.93%	99.96%	463864
DDoS-SynonymIP_Flood	99.92%	99.87%	99.89%	412675
DDoS-ICMP_Fragmentation	99.53%	99.26%	99.40%	52443
DDoS-UDP_Fragmentation	99.01%	99.38%	99.20%	32820
DDoS-ACK_Fragmentation	98.97%	99.16%	99.07%	32211
DDoS-HTTP_Flood	98.16%	97.95%	98.05%	3216
DDoS-SlowLoris	87.02%	96.11%	91.34%	2727
DoS-UDP_Flood	99.90%	99.83%	99.87%	380875
DoS-TCP_Flood	99.97%	99.82%	99.90%	306346
DoS-SYN_Flood	99.84%	99.76%	99.80%	233180
DoS-HTTP_Flood	98.69%	98.03%	98.36%	8306
Recon-HostDiscovery	76.94%	83.96%	80.30%	15479
Recon-OSScan	62.74%	38.84%	47.98%	11304
Recon-PortScan	56.25%	59.77%	57.96%	9590
Recon-PingSweep	73.08%	7.09%	12.93%	268
VulnerabilityScan	94.47%	97.27%	95.85%	4322
MITM-ArpSpoofing	88.89%	81.47%	85.02%	35240
DNS_Spoofing	72.19%	69.28%	70.71%	20573
DictionaryBruteForce	68.48%	29.58%	41.31%	1535
BrowserHijacking	81.07%	20.33%	32.50%	674
XSS	29.71%	12.18%	17.28%	427
Uploading_Attack	0.00%	0.00%	0.00%	137
SqlInjection	55.56%	19.13%	28.46%	575
CommandInjection	53.55%	43.21%	47.83%	611
Backdoor_Malware	39.02%	25.26%	30.67%	380
Mirai-greeth_flood	99.00%	99.63%	99.31%	113958
Mirai-udpplain	99.94%	99.94%	99.94%	102242
Mirai-greip_flood	99.52%	98.64%	99.08%	86020
BenignTraffic	90.74%	97.01%	93.77%	126215
Accuracy	99.24%	99.24%	99.24%	
Macro avg	82.99%	75.63%	77.21%	5358383
Weighted avg	99.21%	99.24%	99.21%	5358383
Bot-IoT				
normal	99.0%	93.5%	96.2%	107
HTTP	98.6%	97.6%	98.1%	504
Keylogging	72.2%	92.9%	81.3%	14
OS_Fingerprint	76.9%	34.1%	47.3%	3621
Service_Scan	86.0%	97.0%	91.1%	14542
TCP	99.9%	100.0%	100.0%	318337
UDP	100.0%	100.0%	100.0%	396580
Macro avg	90.4%	87.9%	87.7%	733705
Weighted avg	95.0%	93.7%	93.6%	1467410
N-BaIoT				
Benign	100.00%	100.00%	100.00%	128410
Combo	100.00%	100.00%	100.00%	111186
Junk	100.00%	100.00%	100.00%	103311
Scan	100.00%	100.00%	100.00%	52146
TCP	100.00%	100.00%	100.00%	158466
UDP	100.00%	100.00%	100.00%	146498
ACK	100.00%	0.00%	0.00%	172261
SYN	72.00%	100.00%	83.00%	435529
udpplain	100.00%	100.00%	100.00%	104715
Accuracy		87.80%		
Macro avg	97.00%	89.00%	87.00%	1412522
Weighted avg	91.00%	88.00%	83.00%	1412522

Tab. 6. Ablation study results in the CICIoT2023 dataset, evaluating the contribution of key architectural components to the proposed model.

Variant	Binary		Category		Subcategory	
	F_1 [%]	AUC [%]	F_1 [%]	Δ [pts]	F_1 [%]	Δ [pts]
HC-MTDNN	99.84	99.95	81.53	—	77.21	—
No attention	99.77	99.92	78.48	–3.05	74.06	–3.15
No batch normalization	99.78	99.92	77.99	–3.54	74.00	–3.21
No gating	99.76	99.92	77.43	–4.10	73.72	–3.49
No gating, no attention	99.80	99.93	78.93	–2.60	73.86	–3.35
No shared encoder	99.72	99.86	71.12	–10.41	71.41	–5.80

sequence-aware components. These additions could improve classification fidelity without sacrificing real-time capability. Addressing encrypted traffic detection through enhanced metadata analysis and equipping the model with mechanisms for continuous learning and uncertainty estimation would further expand its applicability.

