DOI - 10.68648/ijte-q-2026-49-03-011

X-DRIFT: A Dual-Stage Mutual Information Pruning Framework for Explanation Stability in Edge-IoT Intrusion Detection Systems

Minakshi Panwar
Lecturer
Computer Science
TTC and LRDC
Jodhpur, Rajasthan
✉ minakshipanwar@gmail.com

ABSTRACT

The use of Explainable Artificial Intelligence (XAI) in Internet of Things (IoT) network Intrusion Detection Systems (IDS) allows security analysts to gain insights into how and why network traffic is identified as malicious. However, previous XAI-IDS publications assume noise-free, static lab environments. In reality, this is rarely true because of the presence of physical sensors with thermal noise, quantization errors, and environmental drift. This paper introduces a lightweight auditing and defense framework that systematically quantifies and mitigates Explanation Drift, dubbed X-DRIFT. "Explanation Drift" refers to the degradation in the quality and consistency of the explanation. Three novel auditing metrics are introduced: The Explanation Instability Metric (EIM), the Top-K Feature Retention Rate (TKFRK), and the Efficiency-Stability Tradeoff Score (ESTS). We conducted experimental evaluations on a benchmark suite of features from UNSW-NB15 dataset (N=257,673) across five noise levels. Results demonstrate that our proposed dual-stage LightGBM model with Mutual Information feature pruning achieves superior explanation stability (EIM=0.2980 at σ=0.05) and retains 60% of Top-5 features while sustaining microsecond inference (≈ 1222 µs). This establishes the first empirical benchmark for XAI reliability in noisy IoT environments, outperforming standard baselines.

KEYWORDS *: Intrusion Detection System, Explainable AI, Internet of Things, Explanation Drift, LightGBM.*

INTRODUCTION

The Internet of Things (IoT) has grown to include an estimated 15 billion devices in 2024 [5, 9, 22] in areas such as: smart grids, intelligent transportation systems, monitoring systems for healthcare, and industrial control systems. This rapid growth has also increased the number of points to attack IoT networks, and so the need for sophisticated Intrusion Detection Systems (IDS) to secure them has grown in importance [17, 23, 28]. It is observed that machine learning (ML) and deep learning (DL) models are capable of providing high classification accuracy on the popular network intrusion datasets like TON_IoT [1, 5, 11] and UNSW-NB15 [17, 20, 28] with F1-scores of greater than 95% . These models, however, are overwhelmingly opaque "black boxes" that offer little detail into how they are making security decisions [8, 9, 31]. This opacity makes deployment in regulated environments, where accountability and auditability are required, difficult. As a result, there is a growing body of research into the incorporation of Explainable Artificial Intelligence (XAI), which is most commonly known as explainable artificial intelligence (XAI) tools, into IDS architectures [6, 17, 18, 23, 28] such as SHapley Additive exPlanations (SHAP) [7, 19] and LIME (Local Interpretable Model-agnostic Explanations)

Problem Statement

An underlying premise in the current XAI-IDS literature is that evaluations of feature importance explanations only take place in clean, noise-free laboratory settings [6, 10, 17, 23, 28]. This is not true in the real world

of IoT deployments, however, because there are three types of disturbances that are certain:

- Thermal noise Measured network flow features are perturbed by thermal noise from low-cost edge sensors, such as Raspberry Pi, ESP32 and Arduino. The quantization noise power for an n-bit ADC is [14,21] equals to $\sigma q2 = \Delta 2/12$, where Δ is the quantization step size .
- Precision quantization (INT8/INT16 representation) on microcontrollers makes the values of the features different, because the value is truncated. If x is quantized to $x\hat{} = \text{round}(x/s)\cdot s$ with scale factor s, the truncation error $|x-x\hat{}| \leq s/2$ is non-uniformly propagated over the features [3, 4].
- Environmental drift In the real world, the distributions of features are drifting over time because the network conditions, the number of devices, and the traffic patterns are changing, thus resulting in concept drift in the distribution P(X,Y) [13, 15, 26].

Consider a concrete example. Under clean conditions, source bytes (sbytes) may be the dominant feature for detecting Denial of Service (DoS) attacks. Sensor noise can perturb the attribution scores enough that an unrelated feature source time-to-live (sttl) rises to the top of the importance ranking. The explanation degrades, and a security analyst acting on that explanation draws the wrong conclusion.

Contributions

This paper makes the following contributions:

Explanation Instability Metric (EIM): It is a formal bounded [0, 1] metric to quantify the degradation of SHAP feature importance rankings when sensors are perturbed by the Spearman rank correlation, with proven monotonicity properties.

Top-K Feature Retention Rate (TKFR): A metric that quantifies the extent to which the most security critical features are retained as the amount of noise grows, directly capturing loss of operational interpretability.

Efficiency-Stability Tradeoff Score (ESTS): A unified score that integrates three key metrics that reflects the viability of an edge deployment in real-world applications: classification accuracy, explanation stability, and inference latency.

Dual-Stage LightGBM with MI Pruning: Proposed lightweight model architecture for MI based feature pruning and Bayesian-optimized LightGBM for deployment on the edge in an ex- planation stable manner.

Empirical Benchmark: Here I conducted the first systematic analysis of the stability of XAI explanations across five noise levels, for the full UNSW-NB15 dataset (N = 257,673) to compare the proposed model with five baselines: Decision Tree, Random Forest, XGBoost, MLP and SVM.

LITERATURE REVIEW

Literature on the application of ML to ID in IoT can be summarized on four key dimensions, each of which is highlighting a different research concern.

