

SHAP-BASED EXPLAINABILITY IN IOT INTRUSION DETECTION: A COMPARATIVE ANALYSIS OF MACHINE LEARNING MODELS

Isokova Adiba Shamshiddin Kizi

Advanced Research Institute, "New Uzbekistan" University, Tashkent, Uzbekistan

E-mail: missadiba010120@gmail.com

KEYWORDS

Explainable AI, SHAP, Intrusion Detection System, IoT Security, Machine Learning, Vehicle Telematics, Feature Importance, CatBoost, XGBoost, LightGBM, Random Forest

ABSTRACT

The deployment of machine learning (ML)-based intrusion detection systems (IDS) in Internet of Things (IoT) and telematics environments has achieved strong detection performance yet suffers from a critical opacity problem: security analysts cannot understand why a model flags a given network flow as malicious. This paper presents a rigorous comparative evaluation of five widely-adopted ML classifiers — Random Forest (RF), Extreme Gradient Boosting (XGBoost), CatBoost, LightGBM, and Support Vector Machine (SVM) — for IoT intrusion detection, coupled with a systematic SHAP (SHapley Additive exPlanations)-based explainability analysis covering both global feature attribution and local instance-level interpretation. Experiments are conducted on two benchmark datasets: CICIDS2017 and UNSW-NB15, encompassing 15 attack categories. Results demonstrate that ensemble tree-based models — particularly XGBoost (F1: 99.41%, AUC: 0.9987 on CICIDS2017) and LightGBM (F1: 99.38%, AUC: 0.9985) — consistently outperform SVM across all metrics and datasets, while SHAP analysis reveals dataset-specific feature importance divergences that carry direct implications for model transferability. Global SHAP analysis identifies flow-duration, packet-length statistics, and inter-arrival timing as the most discriminative features across both datasets, whereas local SHAP waterfall plots expose model-specific reasoning pathways that differ significantly between tree-based and kernel-based classifiers. The vehicle-telematics applicability of these findings is examined through a dedicated analysis of SHAP explanations under simulated CAN-bus traffic features. This work bridges the explainability gap in IoT IDS research and provides actionable guidance for security practitioners deploying transparent ML-based detection systems.

I. INTRODUCTION

The proliferation of Internet of Things (IoT) devices across smart homes, industrial automation, healthcare infrastructure, and connected transportation has fundamentally transformed the cyber-threat landscape. By 2025, the global installed base of IoT endpoints exceeded 16.7 billion units [1], and projections indicate continued exponential growth through the end of the decade. This density

creates a vast and heterogeneous attack surface that adversaries exploit through distributed denial-of-service (DDoS) campaigns, botnet infiltration, protocol-specific exploits, and man-in-the-middle attacks. The telematics domain — encompassing vehicle-to-everything (V2X) communications, Controller Area Network (CAN) bus systems, and fleet management infrastructure — faces an analogous escalation of threats, with connected

vehicles now representing high-value targets for both cybercriminal and nation-state actors [2].

Artificial intelligence (AI)-based intrusion detection systems have emerged as the principal defense mechanism against these evolving threats, offering detection capabilities that rule-based systems cannot match in high-dimensional, high-velocity traffic environments. Machine learning classifiers, in particular, have demonstrated the ability to distinguish benign traffic from attack traffic with accuracy rates exceeding 99% on standard benchmark datasets [3]. However, this performance comes at a significant operational cost: the decision logic of high-performing ML models — particularly ensemble methods and deep learning architectures — is fundamentally opaque. A security analyst confronting an alert from a black-box IDS cannot determine which packet attributes triggered the alarm, cannot assess whether the detection is generalizable to novel attack variants, and cannot confidently defend the system's decision in a regulatory or legal context.

Explainable artificial intelligence (XAI) has emerged as the methodological response to this opacity problem. Among the available XAI techniques, SHAP (SHapley Additive exPlanations), introduced by Lundberg and Lee [4], has established itself as the dominant approach in network security research owing to its theoretical guarantees of consistency and local accuracy, its model-agnostic applicability, and its ability to decompose predictions into additive feature contributions grounded in cooperative game theory. However, despite the growing body of literature applying SHAP to IDS, a systematic comparative evaluation of SHAP-based explainability across multiple competing ML classifiers — under identical experimental conditions on multiple IoT benchmark datasets — remains absent from the literature.

This gap is consequential for practitioners. A security engineer selecting an ML model for deployment in a resource-constrained IoT gateway must balance detection performance against computational overhead and explanation quality. Without a rigorous comparison that simultaneously evaluates all three dimensions, model selection

remains an ad-hoc process. Furthermore, the domain specificity of SHAP explanations — whether feature importance rankings are consistent across different network traffic datasets — has not been systematically investigated, leaving open the question of whether a model trained on one IoT benchmark can be trusted in a deployment context with different traffic characteristics.

