Published in AJSE, Vol: 25, Issue:1
Received on 2nd July 2026
Revised on 17th August 2026
Published on 31st August 2026

AIUB JOURNAL OF SCIENCE AND ENGINEERING
ISSN: 1608 - 3679 (print) 2520 - 4890 (Online)
https://doi.org/10.53799/xjzz1h42

LAFR-IoT: A Lightweight AI-Driven Network Forensic Readiness Framework for Resource-Constrained IoT–Edge Devices

Md Manirul Islam, Md Mehedi Hasan Ratul, Md Tahsin Tasnim Aurin, and Sazzad Hossain

***Abstract*—The rapid expansion of Internet of Things (IoT) deployments has increased the volume of network evidence generated at the edge, yet most IoT devices remain constrained by limited processing power, memory, energy, and storage. These limitations make conventional full-packet logging and heavyweight intrusion detection impractical for proactive forensic readiness. This study proposes LAFR-IoT, a lightweight AI-driven network forensic readiness framework for resource-constrained IoT–Edge devices. The framework integrates lightweight machine learning, edge-aware model selection, explainability-based feature analysis, and selective forensic logging to detect, triage, and preserve high-value network events before evidence is lost. Experiments were conducted using the CICIoT2023 dataset, where network traffic was mapped into eight forensic categories: Benign, BruteForce, DDoS, DoS, Mirai, Recon, Spoofing, and Web. Eight lightweight models were evaluated. Although a shallow decision tree achieved the highest validation macro-F1, a linear support vector machine achieved the best overall Edge Forensic Score because of its low latency, small model size, and deployment efficiency. On the independent test set, the linear model achieved 0.7512 accuracy, 0.5741 macro-F1, 0.7615 weighted-F1, 0.0051 ms per sample inference latency, and a 0.0077 MB model size. The proposed selective logging mechanism reduced storage usage by 74.59% while preserving 98.96% of malicious events. These results show that lightweight edge AI can support proactive IoT forensic readiness by balancing detection performance, explainability, storage efficiency, and resource-aware deployment.**

***Index Terms*— CICIoT2023, edge AI, explainable AI, IoT forensics, lightweight machine learning, network forensic readiness, resource-constrained IoT, selective logging.**

I. Introduction

THE Internet of Things (IoT) has become a foundational technology for smart homes, healthcare systems, industrial automation, transportation, agriculture, environmental monitoring, and smart-city infrastructures. These environments depend on large numbers of interconnected sensors, actuators, gateways, and embedded devices that continuously exchange network traffic. While this connectivity improves automation and operational intelligence, it also expands the attack surface. IoT devices are frequently exposed to threats such as distributed denial-of-service attacks, botnet activity, reconnaissance, spoofing, brute-force attempts, and web-based exploitation. When such incidents occur, network traffic generated at or near the device can provide valuable forensic evidence for identifying attack behavior, reconstructing incidents, and supporting post-incident investigation.

However, forensic readiness in IoT environments remains challenging. Traditional digital forensics is often reactive, with evidence collection beginning only after an incident is detected. This approach is problematic for IoT systems because many devices have limited storage, volatile memory, restricted processing capability, and constrained energy budgets. As a result, important network evidence may be overwritten, discarded, or never collected. Full logging of all network events is also impractical because constrained edge devices cannot continuously store large volumes of traffic data. Cloud-based monitoring may reduce device-side workload, but it introduces dependency on network connectivity, increases latency, raises privacy concerns, and may not preserve evidence generated before transmission. IoT systems therefore require proactive forensic readiness mechanisms that can identify and preserve high-value evidence directly at or near the edge.

Machine learning has been widely explored for IoT intrusion detection, but most existing studies emphasize classification accuracy while paying limited attention to forensic readiness. A high-accuracy model may not be practical for constrained IoT devices if it requires large memory, long inference time, or high energy consumption. Similarly, a detection system that only produces attack labels may be insufficient for forensic investigation unless it can explain why an event was considered suspicious and determine whether the event should be preserved as evidence. This creates a gap between conventional IoT intrusion detection and practical IoT forensic readiness. A

M. M. Islam is with the Faculty of Science and Technology at American International University-Bangladesh, Dhaka 1229, Bangladesh and with the Faculty of Artificial Intelligence and Digital Technologies at Samarkand State University named after Sharof Rashidov, Samarkand 140104, Uzbekistan (e-mail: manirul@aiub.edu).

M. M. H. Ratul is with the Faculty of Science and Technology at American International University-Bangladesh, Dhaka 1229, Bangladesh (e-mail: 21-45007-2@student.aiub.edu).

M. T. T. Aurin is with the Institute of Continuing Education at American International University-Bangladesh, Dhaka 1229, Bangladesh (e-mail: tahsin.tasnim@aiub.edu).

S. Hossain is with the Faculty of Artificial Intelligence and Digital Technologies at Samarkand State University named after Sharof Rashidov, Samarkand 140104, Uzbekistan (e-mail: sazzad69@gmail.com).

suitable forensic readiness system should not only detect suspicious events but also triage them, explain the indicators that drove a decision, and selectively log events according to forensic value and resource constraints.

To address this gap, this study proposes LAFR-IoT, a lightweight AI-driven network forensic readiness framework for resource-constrained IoT–Edge devices. The framework classifies IoT network events, selects an edge-suitable model using a resource-aware scoring method, identifies important traffic features, and reduces log storage through AI-driven selective logging. Instead of storing all network events, the framework assigns different logging levels based on prediction outcome and confidence: high-confidence malicious events are preserved as full forensic logs, medium-risk events are stored as summary logs, uncertain events are stored minimally, and low-risk benign events are not stored in detail. This design preserves high-value forensic evidence while reducing unnecessary storage overhead.

The main contributions of this study are summarized as follows.

1) A lightweight AI-driven forensic readiness framework is proposed for IoT–Edge devices, integrating detection, triage, explainability, and selective evidence preservation.
2) A deployment-aware Edge Forensic Score is introduced for selecting models based on both classification performance and resource constraints.
3) Multiple lightweight models are evaluated jointly using classification metrics, inference latency, model size, and an energy proxy.
4) An AI-driven selective logging mechanism is proposed to reduce storage usage while preserving malicious evidence.
5) Explainability and feature-importance analysis are used to identify the network-traffic indicators that support forensic interpretation.

The remainder of this paper is organized as follows. Section II reviews related work on IoT intrusion detection, forensic readiness, edge AI, and explainability. Section III presents the proposed framework. Section IV describes the dataset, preprocessing, model selection, evaluation metrics, and experimental setup. Section V presents the results and analysis. Section VI discusses the implications of the findings. Section VII outlines limitations and future work, and Section VIII describes the publicly released reproducibility artifacts. Section IX concludes the paper.

II. Related Work

Recent research on IoT security has increasingly focused on machine-learning-based intrusion detection, edge deployment, explainability, and forensic readiness. The CICIoT2023 dataset introduced by Neto et al. [1] has become an important benchmark because it provides large-scale IoT traffic generated from real devices under diverse attacks, enabling evaluation under comparatively realistic network conditions. However, most CICIoT2023-based studies still emphasize classification accuracy rather than forensic evidence preservation, storage reduction, or resource-aware edge deployment.