References

[1] R. Chataut, A. Phoummalayvane, and R. Akl, "Unleashing the Power of IoT: A Comprehensive Review of IoT Applications and Future Prospects in Healthcare, Agriculture, Smart Homes, Smart Cities, and Industry 4.0", *Sensors*, vol. 23, art. no. 7194, 2023 (https://doi.org/10.3390/s23167194).

[2] F. Nie, W. Liu, G. Liu, and B. Gao, "M2VT-IDS: A Multi-task Multi-view Learning Architecture for Designing IoT Intrusion Detection System", *Internet of Things*, vol. 25, art. no. 101102, 2024 (https://doi.org/10.1016/j.iot.2024.101102).

[3] P.S. Bangare and K.P. Patil, "Security Issues and Challenges in Internet of Things (IoT) System", *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Greater Noida, India, 2022 (https://doi.org/10.1109/ICACITE53722.2022.9823709).

[4] A.A. Rokhade *et al.*, "Anomaly Detection for IoT Security: Comprehensive Survey", *2023 International Conference on Advances in Electronics, Communication, Computing and Intelligent Information Systems (ICAECIS)*, Bangalore, India, 2023 (https://doi.org/10.1109/ICAECIS58353.2023.10170192).

[5] M. Antonakakis *et al.*, "Understanding the Mirai Botnet", *26th USENIX Conference on Security Symposium (SEC'17)*, Vancouver, Canada, 2017 (https://doi.org/10.13140/RG.2.2.24145.54885).

[6] H. Rhachi, Y. Balboul, and A. Bouayad, "Enhanced Anomaly Detection in IoT Networks Using Deep Autoencoders with Feature Selection Techniques", *Sensors*, vol. 25, art. no. 3150, 2025 (https://doi.org/10.3390/s25103150).

[7] A.G. Kumar, A. Rastogi, and V. Ranga, "Evaluation of Different Machine Learning Classifiers on New IoT Dataset CICIoT2023", *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*, Gurugram, India, 2024 (https://doi.org/10.1109/ISCS61804.2024.10581375).

[8] K.R. Narayan *et al.*, "IIDS: Design of Intelligent Intrusion Detection System for Internet-of-Things Applications", *2023 IEEE 7th Conference on Information and Communication Technology (CICT)*, Jabalpur, India, 2023 (https://doi.org/10.1109/CICT59886.2023.10455720).

[9] R. Caruana, "Multitask Learning", *Machine Learning*, vol. 28, pp. 41–75, 1997 (https://doi.org/10.1023/A:1007379606734).

[10] E.C.P. Neto *et al.*, "CICIOT2023: A Real-time Dataset and Benchmark for Large-scale Attacks in IoT Environment", *Sensors*, vol. 23, art. no. 5941, 2023 (https://doi.org/10.3390/s23135941).

[11] B. Meidan *et al.*, "N-BaIoT Dataset to Detect IoT Botnet Attacks", IEEE Pervasive Computing, vol. 17, pp. 12–22, 2018 (https://doi.org/10.1109/MPRV.2018.03367731).

[12] N. Koroniotis *et al.*, "Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset", *Future Generation Computer Systems*, vol. 100, pp. 779–796, 2018 (https://doi.org/10.1016/j.future.2019.05.041).

[13] M.A. Ferrag *et al.*, "Edge-IIoTSET: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning", *IEEE Access*, vol. 10, pp. 40281–40306, 2022 (https://doi.org/10.1109/ACCESS.2022.3165809).

[14] Imran *et al.*, "Realtime Feature Engineering for Anomaly Detection in IoT Based MQTT Networks", *IEEE Access*, vol. 12, pp. 25700–25718, 2024 (https://doi.org/10.1109/ACCESS.2024.3363889).

[15] Z. Hafezian, M. Naderan, and M. Jaderyan, "A Machine Learning-based Approach for Multi-class Intrusion Detection and Classification in IoT Using CICIoT2023 Dataset", *2024 11th International Symposium on Telecommunications (IST)*, Tehran, Iran, 2024 (https://doi.org/10.1109/IST64061.2024.10843502).

[16] A. Hajjouz and E. Avksentieva, "Optimizing Intrusion Detection for DoS, DDoS, and Mirai Attacks Subtypes Using Hierarchical Feature Selection and CatBoost on the CICIoT2023 Dataset", *Data and Metadata*, vol. 3, art. no. 577, 2024 (https://doi.org/10.56294/dm2024577).