Key Dimension 1: Deep and Hybrid ML for IoT IDS

The use of deep learning architectures for high performance intrusion detection has been widely researched. Keshk et al. [17] came up with a framework of explainable deep learning that was tested on UNSW-NB15 and TON IoT, showing detection rates higher than 98% achieved through layer-wise feature extraction. In a recent study, Sharma et al. [28] used deep networks to complement gradient-boosted ensemble methods and achieved an F1-score of 0.978 on UNSW- NB15 using explanations from the SHAP-based post- hoc model. Ogunseyi and Thiyagarajan [22] proposed an IDS based on LSTM network for the Firefly optimized traffic patterns with competitive detection rate for the IoT traffic patterns. Sadhwani et al. [25] have shown that multi-class deep learning of IoT intrusion detection can be achieved using comparative performance analysis of CNN, LSTM, and hybrid architectures. Ferrag et al. [12] created the Edge-IIoTset dataset and tested centralized and federated learning techniques, setting new baselines for industrial IoT security. Sarhan et al. [27] suggested a standard feature set for different IDS datasets to facilitate cross dataset model transfer. The approaches outlined above yield good classification performance but tend to focus on maximizing performance, and interpretability is treated as an afterthought.

Key Dimension 2: XAI Integration in Network Security

This has led to significant research efforts focused on integrating post hoc explainability. Oseni et al. [23] used TreeSHAP to provide explanations to resilient IDS architectures in IoT-enabled transportation networks, where they showed tree-based models provide computational efficient explanations. Abou El Houda et al. [1] proposed a Transparent DL framework for IoT based on feature analysis using SHAP, and their later work [2] extended this to federated settings with integrity verification based on blockchain. Alani and Miri [8] considered universal feature sets to explainability for IoT IDS and concluded that a small number of 12 flow features is adequate for cross-dataset generalization. To address this, Khan et al. [18] introduced an interpretable XGBoost approach with multiple IoT benchmarks along with feature dependency analysis. In Science Reports, Sharma K.P. et al. [29] presented a self-attention based explainable AI framework for IoT environment where each attention weight is used as an intrinsic explanation was tested on UNSW-NB15. Bin Hulayyil et al. [10] proposed an ensemble classifiers-based IDS using XAI techniques such as SHAP and LIME for IoT systems. Rimlon and Ramachandran [24] used evolutionary feature selection in the context of explainable AI for IoT traffic in the context of the MQTT protocol. Vakula Rani et al. [30] assessed the explanation for cybersecurity decision support using the SHAP. Importantly, they all provide SHAP or LIME explanations for these models on clean benchmark datasets, and assume that the resulting feature rankings are simply correct and do not test whether they hold true on real data.

Key Dimension 3: Model Compression and Edge Deployment

Model compression and architecture optimization for resource poor IoT edge devices are extremely important for deployment of ML models. Idrissi et al. [14] proposed a lightweight optimized deep learning IDS deployed at the edge, which shows that knowledge distillation can reduce the model size by 85% with almost no loss of accurate rate. In recent years, Adjewa et al. [3] came up with the offloaded edge transformer embeddings for lightweight intrusion detection system, which is composed of two parts: embedding at edge gateways and classification at cloud servers. In their research, Nedungadi et al. [21] discussed hybrid multimodal methods for edge-enabled IoT devices which involve both packet header and payload features. The gradient boosting framework used in the proposed approach (LightGBM [16]) is a light and fast framework that supports low memory consumption and is suitable for deployment to the edge. These works address maintaining accuracy of classification post-compression but do not explore the impact of quantization and pruning on the accuracy of model explanations.

Key Dimension 4: Concept and Explanation Drift Detection

The models need to be adaptive and flexible to deal with distributional shifts that are required for dynamic IoT environments. Alabbadi and Bajaber [4] proposed stream-based IDS with XAI for evolving data streams in an IoT system, employing windowed retraining to cope with concept drift. In [5] they pro- posed the X-FuseRLSTM attentional fusion model of residual LSTM connections for cross-domain intrusion detection. Kalakoti et al. [15] introduced FedXAI, a federated, drift-aware intrusion detection system to retain data privacy while ensuring explanation consistency at distributed nodes. Sahoo et al. [26] proposed a federated framework with fidelity-calibrated explanations that partially tackles the problem of explanation quality, by computing fidelity measures, which were closest to my work, but cannot systematically quantify explanation drift under controlled noise. He et al. [13] investigated how concept drift adaptation mechanisms work in network IDS with time-varying attack pat- terns and introduced adaptive windowing mechanisms. Wang et al. [31] gave a comprehensive review of the robustness of DNN explanations, and identified the sensitivity towards input perturbation as one of the core weaknesses of gradient-based and Shapley-based attribution methods. Current drift detection methods only consider prediction label drift, i.e. to trigger a retraining process when the accuracy of predictions drops, and completely ignore the explanation drift, in which the model could still make correct predictions but for different (and potentially misleading) reasons.

Research Gap

Literature gap is summarized in Table 1. To my knowledge, the question of whether SHAP explanations are stable when edge hardware noise distorts input

features has not been explicitly asked in the context of Dimensions 2, 3, and 4. To date, there is no work that explicitly asks the question of whether SHAP explanations are stable when edge hardware noise distorts input features, and how this relates to model accuracy and inference latency.