IoT forensic readiness has been studied as a proactive alternative to post-incident investigation. Surveys by Stoyanova et al. [2], Atlam et al. [3], and Lutta et al. [4] identify recurring challenges, including device heterogeneity, limited storage, volatile evidence, privacy constraints, and the absence of standardized acquisition procedures. Ahmed et al. [5] further emphasize the need for dependable evidence identification, preservation, and analysis under constrained processing and connectivity, while Friedl and Pernul [6] analyze the factors that shape practical IoT forensic readiness. Rizvi et al. [7] move readiness toward IoT and edge devices through a resource-constrained AI methodology, and their broader network-forensics survey [8] outlines the opportunities and unresolved challenges associated with AI-assisted evidence analysis. Nevertheless, prior work rarely evaluates detection quality, explainability, latency, model size, energy-related cost, and selective storage reduction within one framework.

Lightweight intrusion detection has also received significant attention. He et al. [9] proposed feature-grouping-based lightweight detection, while Fatima et al. [10], [11] showed that ensemble feature selection can reduce resource overhead in constrained IoT environments. Li et al. [12] compared feature-reduction techniques, and Misrak and Melaku [13] combined improved feature engineering with dynamic quantization. Complementary device- and gateway-oriented studies include Realguard [14], a lightweight supervised mechanism for IoT networks [15], an intrusion-detection framework for energy-constrained devices [16], and an SGD-classifier approach based on efficient feature subsets [17]. These studies demonstrate that compact feature sets and low-complexity learners can improve deployability, but most do not connect detection decisions to proactive forensic logging and evidence preservation.

Several works have applied deep and hybrid learning to IoT intrusion detection. Yaras and Dener [18] proposed a hybrid deep-learning IDS; Ebrahimi et al. [19] combined convolutional learning with explainable AI; Gheni and Al-Yaseen [20] applied two-step clustering on CICIoT2023; and Mahadevappa et al. [21] proposed a secure edge-computing model for detecting and isolating intruders. Other influential systems include N-BaIoT for botnet detection with deep autoencoders [22], a distributed deep-learning attack-detection scheme [23], and an anomaly-based deep IoT IDS [24]. These contributions establish the value of nonlinear and distributed detection, but their comparatively heavier models and detection-centric objectives do not directly solve selective forensic evidence preservation on constrained edge devices.

Explainability has become another important direction. Sharma et al. [25] applied explainable deep learning to IoT intrusion detection; Alabbadi and Bajaber [26] studied explainable detection over IoT data streams; Bin Hulayyil et al. [27] developed an explainable IoT intrusion-detection framework; and Arreche et al. [28] proposed XAI-IDS to improve transparency. General-purpose methods such as LIME [29] and SHAP-based tree explanations [30] provide established mechanisms for identifying features that drive model outputs. These studies show that explanations can improve analyst trust and interpretation, but explainability is

still rarely linked to selective forensic logging, where explanations can help determine which evidence should be preserved.

Overall, the literature shows progress in IoT intrusion detection, lightweight design, edge security, explainability, and forensic readiness, but these areas are typically addressed separately. This study bridges that gap by integrating resource-aware model selection, explainability-supported triage, and AI-driven selective logging within a unified edge-oriented forensic readiness framework for practical deployment on constrained devices.

III. Proposed Framework

This study proposes LAFR-IoT, a lightweight AI-driven network forensic readiness framework for resource-constrained IoT–Edge devices. It shifts IoT forensics from reactive post-incident investigation to proactive edge-level evidence preparation through local event classification, risk-aware triage, explainability-supported interpretation, and selective forensic logging. Its objective is to preserve high-value evidence while reducing storage, latency, and computational overhead. The framework comprises six layers: network event acquisition; preprocessing and feature preparation; lightweight machine-learning detection; explainability and triage; AI-driven selective logging; and forensic readiness storage. These layers operate collaboratively to prioritize suspicious events, retain contextually relevant artefacts, and minimize unnecessary data collection. The resulting evidence repository supports faster incident reconstruction while maintaining suitability for deployment on constrained edge platforms. The overall architecture is shown in Fig. 1.

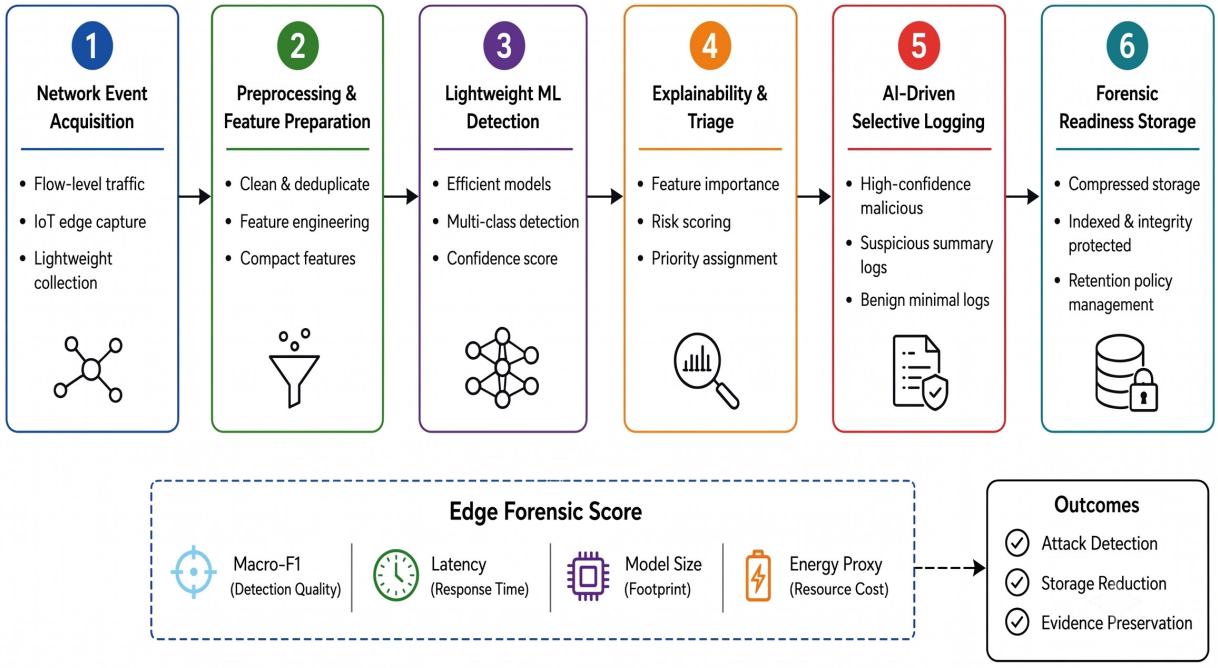

Fig. 1. Proposed LAFR-IoT framework for lightweight AI-driven network forensic readiness.

The network event acquisition layer collects flow-level IoT traffic information rather than full packet payloads, which reduces privacy risk and storage overhead. Typical input features include flow duration, header length, inter-arrival time, protocol type, statistical flow indicators, and flag counts. These features represent network behavior without requiring continuous full-packet capture, making them suitable for constrained devices.