[17] N. Thereza and K. Ramli, "Development of Intrusion Detection Models for IoT Networks Utilizing CICIoT2023 Dataset", *2023 3rd International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS)*, Bali, Indonesia, 2023 (https://doi.org/10.1109/ICON-SONICS59898.2023.10435006).

[18] C. Eunaicy, c. Jayapratha, and H.S. Hemachitra, "IoT Guardian: A Novel Feature Discovery and Cooperative Game Theory Empowered Feature Selection with ML Model for IoT Threats and Attack Detection", *International Journal of Computer Networks and Communications*, vol. 16, pp. 25–42, 2024 (https://doi.org/10.5121/ijcnc.2024.16202).

[19] J.R.K. Rajasekaran, B. Natarajan, and A. Pahwa, "Modified Matrix Completion-based Detection of Stealthy Data Manipulation Attacks in Low Observable Distribution Systems", *IEEE Transactions on Smart Grid*, vol. 14, pp. 4851–4862, 2023 (https://doi.org/10.1109/TSG.2023.3266834).

[20] H. Dong and I. Kotenko, "An Autoencoder-based Multi-task Learning for Intrusion Detection in IoT Networks", *2023 IEEE Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*, Yekaterinburg, Russia, 2023 (https://doi.org/10.1109/USBEREIT58508.2023.10158807).

[21] Z. Wang *et al.*, "A Lightweight Intrusion Detection Method for IoT Based on Deep Learning and Dynamic Quantization", *PeerJ Computer Science*, vol. 9, art. no. 1569, 2023 (https://doi.org/10.7717/peerj-cs.1569).

[22] J. Chen, J. Xiao, and J. Xu, "VGGIncepNet: Enhancing Network Intrusion Detection and Network Security Through Non-image-to-image Conversion and Deep Learning", *Electronics*, vol. 13, art. no. 3639, 2024 (https://doi.org/10.3390/electronics13183639).

[23] A. Hinojosa and N.E. Majd, "Edge Computing Network Intrusion Detection System in IoT Using Deep Learning", *2024 33rd International Conference on Computer Communications and Networks (ICCCN)*, Kailua-Kona, USA, 2024 (https://doi.org/10.1109/ICCCN61486.2024.10637611).

[24] T. Hasan and S. Tasnim, "Multidimensional Feature Learning Enhancement in IoT Intrusion Detection: An Adaptive Cost-sensitive Autoencoder and Weighted Ensemble Approach", *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*, Ottawa, Canada, 2024 (https://doi.org/10.1109/WF-IoT62078.2024.10811174).

Tab. 7. Baseline results for binary, 8-class, and 34-class classification.

Metric	Logistic regression	Perceptron	Adaboost	RF	DNN
	2-class (binary)				
Accuracy	0.9890	0.9818	0.9959	0.9968	0.9944
Recall	0.8904	0.7970	0.9473	0.9652	0.9333
Precision	0.8632	0.8254	0.9656	0.9654	0.9476
F1 score	0.8763	0.8105	0.9563	0.9653	0.9403
	8-class				
Accuracy	0.8317	0.8663	0.3514	0.9944	0.9911
Recall	0.6961	0.6591	0.4878	0.9100	0.9066
Precision	0.5124	0.5239	0.4649	0.7054	0.6794
F1 score	0.5394	0.5551	0.3687	0.7193	0.6973
	34-class				
Accuracy	0.8023	0.8196	0.6079	0.9916	0.9861
Recall	0.5952	0.5075	0.6077	0.8316	0.7319
Precision	0.4868	0.4546	0.4796	0.7045	0.6653

[25] S. Yan, H. Han, X. Dong, and Z. Xu, "Lightweight Deep Learning Method Based on Group Convolution: Detecting DDoS Attacks in IoT Environments", *2024 10th International Symposium on System Security, Safety, and Reliability (ISSSR)*, Xiamen, China, 2024 (https://doi.org/10.1109/ISSSR61934.2024.00027).

[26] M.A. Elaziz, I.A. Fares, and A.O. Aseeri, "CKAN: Convolutional Kolmogorov-Arnold Networks Model for Intrusion Detection in IoT Environment", *IEEE Access*, vol. 12, pp. 134837–134851, 2024 (https://doi.org/10.1109/ACCESS.2024.3462297).