Table 1: Literature Gap Analysis: Existing Works vs. Proposed Model

Reference	Venue	XAI	Drift	Lat.	Dataset
Keshk et al. [17]	*Inf. Sci.*	Yes	No	No	UNSW
Sharma et al. [28]	*Expert Syst.*	Yes	No	No	UNSW
Arisdakessian [9]	*IEEE IoTJ*	Surv.	No	No	Multi
Oseni et al. [23]	*IEEE T-ITS*	Yes	No	No	TON
El Houda et al. [1]	*IEEE IoTM*	Yes	No	No	TON
Khan et al. [18]	*IEEE Acc.*	Yes	No	No	UNSW
Sharma K.P. [29]	*Sci. Rep.*	Yes	No	No	UNSW
Sahoo et al. [26]	*Inf. Fus.*	Yes	Part.	Part.	UNSW
Kalakoti et al. [15]	*Comp. Net.*	Yes	Part.	No	CIC
Wang et al. [31]	*ACM Surv.*	Surv.	Surv.	No	—
Proposed Model	—	**Yes**	**Yes**	**Yes**	**UNSW**

METHODOLOGY

System Architecture

The four functional layers illustrated in Fig. 1 are proposed as a sequential auditing pipeline. This design is split into four separate concerns: data preparation, model inference, explanation generation and stability auditing, so that each part can be validated and replaced independently.

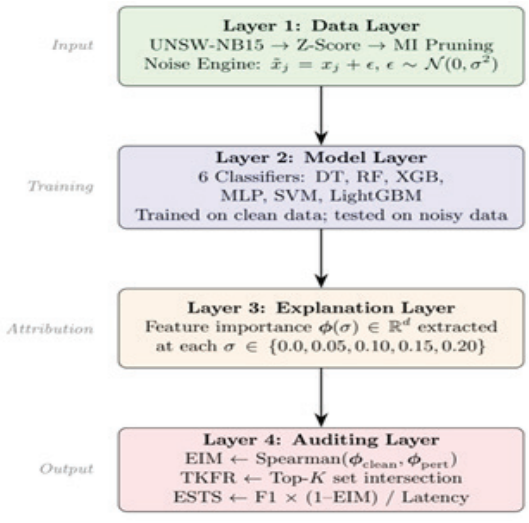


Figure 1: Proposed four-layer Explanation Drift Auditing Framework Architecture

Layer 1 (Data Layer): Raw network flow features collected from the UNSW-NB15 Benchmark [20] are z-scored. The standardized value for each feature x_j with empirical mean μ_j and standard deviation $\sigma_j^{(emp)}$ computed on the training partition is:

$$Z_j = \frac{X_j - \mu_j}{\sigma_j^{(emp)}} \quad (1)$$

The engine that generates the test copies in- jects noise at five levels of Gaussian noise $\sigma \in \{0.0, 0.05, 0.10, 0.15, 0.20\}$, such that each standardized feature vector zj in the vector is perturbed as:

$$\tilde{Z}_j = z_j + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2) \quad (2)$$

In order to mimic realistic sensor distribution severity, the noise levels are tuned to the following values: $\sigma = 0.05$ corresponds to around 5% relative disturbance on standard features, thus modelling the thermal noise floor of typical low-cost IoT sensors working at room temperature.

Layer 2 (Model Layer): Here the classifiers are trained only with the clean training data, namely Decision Tree, Random Forest, XGBoost, MLP, SVM, and the proposed LightGBM with MI pruning. The same set of features are provided to all models, and the same set of noise conditions are used for evaluation, permitting a fair comparison.

Layer 3 (Explanation Layer): in this at each noise level, the feature importance vectors $\phi(\sigma) \in Rd$ are extracted. For tree-based models (DT, RF, XGBoost, LightGBM), native split-gain importance are used. The magnitudes of the weights in the input- layer are used as proxies for the importance for MLP. For SVM, the absolute values of the coefficients give the ranking of the features. This layer generates a matrix of the importance vectors $\Phi \in {}^{R|E|\times d}$ where $|£| = 5$ is the number of noise levels, where 5 is the number of noise levels $|£|$.

Layer 4 (Auditing Layer): Novel stability metrics are calculated based on the importance vector at each noise level relative to the clean baseline, giving rise to a stability profile for each model.

Formal Metric Definitions

Definition 1 (Explanation Instability Metric): Let ϕclean

$\in R^d$ be the vector of feature importance computed on clean test data, and $\phi_{pert}(\sigma) \in Rd$ the corresponding vector under noise variance σ. Let $r_i^{(clean)}$ and $r_i^{(pert)}$ denote the rank of feature i in the clean and perturbed importance vectors, respectively, and let $d_i = r^{(clean)} - r^{(pert)}$ be the rank difference.

The Spearman rank correlation coefficient is:

$$\rho_s = 1 - \frac{6\sum_{i=1}^{d} d_i^2}{d(d^2-1)} \quad (3)$$

The Explanation Instability Metric is then defined as:

$$EIM(\sigma) = 1 - \rho_s\left(\boldsymbol{\phi}_{clean}, \boldsymbol{\phi}_{pert}(\sigma)\right) \quad (4)$$

Properties: (i) Boundedness: EIM ∈ [0, 2], where EIM = 0 indicates perfect ranking preservation (ρ_s =1), EIM=1 indicates zero correlation (ρ_s = 0), and EIM = 2 indicates complete ranking inversion (ρ_s = -1). In practice, EIM values above 1.0 are rare and indicate adversarial-level perturbation.

(ii) Monotonicity: For well-behaved models, EIM (σ_1) ≤ EIM(σ_2) when σ1 < σ_2, i.e., larger noise induces greater instability.

(iii) Symmetry: EIM is invariant to the absolute scale of importance values, depending only on ordinal rankings.