The preprocessing and feature-preparation layer handles invalid values, removes non-informative features, converts features into numerical form, and maps raw attack labels into broader forensic categories. Its purpose is to produce a compact feature representation that supports fast inference and low memory consumption.

The lightweight machine-learning detection layer classifies network events into forensic categories using compact models, including a shallow decision tree, logistic regression, ridge classifier, stochastic gradient descent (SGD) classifier, Gaussian naive Bayes, a linear support vector machine (SVM), a tiny gradient-boosting model, and an extra-tree classifier. These models were selected because they offer lower computational cost and smaller memory footprints than large neural architectures.

The explainability and triage layer addresses the fact that, in forensic applications, a prediction label alone is insufficient; investigators must understand which traffic features contributed to a suspicious classification. Feature importance and permutation importance are therefore used to identify influential indicators, allowing the framework to preserve compact but meaningful evidence such as the predicted class, confidence value, important traffic features, and event priority.

The AI-driven selective logging layer is the core forensic-readiness component. Instead of logging every event with equal detail, the framework assigns logging levels according to prediction output and confidence: high-confidence

malicious events are stored as full forensic logs, medium-confidence suspicious events as summary logs, uncertain events as minimal logs, and high-confidence benign events without detailed logging. This policy is summarized in Table I.

TABLE I
SELECTIVE FORENSIC LOGGING POLICY

Event Condition	Logging Decision	Forensic Purpose
Predicted non-benign, confidence ≥ 0.90	Full Forensic Log	Preserve complete event metadata and explanation
Predicted non-benign, 0.60 ≤ confidence < 0.90	Summary Forensic Log	Preserve compact evidence for later review
Any prediction, confidence < 0.60	Minimal Log	Preserve a low-cost uncertainty trace
Predicted benign, 0.60 ≤ confidence < 0.90	Minimal Log	Retain an ambiguous benign trace for review
Predicted benign, confidence ≥ 0.90	No Detailed Log	Reduce unnecessary storage consumption

Finally, the forensic readiness storage layer stores selected evidence with an event identifier, timestamp, predicted label, confidence score, important features, logging level, and source information. This storage is compact, structured, and suitable for later incident reconstruction.

To support deployment-aware model selection, the framework introduces an Edge Forensic Score that jointly captures detection quality and resource cost. For each candidate model, an energy proxy is first defined as the product of inference latency ℓ and serialized model size s:

$$E_p = \ell \times s \quad (1)$$

The Edge Forensic Score is then computed as a weighted combination of the normalized macro-F1 score and the normalized, inverted resource terms:

$$\mathrm{EFS} = w_1 F1' + w_2(1 - \ell') + w_3(1 - s') + w_4(1 - E_p') \quad (2)$$

where the prime notation denotes min–max normalization across all evaluated models, and the weights w_1–w_4 are non-negative and sum to one. Equal weighting is used by default so that detection quality and the three resource terms—latency, model size, and energy proxy—contribute comparably. A higher macro-F1 increases the score, whereas lower latency, smaller model size, and lower energy proxy also increase it. This reflects the practical requirement that the most suitable edge model is not necessarily the most accurate one, but the one offering the best trade-off between detection capability and deployment cost.

IV. METHODOLOGY

A. Dataset and Preprocessing

The experimental evaluation was conducted using the CICIoT2023 dataset, which contains IoT network traffic representing benign and malicious behaviors. The original labels were mapped into eight broader forensic categories: Benign, BruteForce, DDoS, DoS, Mirai, Recon, Spoofing, and Web, so that related attack behaviors are grouped into interpretable classes that support event-level triage. The final experimental dataset contained 125,468 samples described by 47 columns, with forensic_category as the target. To keep the experiment feasible under resource-aware conditions, a controlled sampling strategy was applied, drawing 8,000 samples per source file with a maximum of 25,000 samples per class.

The preprocessing pipeline was designed for lightweight edge deployment. Dataset files were loaded and combined using a memory-safe procedure; invalid and infinite values were handled; duplicate records were removed; and samples with missing targets were excluded. Because the target environment is resource-constrained edge hardware, only numeric flow-level features were retained, and constant or non-informative columns were removed to reduce dimensionality. Feature values were stored in a memory-efficient numerical format to limit RAM usage during training and inference, reflecting the goal of operating on compact traffic summaries rather than full packet payloads. The data were divided using a stratified 70:15:15 split, producing 87,827 training, 18,820 validation, and 18,821 test samples; the validation set was used for model comparison and edge-aware selection, while the test set was reserved for final evaluation.

B. Lightweight Models

Eight lightweight models were evaluated: a shallow decision tree, logistic regression, a ridge classifier, an SGD classifier, Gaussian naive Bayes, a linear SVM, a tiny gradient-boosting model, and an extra-tree classifier. They were chosen because they are suitable for resource-constrained environments and span linear, probabilistic, tree-based, and compact boosting strategies. The shallow decision tree and extra tree provide interpretable structures and fast inference; the linear models offer compact baselines with low memory requirements; Gaussian naive Bayes provides a simple probabilistic baseline; and the tiny boosting model offers stronger non-linear learning while controlling complexity through shallow depth and a limited number of estimators. This diverse model set enables a balanced comparison of predictive performance, computational efficiency, memory demand, and deployment suitability at the IoT–Edge.

For reproducibility, the model configurations are reported in Table II. All models were trained on identical preprocessed inputs, and a fixed random seed of 42 was applied to stochastic components. The shallow decision tree was restricted to a maximum depth of eight, which defines the "shallow" designation used throughout this paper, and the tiny gradient-boosting model was limited to fifty estimators of depth three to bound its memory footprint. All parameters not listed in Table II retained their implementation defaults; the released artifact records the exact estimator classes, package versions, serialization procedure, and software environment used in the experiments.

TABLE II
HYPERPARAMETER CONFIGURATIONS OF THE EVALUATED MODELS

Model	Hyperparameter Configuration
Shallow Decision Tree	Maximum depth = 8; Gini impurity criterion; random seed = 42
Logistic Regression	L2 penalty; C = 1.0; lbfgs solver; maximum 1,000 iterations
Ridge Classifier	L2 regularization strength α = 1.0
SGD Classifier	Hinge loss; L2 penalty; α = 0.0001; maximum 1,000 iterations; random seed = 42
Gaussian Naive Bayes	Default configuration (no tuned hyperparameters)
Linear SVM	Linear formulation; L2 penalty; C = 1.0; squared-hinge loss; maximum 1,000 iterations; random seed = 42
Tiny Gradient Boosting	50 estimators; maximum depth = 3; learning rate = 0.1; random seed = 42
Extra Tree	Single randomized tree; maximum depth = 8; Gini impurity criterion; random seed = 42

C. Evaluation Metrics

Three groups of metrics were used. Classification quality was measured with accuracy, macro-precision, macro-recall, macro-F1, weighted-F1, the confusion matrix, ROC curves, and precision–recall curves; macro-F1 was emphasized because the forensic categories are imbalanced and minority attack classes are important in investigation. Resource efficiency was measured with training time, inference latency in milliseconds per sample, model size in megabytes, and an energy proxy. Because direct hardware power measurement was not available, the energy proxy was computed as the product of inference latency and model size; although it does not replace physical power profiling, it provides a useful comparative estimate. Forensic readiness was measured with storage reduction, comparing full logging against selective logging, and with the evidence preservation rate, the proportion of malicious events retained by the logging mechanism.