The present paper addresses these gaps through a unified experimental framework. We evaluate five ML classifiers (RF, XGBoost, CatBoost, LightGBM, SVM) on CICIDS2017 and UNSW-NB15, apply SHAP TreeExplainer and KernelExplainer across all models, and conduct a systematic analysis of global feature attribution, local instance-level explanations, cross-dataset feature consistency, and SHAP computational overhead. Additionally, we extend the analysis to the vehicle telematics domain by examining SHAP explanations under CAN-bus-relevant feature subsets, directly connecting our findings to the broader autonomous IDS research agenda.

The main contributions of this work are as follows:

1. A rigorous comparative performance evaluation of five ML classifiers for multi-class IoT intrusion detection on two independent benchmark datasets.
2. A systematic SHAP-based global explainability analysis that ranks and compares feature importance across all five classifiers and both datasets.
3. A local SHAP explanation analysis that reveals model-specific reasoning pathways for individual attack instances, including waterfall and force-plot interpretations.
4. A cross-dataset SHAP consistency analysis quantifying the transferability of feature importance rankings between CICIDS2017 and UNSW-NB15.
5. A dedicated telematics-domain extension examining SHAP explanations under CAN-bus traffic features, bridging IoT and vehicular intrusion detection research.
6. An analysis of SHAP computational overhead across models, providing actionable guidance

for practitioners deploying XAI in resource-constrained IoT environments.

The remainder of this paper is organized as follows: Section II reviews related work. Section III describes the datasets, preprocessing, and experimental setup. Section IV presents and analyzes performance results. Section V details the SHAP global explainability analysis. Section VI presents local SHAP interpretations. Section VII analyzes SHAP consistency across datasets. Section VIII extends findings to the telematics domain. Section IX discusses implications and limitations. Section X concludes the paper.

II. RELATED WORK

A. Machine Learning for IoT Intrusion Detection

The application of machine learning to network intrusion detection has a rich history that predates the IoT era, but the specific challenges of IoT environments — device heterogeneity, traffic volume, class imbalance, and resource constraints — have motivated a substantial body of recent work. Siganos et al. [5] conducted a comparative evaluation of Random Forest, XGBoost, Decision Tree, and DNN on CIC-IoT-2022 and IEC 60870-5-104 datasets, demonstrating that ensemble tree-based methods consistently outperformed single-model approaches and that SHAP explanations revealed dataset-specific feature importance patterns. Alabbadi and Bajaber [6] proposed a multi-model architecture integrating 1D-CNN, DNN, and TabNet with SHAP explainability on TON-IoT, achieving high detection accuracy while enabling fine-grained feature attribution analysis. Alfahaid et al. [7] evaluated six classical ML classifiers augmented with SHAP on the IoT Modbus dataset, finding that protocol-specific features dominated the SHAP value distributions — a finding with direct relevance to our cross-dataset consistency analysis.

More recently, Wang et al. [8] applied SHAP and LIME to Decision Tree and Logistic Regression models on CICIDS2017 and UNSW-NB15, demonstrating that combining the two XAI techniques yields more comprehensive transparency than either alone. Mohale and Obagbuwa [9] evaluated XGBoost, CatBoost, and Random Forest with SHAP, LIME, and ELI5 on UNSW-NB15,

reporting that XGBoost and CatBoost achieved the highest accuracy of 87% while maintaining strong interpretability. Jain et al. [10] proposed a CNN-XGBoost hybrid achieving 99.92% accuracy on CICIOT2024-DIAD, with SHAP providing packet-level feature attribution. These works motivate our unified comparison but do not provide a systematic cross-model SHAP analysis under identical experimental conditions.

B. XAI Techniques in Network Security

The application of SHAP to network security has matured significantly since its introduction by Lundberg and Lee [4]. Neupane et al. [11] provided an early survey contrasting SHAP, LIME, and Grad-CAM for IoT IDS, establishing that SHAP's game-theoretic guarantees of local accuracy and consistency make it preferable for security contexts where explanation fidelity is paramount. Pawlicki et al. [12] examined the practical deployment of SHAP and LIME across diverse IDS architectures, highlighting computational overhead as the primary barrier to real-time application. Sharma et al. [13] integrated DNN and CNN with SHAP and LIME on NSL-KDD and UNSW-NB15, demonstrating that dual-dataset evaluation exposes model robustness limitations that single-dataset studies miss — a methodological principle we adopt in the present work.

Bilal et al. [14] combined federated DNN with SHAP and LIME across three IoT datasets, demonstrating the scalability of SHAP explanations in distributed settings. Ebrahimi et al. [15] proposed a 1D-CNN augmented with SHAP and LIME on TON-IoT, reporting competitive accuracy with interpretable feature attributions. Khan et al. [16] systematically reviewed XAI-based IDS for Industry 5.0, identifying SHAP and LIME as dominant methods but noting significant gaps in adversarial robustness and cross-domain applicability.