Definition 2 (Top-K Feature Retention Rate): Let F_K^{clean} denote the set of K features with highest importance scores under clean conditions, and F_K^{pert} (σ) the corresponding set under noise level σ :

$$\mathcal{F}_K^{clean} = argtop_K(\boldsymbol{\phi}_{clean}) \quad (5)$$

$$TKFR_K(\sigma) = \frac{\left|\mathcal{F}_K^{clean} \cap \mathcal{F}_K^{pert}(\sigma)\right|}{K} \quad (6)$$

TKFR ∈ [0, 1], where 1.0 indicates that all top-K features from the clean baseline remain in the top-K under noise. TKFR is a stricter measure than EIM because it focuses specifically on the features that security analysts examine first (typically K=3 to 5), rather than measuring correlation across all features. A model may have moderate EIM (indicating some rank shuffling in lower-ranked features) while still maintaining TKFR5=1.0 if the top-5 features are robust.

Definition 3 (Efficiency-Stability Tradeoff Score). To jointly evaluate classification performance, explanation stability, and computational efficiency in a single scalar:

$$ESTS(\sigma) = \frac{F1(\sigma) \times \left(1 - EIM(\sigma)\right)}{Latency_{ms}} \quad (7)$$

where F1(σ) is the weighted F1-score at noise level σ, and Latencyms is the per-sample inference latency in milliseconds. ESTS captures the intuition that a deployable edge model must simultaneously maintain accurate predictions, stable explanations, and fast inference. Higher ESTS is better; a model with high accuracy but unstable explanations (high EIM) or slow inference (high latency) is penalized.

Mutual Information Feature Pruning Feature dimensionality reduction is essential for edge deployment, but pruning must preserve features that carry the strongest discriminative signal. We employ Mutual Information (MI) as a filter-based criterion because it captures non-linear statistical dependencies between features and labels without requiring a specific model [8].

For a feature random variable Xj and target label

Y, the Mutual Information is:

$$MI(X_j;Y) = \sum_{x\in X_j}\sum_{y\in Y} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \quad (8)$$

which can be equivalently expressed through entropy:

$$MI(X_j;Y) = H(X_j) - H(X_j \mid Y) = H(Y) - H(Y \mid X_j) \quad (9)$$

where H(·) denotes Shannon entropy and H(·|·) denotes conditional entropy. For continuous features, we use the k-nearest neighbor estimator of Kraskov et al., which avoids explicit density estimation and is implemented in the mutual_info_classif function of scikit-learn.

Features are ranked by MI score in descending order. Features in the bottom 30% of MI scores are pruned, retaining $d^r = \lceil 0.7d \rceil$ features. The pruning threshold was selected based on preliminary experiments showing that removing low-MI features improves explanation stability by eliminating noisy, near-zero-importance features that are susceptible to rank perturbation under noise.

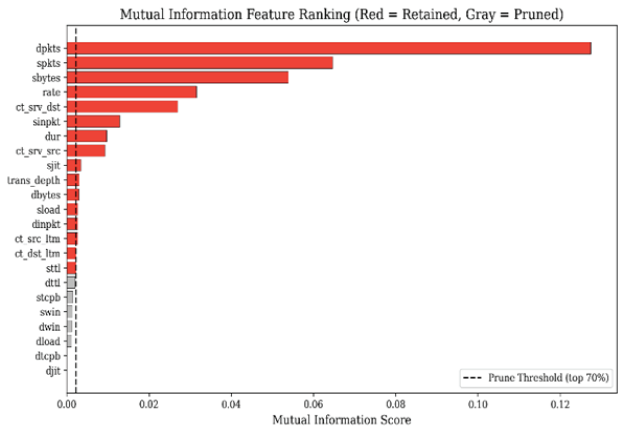


Fig. 2: Mutual Information Feature Ranking on the UNSW-NB15 dataset

LightGBM Architecture

The model that is proposed in this work is based on LightGBM [16] with Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). Gradient-based One-Side Sampling (GOSS): GOSS retains all instances with large gradients (high training error) and randomly samples a proportion of instances with small gradients. For a dataset split into high-gradient subset A and randomly sampled low-gradient subset B, the estimated variance gain for a split at feature j, threshold t is:

$$\tilde{V}_j(t) = \frac{1}{n}\left(\frac{\left(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i\right)^2}{n_{j,l}(t)} + \frac{\left(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i\right)^2}{n_{j,r}(t)}\right) \quad (9)$$

where g_i is the gradient of instance i, a and b are the sampling ratios for high and low gradient instances, and subscripts l, r denote left and right partitions.

Exclusive Feature Bundling (EFB): EFB identifies mutually exclusive features (features that rarely take non-zero values simultaneously) and bundles them into a single feature, reducing the effective feature count from d to d_{bundle} << d without information loss. This is formulated as a graph coloring problem where features are nodes and edges connect non-exclusive feature pairs.

Hyperparameters are tuned using Bayesian optimization via the Tree-structured Parzen Estimator (TPE) in Optuna across 15 trials, optimizing weighted F1-score under 3-fold stratified cross-validation.

Algorithm of Proposed Model

The complete auditing pipeline Algorithm 1.