D. Edge Forensic Score and Model Selection

The Edge Forensic Score defined in (1) and (2) was used to select the final model. Macro-F1 represents detection quality, while latency, model size, and energy proxy represent deployment cost. By combining normalized detection and resource terms, the score identifies the model that is simultaneously effective and deployable on constrained edge devices, rather than the model with the highest accuracy alone.

E. Selective Forensic Logging Simulation

The selected model was used to simulate selective forensic logging on the independent test set. Prediction confidence was derived from normalized decision scores produced by the selected linear SVM. The logging policy was applied deterministically as follows: a predicted non-benign event with confidence at least 0.90 received a full forensic log; a predicted non-benign event with confidence from 0.60 to below 0.90 received a summary forensic log; any event with confidence below 0.60 received a minimal uncertainty log; a predicted benign event with confidence from 0.60 to below 0.90 also received a minimal log; and a predicted benign event with confidence at least 0.90 received no detailed log. The 0.90 high-confidence and 0.60 low-confidence thresholds were calibrated on the validation set only by comparing candidate threshold pairs with respect to the storage-reduction/evidence-preservation trade-off, and they were fixed before the test set was examined. For storage accounting, the full-logging baseline assigned 5 KB to every event, whereas the selective total was calculated from the event-level logging category and storage cost recorded by the released simulation. The evidence-preservation rate was computed from malicious test events assigned to any nonzero logging level.

F. Experimental Setup

The experiments were implemented in Python using scikit-learn and the gradient-boosting implementation recorded in the released environment file, and were executed in a cloud notebook environment. All models were evaluated using the same preprocessing pipeline, data partitions, and execution conditions to ensure a consistent comparison. All outputs were saved as tables and figures, including validation performance, final test performance, resource consumption, feature importance, storage reduction, forensic readiness metrics, feature ablation, and confidence intervals. The setup was designed to be reproducible and resource-aware; the exact package versions, estimator classes, timing procedure, serialization method, and environment specification are included in the artifacts described in Section VIII.

V. RESULTS AND ANALYSIS

A. Dataset and Class Distribution

The class distribution of the CICIoT2023 forensic categories is shown in Fig. 2. Mirai, DoS, DDoS, and Benign are the dominant categories, each approaching the class cap of 25,000 samples, while Spoofing and Recon have moderate representation and Web and BruteForce are substantially smaller. This imbalance is important because accuracy alone may overestimate performance; macro-F1, macro-recall, weighted-F1, and class-wise results are therefore reported throughout. The stratified data split preserved these relative class proportions across the training, validation, and test sets. The limited representation of Web and BruteForce also makes their class-specific results particularly sensitive to individual prediction errors.

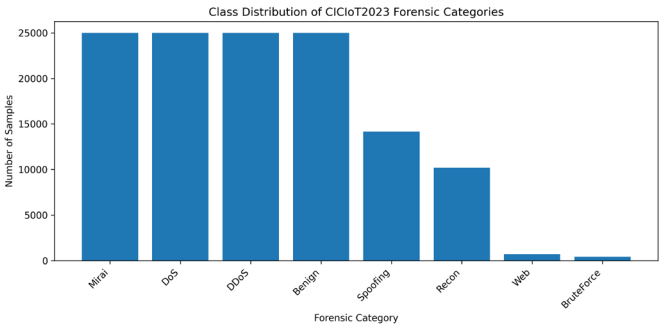

Fig. 2. Class distribution of CICIoT2023 forensic categories.

B. Model Performance Comparison

The validation performance of all models is summarized in Table III. The shallow decision tree achieved the highest

validation macro-F1 of 0.8464, with an accuracy of 0.9332 and weighted-F1 of 0.9319, and the tiny boosting model ranked second with a macro-F1 of 0.7311. The linear SVM achieved a lower validation macro-F1 of 0.5754 but offered stronger deployment efficiency through its low latency and small model size.

TABLE III
VALIDATION PERFORMANCE OF LIGHTWEIGHT MODELS

Model	Accuracy	Macro P	Macro R	Macro F1	Weighted F1
Shallow Decision Tree	0.9332	0.9506	0.7960	0.8464	0.9319
Tiny Gradient Boosting	0.9131	0.7415	0.8475	0.7311	0.9307
Linear SVM	0.7525	0.5895	0.6234	0.5754	0.7627
Logistic Regression	0.7177	0.5900	0.6235	0.5536	0.7438
SGD Classifier	0.7273	0.5713	0.5836	0.5487	0.7417
Ridge Classifier	0.7060	0.5754	0.5988	0.5365	0.7140
Extra Tree	0.7091	0.6119	0.5059	0.5011	0.6868
Gaussian NB	0.5718	0.6451	0.5107	0.4302	0.5980

These results show that the shallow decision tree is the strongest accuracy-oriented validation model. However, the aim of this work is not solely to maximize predictive performance; because the system targets resource-constrained IoT devices, the final model must also be small, fast, and energy-efficient. Therefore, model selection must reflect a balanced trade-off between detection quality and practical deployment cost.

C. *Edge-Aware Model Selection*

To select the final model, the Edge Forensic Score was applied. The score jointly considers predictive performance, inference latency, model size, and the energy proxy to identify the most deployment-suitable model. The resulting ranking is shown in Fig. 3 and reported in Table IV.

TABLE IV
EDGE FORENSIC SCORE RANKING

Rank	Model	Macro F1	Latency	Model Size	Energy Proxy	EFS
1	Linear SVM	0.5754	0.0051	0.0077	0.000039	0.8148
2	SGD Classifier	0.5487	0.0050	0.0067	0.000033	0.8012
3	Extra Tree	0.5011	0.0037	0.0316	0.000118	0.7725
4	Shallow Decision Tree	0.8464	0.0201	0.0221	0.000443	0.7203
5	Logistic Regression	0.5536	0.0108	0.0065	0.000070	0.7141
6	Ridge Classifier	0.5365	0.0108	0.0065	0.000070	0.7038
7	Gaussian NB	0.4302	0.0082	0.0103	0.000084	0.6765
8	Tiny Gradient Boosting	0.7311	0.0130	0.3992	0.005208	0.2890

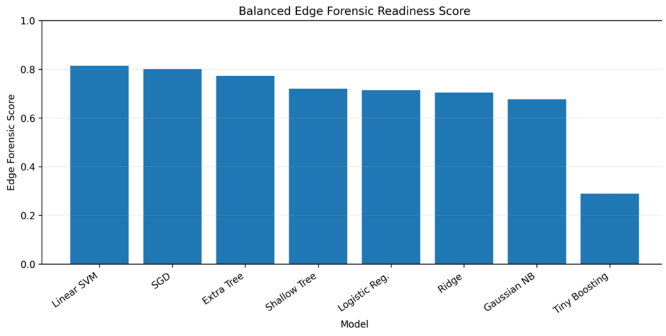

Fig. 3. Balanced edge forensic readiness score across evaluated models.