[27] H. Kamal and M. Mashaly, "Enhanced Hybrid Deep Learning Models-based Anomaly Detection Method for Two-stage Binary and Multi-class Classification of Attacks in Intrusion Detection Systems", *Algorithms*, vol. 18, 2025 (https://doi.org/10.3390/a18020069).

[28] Z. Ahmad *et al.*, "Anomaly Detection Using Deep Neural Network for IoT Architecture", *Applied Sciences*, vol. 11, art. no. 7050, 2021 (https://doi.org/10.3390/app11157050).

[29] A.N. Gummadi, J.C. Napier, and M. Abdallah, "XAI-IoT: An Explainable AI Framework for Enhancing Anomaly Detection in IoT Systems", *IEEE Access*, vol. 12, pp. 71024–71054, 2024 (https://doi.org/10.1109/ACCESS.2024.3402446).

[30] 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, art. no. 713, 2024 (https://doi.org/10.3390/s24020713).

[31] Z. Zulfiqar *et al.*, "DeepDetect: An Innovative Hybrid Deep Learning Framework for Anomaly Detection in IoT Networks", *Journal of Computational Science*, vol. 83, art. no. 102426, 2024 (https://doi.org/10.1016/j.jocs.2024.102426).

[32] A. Alrefaei and M. Ilyas, "Using Machine Learning Multiclass Classification Technique to Detect IoT Attacks in Real Time", *Sensors*, vol. 24, art. no. 4516, 2024 (https://doi.org/10.3390/s24144516).

[33] Z. Alwaisi, T. Kumar, E. Harjula, and S. Soderi, "Securing Constrained IoT Systems: A Lightweight Machine Learning Approach for Anomaly Detection and Prevention", *Internet of Things*, vol. 28, art. no. 101398, 2024 (https://doi.org/10.1016/j.iot.2024.101398).

[34] I. Ullah and Q.H. Mahmoud, "Design and Development of RNN Anomaly Detection Model for IoT Networks", *IEEE Access*, vol. 10, pp. 62722–62750, 2022 (https://doi.org/10.1109/ACCESS.2022.3176317).

[35] M.M. Khan and M. Alkhathami, "Anomaly Detection in IoT-based Healthcare: Machine Learning for Enhanced Security", *Scientific Reports*, vol. 14, art. no. 5872, 2024 (https://doi.org/10.1038/s41598-024-56126-x).

[36] S.-M. Tseng, Y.-Q. Wang, and Y.-C. Wang, "Multi-class Intrusion Detection Based on Transformer for IoT Networks Using CIC-IoT-2023 Dataset", *Future Internet*, vol. 16, art. no. 284, 2024 (https://doi.org/10.3390/fi16080284).

[37] R. Zhao *et al.*, "A Novel Intrusion Detection Method Based on Lightweight Neural Network for Internet of Things", *IEEE Internet of Things Journal*, vol. 9, pp. 9960–9972, 2022 (https://doi.org/10.1109/JIOT.2021.3119055).

[38] M. Verkerken *et al.*, "A Novel Multi-stage Approach for Hierarchical Intrusion Detection", *IEEE Transactions on Network and Service Management*, vol. 20, pp. 3915–3929, 2023 (https://doi.org/10.1109/TNSM.2023.3259474).

[39] A. Momand, S.U. Jan, and N. Ramzan, "ABCNN-IDS: Attention-based Convolutional Neural Network for Intrusion Detection in IoT Networks", *Wireless Personal Communications*, vol. 136, pp. 1981–2003, 2024 (https://doi.org/10.1007/s11277-024-11260-7).

[40] S. Ali *et al.*, "Effective Multitask Deep Learning for IoT Malware Detection and Identification Using Behavioral Traffic Analysis", *IEEE Transactions on Network and Service Management*, pp. 1199–1209, 2022 (https://doi.org/10.1109/TNSM.2022.3200741).

Mohamed Amine Beghoura, Ph.D.
Department of Computer Science
https://orcid.org/0000-0002-8355-8071
E-mail: mohamedamine.beghoura@univ-bba.dz
University of Mohamed El Bachir El Ibrahimi, Bordj Bou Arreridj, Algeria
https://www.univ-bba.dz

Younes Belouche, Ph.D. student
Department of Computer Science
https://orcid.org/0009-0001-7809-2561
E-mail: younes.belouche@univ-bba.dz
University of Mohamed El Bachir El Ibrahimi, Bordj Bou Arreridj, Algeria
https://www.univ-bba.dz