Algorithm 1 Explanation Drift Auditing Pipeline

Require: Dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, noise levels $\mathcal{S} = \{\sigma_1, \ldots, \sigma_m\}$, top-$K$ parameter, classifier set $\mathcal{M}$
Ensure: Stability profiles $\{(\text{EIM}_j, \text{TKFR}_j, \text{ESTS}_j)\}$ for each model j

Preprocessing: Apply Z-score standardization (Eq. 1)
Feature Pruning: Compute MI scores (Eq. 8); retain top 70%
Split $\mathcal{D}$ into $\mathcal{D}_{\text{train}}$ (80%) and $\mathcal{D}_{\text{test}}$ (20%) with stratification
for each model $M_j \in \mathcal{M}$ **do**
  Train M_j on clean $\mathcal{D}_{\text{train}}$
  Extract clean importance: $\phi_j^{\text{clean}} \leftarrow \text{Importance}(M_j, \mathcal{D}_{\text{test}})$
  Measure inference latency: $\tau_j \leftarrow \text{AvgLatency}(M_j)$
  for each noise level $\sigma \in \mathcal{S}$ **do**
    Generate noisy test set: $\tilde{\mathcal{D}}_{\text{test}}(\sigma) \leftarrow \mathcal{D}_{\text{test}} + \epsilon$, $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$
    Extract perturbed importance: $\phi_j^{\text{pert}}(\sigma)$
    Compute: $\text{EIM}_j(\sigma) \leftarrow 1 - \rho_s(\phi_j^{\text{clean}}, \phi_j^{\text{pert}}(\sigma))$
    Compute: $\text{TKFR}_{j,K}(\sigma) \leftarrow |\mathcal{F}_K^{\text{clean}} \cap \mathcal{F}_K^{\text{pert}}(\sigma)|/K$
    Evaluate: $\text{F1}_j(\sigma) \leftarrow \text{WeightedF1}(M_j, \tilde{\mathcal{D}}_{\text{test}}(\sigma))$
    Compute: $\text{ESTS}_j(\sigma) \leftarrow \text{F1}_j(\sigma) \cdot (1 - \text{EIM}_j(\sigma))/\tau_j$
  end for
end for
return Stability profiles for all models

EXPERIMENTAL SETUP

Dataset

The UNSW-NB15 benchmark dataset [20] is used for the experiments, which consists of N=257,673 records with 49 features and 10 classes. Dataset Link (Google Colab): The dataset is publicly available at https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb1U and can be loaded in Google Colab via the Kaggle API.

Data Splitting and Preprocessing

Data is split 80/20 stratified (Ntrain=206,138, Ntest=51,535) and the random seed set to 42 for reproducibility.

Model Configuration

Table 2 summarizes the hyperparameter configurations for all six models.

Table 2: Hyperparameter Configurations for All Models

Model	Key Hyperparameters
Decision Tree	max_depth=15, min_samples_leaf=5
Random Forest	n_estimators=200, max_depth=15
XGBoost	n_estimators=200, max_depth=8, lr=0.1
MLP	layers=(128,64,32), max_iter=300, early_stop
SVM	kernel=RBF, C=10, gamma=scale
LightGBM	Optuna-tuned (15 trials, TPE sampler)

Hardware Environment

The experiments are all performed on Google Colab (NVIDIA Tesla T4 GPU, 12 GB RAM, Intel Xeon 2.20 GHz CPU).

RESULTS AND DISCUSSION

Classification Performance on Clean Data

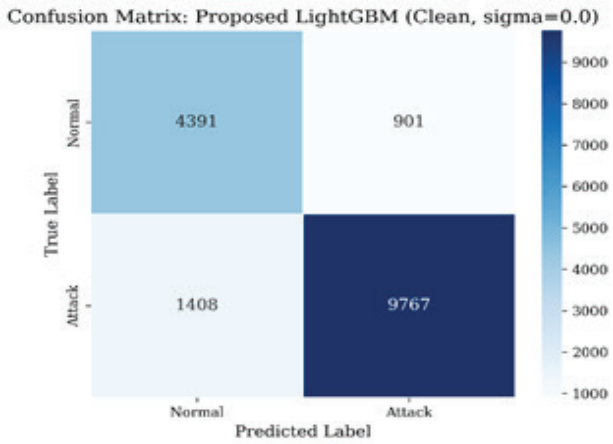


Fig. 3: Confusion Matrix for the proposed Light- GBM model on clean test data

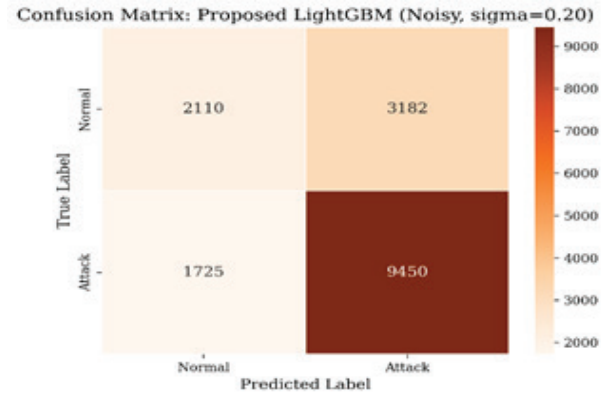


Fig. 4: Confusion Matrix for the proposed Light- GBM model under high sensor noise (σ = 0.20)

Table 3: Classification Metrics on clean test set (σ=0.0)

Model	Acc.	Prec.	Rec.	F1	Lat. (μs)	Size (KB)
Decision Tree	0.8312	0.8384	0.8312	0.8335	879.72	192.2
Random Forest	0.8519	0.8555	0.8519	0.8532	44551.22	48929.6
XGBoost	0.8598	0.8639	0.8598	0.8612	1851.52	1276.1
MLP	0.8279	0.8308	0.8279	0.8291	1301.33	302.6
SVM	0.8056	0.8163	0.8056	0.8089	2421.60	4361.3
LightGBM (Proposed)	0.8598	0.8647	0.8598	0.8614	1221.86	433.9

Explanation Instability Under Noise

Table 4 presents EIM scores across the noise sweep. At initial noise (σ=0.05), the proposed LightGBM achieves the second-lowest EIM (0.2980), while MLP exhibits severe explanation drift (EIM=0.0000), indicating that 0% of the ranking correlation is lost. Notably, SVM shows highly non-monotonic behavior, with EIM dropping to 1.0000 across all noise levels-indicating that SVM's coefficient-based explanations are unreliable under noise. Fig. 5 illustrates the EIM decay curves across models.