Although the shallow decision tree achieved the highest validation macro-F1, the linear SVM achieved the highest Edge Forensic Score of 0.8148 because it combined a very small model size of 0.0077 MB, low latency of 0.0051 ms per sample, and a low energy proxy of 0.000039. The resulting ranking places the SGD classifier second, the extra tree third, and the shallow decision tree fourth. This distinction is important: the shallow decision tree is the strongest accuracy-oriented model, whereas the linear SVM is the strongest deployment-oriented model. Since the goal is lightweight forensic readiness at the IoT–Edge, the linear SVM was selected as the final model.

D. *Final Test Performance and Class-Wise Analysis*

The selected linear SVM was evaluated on the independent test set, where it achieved an accuracy of 0.7512, macro-precision of 0.5899, macro-recall of 0.6208, macro-F1 of 0.5741, and weighted-F1 of 0.7615. The weighted-F1 exceeding the macro-F1 indicates that the model performs better on high-support classes than on low-support classes. The class-wise report is given in Table V.

TABLE V
CLASS-WISE CLASSIFICATION REPORT (LINEAR SVM)

Class	P	R	F1	Support
Benign	0.7851	0.7669	0.7759	3,750
BruteForce	0.0475	0.4098	0.0852	61
DDoS	0.8347	0.6128	0.7068	3,750
DoS	0.6887	0.8667	0.7675	3,750
Mirai	0.9925	0.9909	0.9917	3,750
Recon	0.6002	0.5232	0.5591	1,529
Spoofing	0.7136	0.5390	0.6141	2,126
Web	0.0566	0.2571	0.0928	105

The strongest class-level performance was achieved for Mirai, with an F1-score of 0.9917, and Benign, DoS, and DDoS also performed reasonably. The weakest classes were BruteForce and Web, with F1-scores of 0.0852 and 0.0928, mainly due to very small test supports of 61 and 105 samples, respectively.

E. Confusion Matrix Analysis

The confusion matrix of the selected model is shown in Fig. 4. Mirai was classified with very high reliability, with 3,716 correct predictions out of 3,750 samples, corresponding to a normalized recall of 0.99. DoS achieved 3,250 correct predictions and a normalized recall of 0.87, while Benign traffic achieved 2,876 correct predictions and a normalized recall of 0.77.

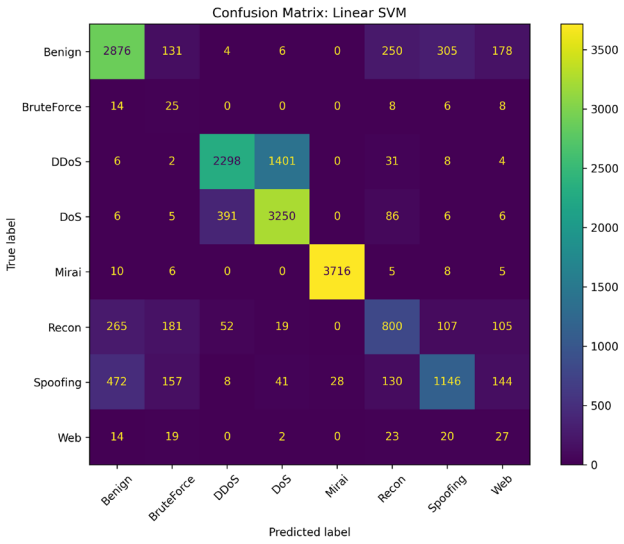

Fig. 4. Confusion matrix of the selected Linear SVM model.

The matrix also shows that DDoS and DoS are frequently confused: 1,401 DDoS samples were predicted as DoS, and 391 DoS samples were predicted as DDoS. This is understandable because DDoS and DoS traffic share similar high-volume flow characteristics. From a forensic perspective, this error is less harmful than misclassifying an attack as benign, because the event remains within an attack-related category and can still be preserved.

F. ROC and Precision–Recall Analysis

The ROC curves in Fig. 5 show strong separability for most classes. Mirai achieved the highest AUC of 0.997, followed by Benign with 0.963, DDoS with 0.957, BruteForce with 0.950, DoS with 0.943, Recon with 0.927, Web with 0.913, and Spoofing with 0.899. The comparatively lower AUC for Spoofing suggests greater overlap with other traffic categories and indicates that this class remains more difficult to isolate reliably. Nevertheless, all classes achieved AUC values close to or above 0.90, confirming the model's generally robust discriminatory capability. These results indicate that the selected model can effectively distinguish each class from the remaining traffic, although the precision–recall curves provide a more conservative assessment for the minority classes.

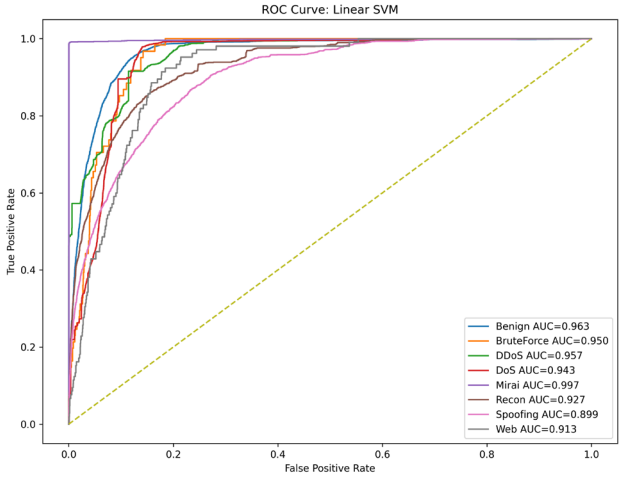

Fig. 5. ROC curves of the Linear SVM across forensic categories.

The precision–recall curves in Fig. 6 provide a more realistic view under class imbalance. Mirai achieved the highest average precision (AP) of 0.994, followed by DDoS with 0.869, Benign with 0.848, and DoS with 0.742. Recon and Spoofing reached moderate AP values of 0.599 and 0.586, whereas BruteForce and Web achieved low AP values of 0.149 and 0.048. The contrast between high ROC-AUC and low minority-class AP is important: ROC-AUC can remain high even when minority-class precision is weak, while precision–recall curves reveal the effect of class imbalance more clearly.

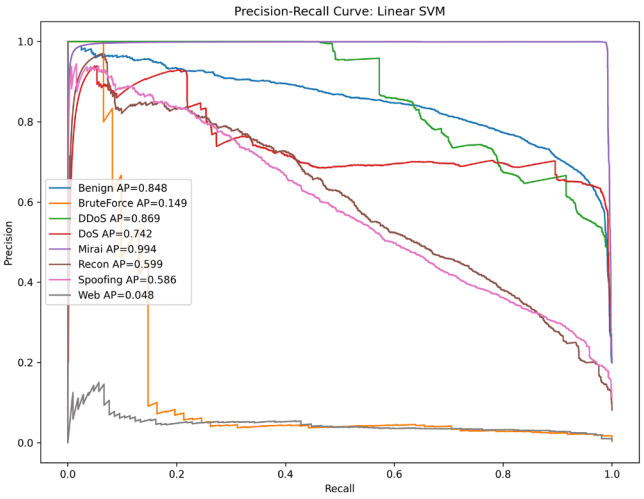

Fig. 6. Precision–recall curves of the Linear SVM across forensic categories.

G. Resource Consumption and Edge Deployment

Resource consumption is central to edge deployment. Among all models, the linear SVM required a longer offline training time of roughly 257 s, but training time does not affect real-time edge operation; the decisive deployment metrics are model size and inference latency. The linear SVM achieved a model size of only 0.0077 MB and a latency of 0.0051 ms per sample, making it practical for low-resource devices. By comparison, the tiny gradient-boosting model, although

competitive in macro-F1, required a far larger 0.3992 MB footprint and a higher energy proxy of 0.005208, which lowered its Edge Forensic Score and made it less suitable for constrained hardware.

H. Explainability and Feature Importance

Explainability is essential for forensic readiness because investigators must understand why an event was classified as suspicious. Built-in feature importance and permutation importance were used to identify influential traffic indicators, and the fifteen most important features are shown in Fig. 7.

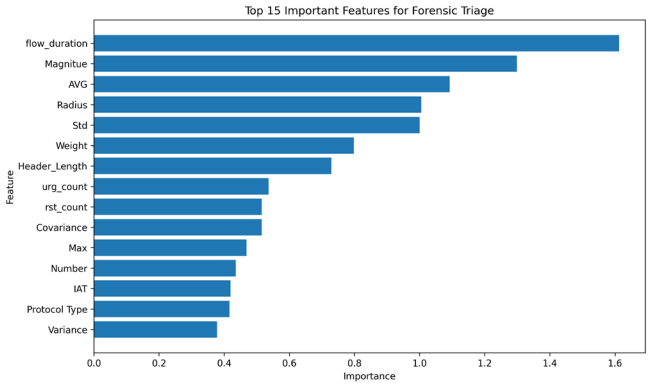

Fig. 7. Top-15 feature importance for forensic triage.

The built-in importance ranking was led by flow duration (1.6123), followed by a statistical magnitude feature (1.2993), an average flow feature (1.0923), a radius feature (1.0052), and a standard-deviation feature (1.0007). The permutation analysis produced a consistent ordering, with the magnitude feature (0.2415), the average feature (0.1451), a weight feature (0.1271), the standard-deviation feature (0.1133), and the reset-flag count (0.1049) ranked highest. These results indicate that flow duration, magnitude-based statistics, timing indicators, header length, reset counts, and variance-based behavior are the most informative indicators for triage, and they can be stored as compact forensic evidence alongside prediction labels and confidence scores.

I. Selective Forensic Logging and Storage Reduction

The framework applies AI-driven selective logging across four categories: full forensic log, summary forensic log, uncertain minimal log, and no detailed log. On the test set, most events were stored as minimal logs, followed by summary and full forensic logs. This distribution demonstrates that the framework prioritizes evidence according to event risk and uncertainty rather than retaining complete records indiscriminately. Consequently, high-value forensic information is preserved while redundant and low-priority data are substantially reduced. The resulting storage comparison is shown in Fig. 8 and summarized in Table VI.

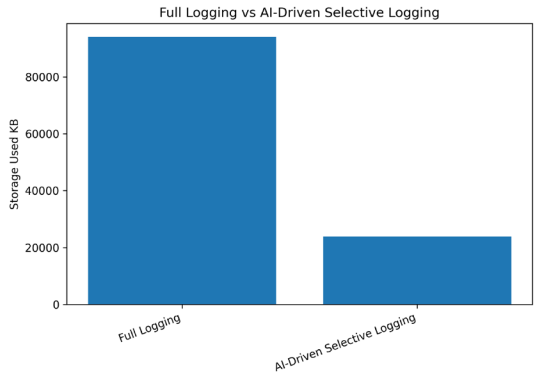

Fig. 8. Full logging versus AI-driven selective logging.

TABLE VI
FORENSIC READINESS AND STORAGE-REDUCTION RESULTS

Metric	Value
Full-logging storage	94,105.0 KB
Full-logging record size	5 KB/event
Selective-logging storage	23,911.3 KB
Storage reduction	74.59%
Total test events	18,821
Total malicious events	15,071
Logged malicious events	14,915
Evidence preservation rate	98.96%

Full logging would require 94,105.0 KB for the 18,821 test events because the baseline assigns 5 KB to every event. The proposed selective policy required 23,911.3 KB, a reduction of 74.59%. Of the 15,071 malicious events, the event-level simulation preserved 14,915, yielding an aggregate evidence-preservation rate of 98.96%. Preservation is not identical to class recall because a malicious event misclassified into another non-benign class remains in the evidence pool. Table VII therefore reports conservative class-wise lower bounds that can be reproduced directly from the confusion matrix: every prediction into a non-benign category is guaranteed to be logged, while additional low-confidence benign predictions may also be retained as minimal logs. Consequently, the actual class-specific preservation rates can only equal or exceed these bounds.

TABLE VII
CLASS-WISE CONSERVATIVE EVIDENCE-PRESERVATION BOUNDS

Class	Attack Events	Conservative Bound
BruteForce	61	≥ 77.05%
DDoS	3,750	≥ 99.84%
DoS	3,750	≥ 99.84%
Mirai	3,750	≥ 99.73%
Recon	1,529	≥ 82.67%
Spoofing	2,126	≥ 77.80%
Web	105	≥ 86.67%

Across all attack classes, the confusion-matrix-derived lower bound is 94.78% (14,284 of 15,071 malicious events). The higher observed aggregate rate of 98.96% arises because the confidence-aware policy additionally retains uncertain events, including some malicious flows predicted as benign with confidence below 0.90.

J. Feature Ablation and Confidence Intervals

A feature-ablation study examined whether a smaller feature set can maintain strong performance. The ablation was conducted with the shallow decision tree—the strongest accuracy-oriented model in Table III—so that the effect of feature-set size could be isolated from model capacity; the macro-F1 values reported in this subsection therefore correspond to that model rather than to the selected linear SVM, which is why they differ from the headline result in Section V-D. Using only the five top-ranked features yielded a macro-F1 of 0.5106; increasing to ten features sharply improved macro-F1 to 0.8178; and the best macro-F1 of 0.8367 was obtained with thirty features, while inference latency fell to 0.0060 ms per sample, compared with 0.0201 ms per sample for the full feature set. This indicates that a carefully selected reduced feature set can preserve strong detection performance while improving edge efficiency. Bootstrap confidence intervals were also computed to assess stability: accuracy had a mean of 0.7510 with a 95% interval of [0.7449, 0.7576], and macro-F1 had a mean of 0.5743 with a 95% interval of [0.5667, 0.5824]. The narrow intervals indicate that the final results are stable across bootstrap samples, although the macro-F1 interval confirms that minority-class performance remains moderate and should be improved in future work.

VI. Discussion

The experimental findings demonstrate that lightweight machine learning can support proactive network forensic readiness in resource-constrained IoT–Edge environments. A key observation is that the best model depends on the optimization objective. If the objective is classification performance alone, the shallow decision tree is the strongest validation model; however, when deployment constraints such as latency, model size, and energy proxy are considered, the linear SVM becomes the most suitable choice. This distinction matters for edge systems, where computational resources are limited and continuous monitoring must proceed with minimal overhead.