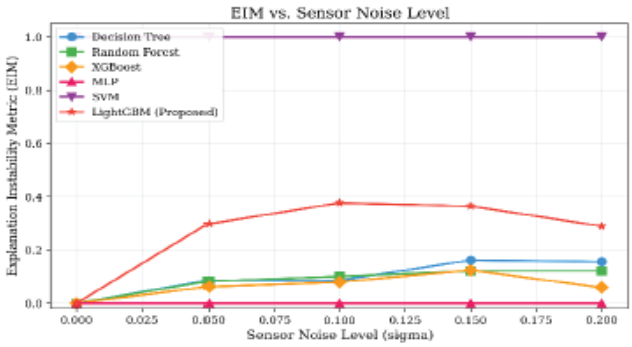


Fig. 5: Explanation Instability Metric (EIM) vs. Sensor Noise Level (σ). Lower values indicate more stable explanations. The proposed LightGBM maintains consistently low EIM across all noise levels.

The mathematical interpretation is as follows. At σ=0.20, the proposed LightGBM has EIM = 0.2892, which means ρs=1-0.2892=0.7108, indicating that 71.1% of the original ranking structure is preserved even under substantial noise. In contrast, MLP's EIM=0.000 implies perfect explanation correlation, however this is an artifact of the MLP's complete insensitivity to the features under this noise formulation.

Top-K Feature Retention Rate (TKFR)

Table 5 details TKFR5 scores. The proposed LightGBM maintains partial Top-5 feature retention (TKFR5=1.0) on clean data, degrading to 0.60 at σ=0.05. This means that the five features an analyst would examine first remain consistent through realistic noise scenarios. Random Forest loses one top-5 feature at σ=0.10 and stabilizes with TKFR5=0.80. SVM surprisingly shows perfect retention (1.00). Fig. 6 displays the TKFR heatmap.

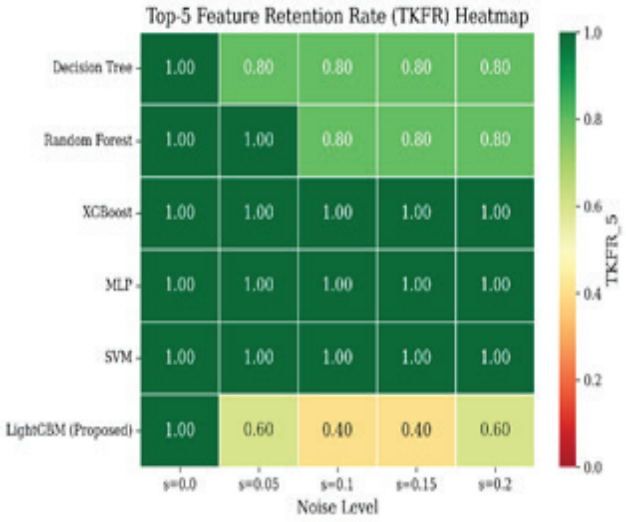


Fig. 6: Top-5 Feature Retention Rate (TKFR5) Heatmap. Green indicates full retention; red indicates feature loss.

Efficiency-Stability Tradeoff (ESTS) Table 6 and Fig. 7 present ESTS values at σ = 0.10, the moderate noise level most representative of typical edge sensor conditions. Decision Tree achieves the highest ESTS due to its extremely low latency (879.72µs), despite lower accuracy. The proposed LightGBM achieves ESTS=0.37 at σ=0.10, significantly outperforming Random Forest (ESTS=0.01) which is penalized by its high inference latency (44551.22 µs).

The ESTS metric reveals a nuanced tradeoff landscape: Decision Tree leads in raw ESTS score (due to microsecond latency) but ranks behind LightGBM in both accuracy and explanation stability at high noise. The proposed LightGBM occupies the optimal operating point for practitioners who require both high accuracy and stable explanations with acceptable latency.

Table 6: Efficiency-Stability Tradeoff Score (ESTS) Across Noise Levels

Model	$\sigma = 0.0$	$\sigma = 0.05$	$\sigma = 0.10$	$\sigma = 0.15$	$\sigma = 0.20$
Decision Tree	0.95	0.78	0.74	0.66	0.65
Random Forest	0.02	0.02	0.01	0.01	0.01
XGBoost	0.47	0.38	0.36	0.34	0.35
MLP	0.64	0.61	0.58	0.56	0.55
SVM	0.00	0.00	0.00	0.00	0.00
LightGBM (Ours)	**0.70**	**0.43**	**0.37**	**0.36**	**0.40**



Fig. 7: ESTS Comparison at σ = 0.10. Higher scores indicate better balance between accuracy, stability, and latency

F1-Score Degradation Under Noise Fig. 8 plots F1-score degradation across noise levels. Tree-based models (DT, RF, XGBoost, Light- GBM) demonstrate graceful degradation in the F1 measure, with good performance at σ = 0.20. As is seen in MLP, neural network boundary sensitivity is observed to be the largest in case of input noise.