The selected linear SVM achieved a compact deployment profile, with a model size of 0.0077 MB and an inference latency of 0.0051 ms per sample, suggesting that lightweight linear models can support near-real-time forensic triage on constrained devices. Nevertheless, its final macro-F1 of 0.5741 shows that resource efficiency alone is not sufficient: minority-class detection remains challenging. The class-wise results show strong detection for Mirai, Benign, DoS, and DDoS, whereas BruteForce and Web were difficult owing to limited sample support and overlapping network behavior.

The most significant contribution of the framework is its selective forensic logging capability, which reduced storage by 74.59% while preserving 98.96% of malicious events. The framework thus reduces unnecessary log storage without losing most high-value evidence. Unlike conventional intrusion detection systems that only classify traffic, the proposed framework detects, explains, prioritizes, and stores events according to forensic value, providing a practical path toward proactive IoT forensic readiness at the edge.

VII. Limitations and Future Work

Although the framework produced promising results, several limitations remain. First, the study used an energy proxy based on latency and model size rather than direct hardware power measurement; future work should validate the framework on real edge devices, such as Raspberry Pi, Jetson Nano, or ESP32-class gateways, using direct power profiling. Second, the dataset exhibited strong class imbalance, and minority classes such as BruteForce and Web achieved weak F1-scores; future research should apply improved class balancing, targeted augmentation, cost-sensitive learning, and class-specific threshold tuning to improve rare-attack detection.

Third, the current study used a sampled version of CICIoT2023 for resource-aware experimentation; future work should test the framework on full-scale data and evaluate cross-dataset generalization using Bot-IoT [31], TON_IoT [32], and Edge-IIoTset [33]. Fourth, the logging mechanism focuses on storage efficiency but does not yet provide tamper-evident hashing, secure time synchronization, evidence-chain metadata, or legal chain-of-custody validation. Future work should integrate cryptographic integrity protection and test end-to-end evidential admissibility. Finally, the validation-calibrated confidence thresholds should be re-evaluated under concept drift, changing traffic distributions, and adversarial manipulation.

VIII. Reproducibility and Artifact Availability

To facilitate independent verification, scrutiny, and extension of this work, the complete experimental pipeline has been publicly released. The artifact comprises preprocessing and forensic-category mapping scripts, training and evaluation code for all eight lightweight models, the exact Edge Forensic Score calculation, the event-level selective-logging simulation, environment and dependency information, and the scripts used to generate every table and figure reported in this paper. Fixed random seeds and documented software settings allow the stratified data splits, model training, edge-aware selection, and forensic-readiness results to be regenerated end-to-end from the publicly available CICIoT2023 dataset. The repository is now publicly and permanently archived on Zenodo under https://doi.org/10.5281/zenodo.21414239.

IX. Conclusion

This study proposed LAFR-IoT, a lightweight AI-driven network forensic readiness framework for resource-constrained IoT–Edge devices, integrating lightweight machine learning, edge-aware model selection, explainability, and selective forensic logging. On CICIoT2023, a linear SVM achieved the best Edge Forensic Score, with 0.7512 accuracy, 0.5741 macro-F1, 0.0051 ms per sample latency, and a 0.0077 MB model size. More importantly, the proposed selective logging mechanism reduced storage usage by 74.59% while preserving 98.96% of

malicious events. These findings demonstrate that lightweight AI can support proactive forensic evidence preservation at the IoT–Edge while minimizing computational and storage overhead.

REFERENCES

[1] 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, Art. no. 5941, 2023, doi: 10.3390/s23135941.

[2] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, "A survey on the Internet of Things (IoT) forensics: Challenges, approaches, and open issues," IEEE Commun. Surveys Tuts., vol. 22, no. 2, pp. 1191–1221, 2020, doi: 10.1109/COMST.2019.2962586.

[3] H. F. Atlam, E. E.-D. Hemdan, A. Alenezi, M. O. Alassafi, and G. B. Wills, "Internet of Things forensics: A review," Internet Things, vol. 11, Art. no. 100220, 2020, doi: 10.1016/j.iot.2020.100220.

[4] P. Lutta, M. Sedky, M. Hassan, U. Jayawickrama, and B. Bakhtiari Bastaki, "The complexity of Internet of Things forensics: A state-of-the-art review," Forensic Sci. Int. Digit. Investig., vol. 38, Art. no. 301210, 2021, doi: 10.1016/j.fsidi.2021.301210.

[5] A. A. Ahmed, K. Farhan, W. A. Jabbar, A. Al-Othmani, and A. G. Abdulrahman, "IoT forensics: Current perspectives and future directions," Sensors, vol. 24, no. 16, Art. no. 5210, 2024, doi: 10.3390/s24165210.

[6] S. Friedl and G. Pernul, "IoT forensics readiness—Influencing factors," Forensic Sci. Int. Digit. Investig., vol. 49, Art. no. 301768, 2024, doi: 10.1016/j.fsidi.2024.301768.

[7] S. Rizvi, M. Scanlon, J. McGibney, and J. W. Sheppard, "Pushing network forensic readiness to the edge: A resource constrained artificial intelligence based methodology," in Proc. 2024 Cyber Research Conf.—Ireland (Cyber-RCI), 2024, pp. 1–8, doi: 10.1109/Cyber-RCI60769.2024.10939120.

[8] S. Rizvi, M. Scanlon, J. McGibney, and J. W. Sheppard, "Application of artificial intelligence to network forensics: Survey, challenges and future directions," IEEE Access, vol. 10, pp. 110362–110384, 2022, doi: 10.1109/ACCESS.2022.3214506.

[9] M. He, Y. Huang, X. Wang, P. Wei, and X. Wang, "A lightweight and efficient IoT intrusion detection method based on feature grouping," IEEE Internet Things J., vol. 11, no. 2, pp. 2935–2949, 2024, doi: 10.1109/JIOT.2023.3294259.

[10] M. Fatima, O. Rehman, I. M. H. Rahman, A. Ajmal, and S. J. Park, "Towards ensemble feature selection for lightweight intrusion detection in resource-constrained IoT devices," Future Internet, vol. 16, no. 10, Art. no. 368, 2024, doi: 10.3390/fi16100368.

[11] M. Fatima, O. Rehman, S. Ali, and M. F. Niazi, "ELIDS: Ensemble feature selection for lightweight IDS against DDoS attacks in resource-constrained IoT environment," Future Gener. Comput. Syst., vol. 159, pp. 172–187, 2024, doi: 10.1016/j.future.2024.05.013.

[12] J. Li, H. Chen, M. O. Shahizan, and L. M. Yusuf, "Enhancing IoT security: A comparative study of feature reduction techniques for intrusion detection system," Intell. Syst. Appl., vol. 23, Art. no. 200407, 2024, doi: 10.1016/j.iswa.2024.200407.

[13] S. F. Misrak and H. M. Melaku, "Lightweight intrusion detection system for IoT with improved feature engineering and advanced dynamic quantization," Discov. Internet Things, vol. 5, Art. no. 97, 2025, doi: 10.1007/s43926-025-00203-8.

[14] X.-H. Nguyen, X.-D. Nguyen, H.-H. Huynh, and K.-H. Le, "Realguard: A lightweight network intrusion detection system for IoT gateways," Sensors, vol. 22, no. 2, Art. no. 432, 2022, doi: 10.3390/s22020432.