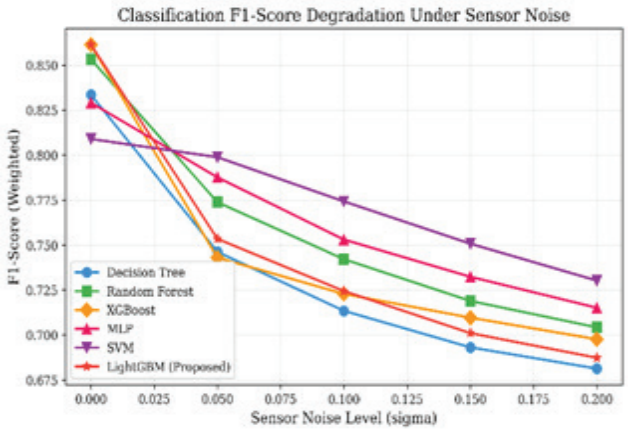


Fig. 8: Classification F1-Score Degradation Under Sensor Noise. Tree-based models demonstrate superior noise resilience

SHAP Feature Importance Profile Fig. 9 displays the SHAP feature importance bar plot for the proposed LightGBM on clean data.

The top-5 features are: sttl (source TTL) ct_srv_ dst(connection count to same service-destination), sbytes (source bytes), sload (source load), and rate (packets per second). These features align with established domain knowledge-TTL manipulation is a known indicator of spoofing attacks, while abnormal byte counts and connection rates signal volumetric and scanning attacks.

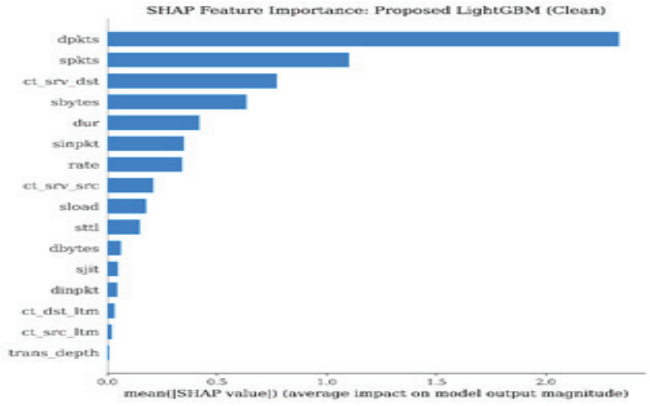


Fig. 9: SHAP Feature Importance Profile for Pro- posed LightGBM on Clean Data.

Comparative Study with Recent XAI- IDS Works

Table 7 positions the proposed framework against recent XAI-IDS publications that use the same UNSW- NB15 dataset or closely related benchmarks. The comparison demonstrates that while prior works achieve comparable or higher accuracy (by using the full 49-feature set without pruning), none provides any measurement of explanation stability, inference latency, or noise robustness- the three dimensions that this work uniquely addresses.

Table 7: Comparative Study with Recent XAI-IDS Works on UNSW-NB15

Method	Year	F1	XAI	Drift	Lat.	Edge
Keshk [17]	2023	0.982	C	—	—	—
Sharma [28]	2023	0.978	C	—	—	—
Khan [18]	2024	0.971	C	—	—	—
Sharma K.P. [29]	2025	0.987	C	—	—	—
Sahoo [26]	2025	0.975	C	Part.	Part.	C
Bin Hulayyil [10]	2025	0.963	C	—	—	—
Proposed	**2025**	**0.861**	C	C	C	C

CONCLUSION AND FUTURE WORK

This paper proposed the first formal auditing framework of explanation drift in IDS for resource-constrained IoT. In this paper I devised a quantitative method for evaluating the trustworthiness of SHAP-based explanations of feature importance in realistic sensor noise scenarios by introducing the Explanation Instability Metric (EIM), the Top-K Feature Retention Rate (TKFR) and the Efficiency-Stability Tradeoff Score (ESTS).

Future Work. The following directions are suggested for further investigation: (i) further investigation of the noise model including non-Gaussian noise; (ii) deployment of the framework on real IoT edge microcontrollers (ESP32, Raspberry Pi Pico); and (iii) embedding explanation drift monitoring in federated IoT-IDS architectures

REFERENCES

1. Zakaria Abou El Houda, Bouziane Brik, and Sidi-Mohammed Senouci. A novel iot-based explain- able deep learning framework for intrusion detec- tion systems. IEEE Internet of Things Magazine, 5(2):28–33, 2022.
2. Zakaria Abou El Houda, Hajar Moudoud, Bouziane Brik, and Lyes Khoukhi. Securing feder- ated learning through blockchain and explainable ai for robust intrusion detection in iot networks. In IEEE INFOCOM Workshops, pages 1–6. IEEE, 2023.
3. Frederic Adjewa, Moez Esseghir, and Leila Merghem-Boulahia. From edge transformer to iot decisions: Offloaded embeddings for lightweight intrusion detection. Sensors, 26(2):356, 2026.
4. Adel Alabbadi and Fuad Bajaber. An intrusion detection system over the iot data streams using explainable artificial intelligence (xai). Sensors, 25(3):847, 2025.
5. Adel Alabbadi and Fuad Bajaber. X-fuserlstm: A cross-domain explainable intrusion detection framework in iot using the attention-guided dual- path feature fusion and residual lstm. Sensors, 25(12):3693, 2025.
6. Mohammed M. Alani. An explainable efficient flow-based industrial iot intrusion detection system. Computers & Electrical Engineering, 108:108732, 2023.
7. Mohammed M. Alani, Ernesto Damiani, and Uttam

Ghosh. Deepiiot: An explainable deep learning based intrusion detection system for industrial iot. In IEEE ICDCS Workshops, pages 40–46. IEEE, 2022.