[15] S. Roy, J. Li, B.-J. Choi, and Y. Bai, "A lightweight supervised intrusion detection mechanism for IoT networks," Future Gener. Comput. Syst., vol. 127, pp. 276–285, 2022, doi: 10.1016/j.future.2021.09.027.

[16] J. Arshad, M. A. Azad, M. M. Abdeltaif, and K. Salah, "An intrusion detection framework for energy constrained IoT devices," Mech. Syst. Signal Process., vol. 136, Art. no. 106436, 2020, doi: 10.1016/j.ymssp.2019.106436.

[17] J. Azimjonov and T. Kim, "Stochastic gradient descent classifier-based lightweight intrusion detection systems using the efficient feature subsets of datasets," Expert Syst. Appl., vol. 237, Art. no. 121493, 2024, doi: 10.1016/j.eswa.2023.121493.

[18] S. Yaras and M. Dener, "IoT-based intrusion detection system using new hybrid deep learning algorithm," Electronics, vol. 13, no. 6, Art. no. 1053, 2024, doi: 10.3390/electronics13061053.

[19] F. Ebrahimi, R. Javidan, R. Akbari, and Y. Hosseini, "Intrusion detection in the Internet of Things using convolutional neural networks: An explainable AI approach," Cybersecurity, vol. 8, Art. no. 66, 2025, doi: 10.1186/s42400-025-00369-2.

[20] H. Q. Gheni and W. L. Al-Yaseen, "Two-step data clustering for improved intrusion detection system using CICIoT2023 dataset," e-Prime—Adv. Electr. Eng. Electron. Energy, vol. 9, Art. no. 100673, 2024, doi: 10.1016/j.prime.2024.100673.

[21] P. Mahadevappa, R. K. Murugesan, R. Al-amri, R. Thabit, A. H. Al-Ghushami, and G. Alkawsi, "A secure edge computing model using machine learning and IDS to detect and isolate intruders," MethodsX, vol. 12, Art. no. 102597, 2024, doi: 10.1016/j.mex.2024.102597.

[22] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, D. Breitenbacher, A. Shabtai, and Y. Elovici, "N-BaIoT—Network-based detection of IoT botnet attacks using deep autoencoders," IEEE Pervasive Comput., vol. 17, no. 3, pp. 12–22, 2018, doi: 10.1109/MPRV.2018.03367731.

[23] A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for Internet of Things," Future Gener. Comput. Syst., vol. 82, pp. 761–768, 2018, doi: 10.1016/j.future.2017.08.043.

[24] T. Saba, A. Rehman, T. Sadad, H. Kolivand, and S. A. Bahaj, "Anomaly-based intrusion detection system for IoT networks through deep learning model," Comput. Electr. Eng., vol. 99, Art. no. 107810, 2022, doi: 10.1016/j.compeleceng.2022.107810.

[25] B. Sharma, L. Sharma, C. Lal, and S. Roy, "Explainable artificial intelligence for intrusion detection in IoT networks: A deep-learning-based approach," Expert Syst. Appl., vol. 238, Art. no. 121751, 2024, doi: 10.1016/j.eswa.2023.121751.

[26] A. Alabbadi and F. Bajaber, "An intrusion detection system over the IoT data streams using explainable artificial intelligence (XAI)," Sensors, vol. 25, no. 3, Art. no. 847, 2025, doi: 10.3390/s25030847.

[27] S. Bin Hulayyil, S. Li, and N. Saxena, "Explainable AI-based intrusion detection in IoT systems," Internet Things, vol. 31, Art. no. 101589, 2025, doi: 10.1016/j.iot.2025.101589.

[28] O. Arreche, T. Guntur, and M. Abdallah, "XAI-IDS: Toward proposing an explainable artificial intelligence framework for enhancing network intrusion detection systems," Appl. Sci., vol. 14, no. 10, Art. no. 4170, 2024, doi: 10.3390/app14104170.

[29] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?': Explaining the predictions of any classifier," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2016, pp. 1135–1144, doi: 10.1145/2939672.2939778.

[30] S. M. Lundberg, G. Erion, H. Chen, et al., "From local explanations to global understanding with explainable AI for trees," Nat. Mach. Intell., vol. 2, no. 1, pp. 56–67, 2020, doi: 10.1038/s42256-019-0138-9.

[31] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," Future Gener. Comput. Syst., vol. 100, pp. 779–796, 2019, doi: 10.1016/j.future.2019.05.041.

[32] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," Sustain. Cities Soc., vol. 72, Art. no. 102994, 2021, doi: 10.1016/j.scs.2021.102994.

[33] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "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, doi: 10.1109/ACCESS.2022.3165809.

Md Manirul Islam (Member, IEEE) received the B.Sc. degree from the University of Baguio, Baguio, Philippines, and the M.Sc. degree from Charles Sturt University, New South Wales, Australia.

He is currently serving as an Associate Professor with the Faculty of Science and Technology, American International

University-Bangladesh (AIUB), Dhaka, Bangladesh, and is also affiliated with the Faculty of Artificial Intelligence and Digital Technologies, Samarkand State University named after Sharof Rashidov, Samarkand, Uzbekistan. Additionally, he holds the position of Director of the Institute of Continuing Education and Director of IT (Network Operations) at AIUB. His experience spans higher education, information technology management, digital infrastructure, and applied research. His current research interests include artificial intelligence, IoT and edge security, cybersecurity, digital transformation, and intelligent systems.

He is a member of the Institution of Engineering and Technology (IET) and the Association for Computing Machinery (ACM), in addition to IEEE. He is the recipient of the Cisco Be the Bridge Award.

Md Mehedi Hasan Ratul received the B.Sc. degree in Computer Science and Engineering from the American International University–Bangladesh (AIUB), Dhaka, Bangladesh. His research interests include cybersecurity, Internet of Things (IoT) security, network forensics, digital forensics, intrusion detection systems, explainable artificial intelligence (XAI), and machine learning for cybersecurity. He has participated in cybersecurity research projects and technical training focused on ransomware defense, security monitoring, and AI-driven cyber defense.

Md Tahsin Tasnim Aurin completed his B.Sc. in Computer Science and Engineering from American International University-Bangladesh (AIUB). He is currently serving as a Junior Lecturer at the Institute of Continuing Education, AIUB. His research interests include artificial intelligence, machine learning, cybersecurity, ethical AI, autonomous systems, and phishing and smishing detection.

Sazzad Hossain (Senior Member, IEEE) received the B.Sc. degree in computer engineering from Moscow Technical University, Moscow, Russia, in 1994, the M.Sc. degree in electrical and computer engineering from Portland State University, Portland, OR, USA, in 2002, and the Ph.D. degree in electrical and computer engineering from Portland State University, Portland, OR, USA, in 2009.

He is currently a Professor with the Faculty of Artificial Intelligence and Digital Technologies, Samarkand State University, Samarkand, Uzbekistan. He previously served as a Member of the University Grants Commission of Bangladesh and as a Director of Bangladesh Satellite Company Ltd. from 2019 to 2024.

Prof. Hossain is a Fellow of the Institution of Engineers Bangladesh and a member of ACM and IEICE, in addition to being a Senior Member of IEEE. He is also a recipient of the National Digital Transformation Award for contributions to AI and quantum computing.