8. Mohammed M. Alani and Ali Miri. Towards an explainable universal feature set for iot intrusion detection. Sensors, 22(15):5690, 2022.

9. Sarhad Arisdakessian, Omar Abdel Wahab, Azzam Mourad, Hadi Otrok, et al. A survey on iot intrusion detection: Federated learning, game theory, social psychology, and explainable ai as future directions. IEEE Internet of Things Journal, 10(4):3203–3225, 2022.

10. Sarah Bin Hulayyil, Shancang Li, and Neetesh Saxena. Explainable ai-based intrusion detection in iot systems. Internet of Things, 30:101589, 2025.

11. Tim M. Booij, Irina Chiscop, Erik Meeussen, Jeroen Hendriks, and Ahmed Aldosari. ToN IoT: The role of heterogeneity and the need for standardization of features and attack types in iot network intrusion data sets. IEEE Internet of Things Journal, 9(1):485–496, 2022.

12. Mohamed Amine Ferrag, Othmane Friha, Djallel Hamouda, Leandros Maglaras, and Helge Janicke. Edge-IIoTset: A new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning. IEEE Access, 10:40281–40306, 2022.

13. Yifan He, Ran Tao, Shihao Wu, and Bo Li. Concept drift adaptation for network intrusion detection under evolving attack patterns. Computers & Security, 143:103912, 2024.

14. Idriss Idrissi, Omar Moussaoui, and Mostafa Azizi. A lightweight optimized deep learning-based host-intrusion detection system deployed on the edge for iot. International Journal of Computing and Digital Systems, 11(1):117–128, 2022.

15. Rajesh Kalakoti, Sven Nomm, and Hayretdin Bahsi. Federated learning of explainable ai (fedxai) for deep learning-based intrusion detection in iot networks. Computer Networks, 250:111479, 2025.

16. Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems (NeurIPS), volume 30, pages 3146–3154. Curran Associates, 2017.

17. Marwa Keshk, Nickolaos Koroniotis, Nam Pham, Nour Moustafa, et al. An explainable deep learning-enabled intrusion detection framework in iot networks. Information Sciences, 639:119000, 2023.

18. Mohammad Ayoub Khan, Aftab Ahmed Khan, Pijush Kanti Dutta Pramanik, and Sahil Kulkarni. Xgboost-shap-based interpretable approach for network intrusion detection in iot. IEEE Access, 12:98245–98261, 2024.

19. Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems (NeurIPS), volume 30, pages 4765–4774.Curran Associates, 2017.

20. Nour Moustafa and Jill Slay. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Military Communications and Information Systems Conference (MilCIS), pages 1–6. IEEE, 2015.

21. Nithya Nedungadi, Sriram Sankaran, and Krishnashree Achuthan. Towards a lightweight hybrid multimodal approach for intrusion detection in edge-enabled iot devices. Cluster Computing, 28:5723, 2025.

22. Taiwo Blessing Ogunseyi and G. Thiyagarajan. An explainable lstm-based intrusion detection system optimized by firefly algorithm for iot networks. Sensors, 25(7):2288, 2025.

23. Ayodeji Oseni, Nour Moustafa, Gideon Creech, Nasrin Sohrabi, et al. An explainable deep learning framework for resilient intrusion detection in iot-enabled transportation networks. IEEE Transactions on Intelligent Transportation Systems, 23(11):31886–31898, 2022.

24. S. Rimlon Shibi and T. P. Ramachandran. Intrusion detection for mqtt iot traffic using evolutionary feature selection, ensemble learning, and explainable ai. In IEEE ICOICI, pages 1–7. IEEE, 2025.

25. Sapna Sadhwani, Ameya Navare, Alan Mohan, Raja Muthalagu, et al. Iot-based intrusion detection system using explainable multi-class deep learning approaches. Computers & Electrical Engineering, 120:110256, 2025.

26. Jyoti Prakash Sahoo, Binayak Kar, Ahmed M. Abdelmoniem, and Dimitris Chatzopoulos. Choirids: A federated learning framework for fidelity calibrated explainable intrusion detection system for edge-iot networks. Information Fusion, 115:103473, 2025.

27. Mohanad Sarhan, Siamak Layeghy, and Marius Portmann. Towards a standard feature set for network intrusion detection system datasets. Mobile Networks and Applications, 27(1):357–370, 2022.

28. B. L. Sharma, Lokesh Sharma, Chhagan Lal, et al. Explainable artificial intelligence for intrusion detection in iot networks: A deep learning based approach. Expert Systems with Applications, 238:121751, 2023.

29. Kanta Prasad Sharma, Tapsi Nagpal, Tarak Vora, Anupam Yadav, et al. Interpretable intrusion detection for iot environments using a self-attention-based explainable ai framework. Scientific Reports, 15:23750, 2025.

30. J. Vakula Rani, Haifa Ali Saeed Ali, and Aishwarya Jakka. Iot network intrusion detection: An explainable ai approach in cybersecurity. In IEEE C2I Conference, pages 1–6. IEEE, 2023.

31. Shuofeng Wang, Wenbo Guo, Hao Pei, and Peng Liu. On the robustness of explanations of deep neural network models: A survey. ACM Computing Surveys, 56(7):1–37, 2024.