C. XAI for Telematics and Vehicular Security

The telematics domain has seen growing adoption of SHAP-based explainability for intrusion detection in vehicular networks. Lundberg et al. [17] proposed VisExp, a SHAP-based visualization framework for DNN-based in-vehicle IDS on CAN

bus data, demonstrating that visual explanations significantly increase expert trust. Khan et al. [18] developed a Transformer-CNN hybrid with SHAP for VANET misbehavior detection on VeReMi, achieving 96.15% multi-class accuracy. Arslan et al. [19] proposed a hierarchical ensemble (LightGBM, XGBoost, CatBoost, TabNet) with TreeSHAP for intelligent transportation systems, achieving 98.20% accuracy with 3.40 ms inference latency. Taheri et al. [20] integrated federated DNN with SHAP and LIME for connected vehicle IDS on CICEVSE2024 and CICIoV2024. These works inform our telematics extension in Section VIII but

do not address the cross-model SHAP comparison that constitutes the primary focus of the present paper.

D. Positioning of the Present Work

Table I situates the present work within the existing literature across five dimensions: number of ML models compared, datasets evaluated, SHAP analysis type (global/local), cross-dataset consistency analysis, and telematics applicability. The table reveals that no existing study simultaneously addresses all five dimensions, confirming the novelty of our contribution.

TABLE I.

Comparison of Present Work with Related Studies

Study	Year	Models	Datasets	SHAP Type	Cross-DS	Telematics
Siganos et al. [5]	2023	RF, XGBoost, DT, DNN	CIC-IoT-2022, IEC	Global	No	No
Wang et al. [8]	2025	DT, LR	CICIDS2017, UNSW-NB15	Global+Local	Partial	No
Mohale & Obagbuwa [9]	2025	XGBoost, CatBoost, RF	UNSW-NB15	Global	No	No
Arslan et al. [19]	2026	LightGBM+XGBoost+CatBoost+TabNet	UNSW-NB15	Global	No	Yes
Alfahaid et al. [7]	2025	RF, XGBoost, SVM, others	IoT Modbus	Global	No	No
This work	2026	RF, XGBoost, CatBoost, LightGBM, SVM	CICIDS2017 + UNSW-NB15	Global+Local	Yes	Yes

DS = Dataset; Global = global feature attribution; Local = instance-level explanation

III. DATASETS, PREPROCESSING, AND EXPERIMENTAL SETUP

A. Datasets

Two publicly available benchmark datasets are used in this study: CICIDS2017 and UNSW-NB15. These datasets are selected because (i) they are among the most widely adopted benchmarks in IoT IDS research, enabling direct comparison with prior work; (ii) they differ substantially in traffic generation methodology, feature space, and attack

taxonomy, enabling a meaningful cross-dataset consistency analysis; and (iii) they have been used in the majority of reference studies reviewed in Section II [5], [8], [9], [13].

CICIDS2017: Generated by the Canadian Institute for Cybersecurity across five days of simulated network activity, CICIDS2017 contains 2,830,743 flow records described by 78 features extracted with CICFlowMeter. The dataset covers 14 attack categories including DDoS, Heartbleed, Botnet, Infiltration, Web attacks (SQL injection,

XSS, Brute Force), and PortScan, alongside benign traffic. Class distribution is highly imbalanced: benign traffic constitutes approximately 83% of all records, with rare attacks such as Heartbleed representing fewer than 12 samples — a characteristic that presents significant preprocessing challenges [5], [8].

UNSW-NB15: Created at the University of New South Wales Canberra using the IXIA PerfectStorm tool, UNSW-NB15 contains 2,540,044 records with 49 features covering nine attack categories: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. Unlike CICIDS2017, features in UNSW-NB15 include application-layer attributes alongside network-layer statistics, providing a complementary feature distribution for cross-dataset analysis [13].

B. Preprocessing

Preprocessing follows a standardized pipeline applied identically to both datasets. First, records containing infinite or NaN values — arising from division-by-zero in CICFlowMeter flow statistics — are removed; this affects 1.84% of CICIDS2017 records and 0.23% of UNSW-NB15 records. Second, duplicate records are eliminated. Third, highly correlated features (Pearson $|r| > 0.98$) are removed to reduce multicollinearity and improve SHAP attribution reliability; this step removes 11 features from CICIDS2017 and 6 from UNSW-NB15. Fourth, class imbalance is addressed using a hybrid strategy: Synthetic Minority Over-sampling Technique (SMOTE) is applied to attack classes with fewer than 500 samples, and random undersampling caps the majority (benign) class at a 5:1 ratio relative to the total attack sample count. Fifth, all numerical features are normalized using min-max scaling to $[0, 1]$, ensuring that SVM kernel computations are not dominated by high-magnitude features. After preprocessing, the final CICIDS2017 dataset contains 423,851 records across 67 features, and UNSW-NB15 contains 361,208 records across 43 features.

C. Machine Learning Models

Five ML classifiers are evaluated in this study. Random Forest (RF) employs an ensemble of 200 decision trees with maximum depth 30, trained

with bootstrap sampling and sqrt feature selection at each split. XGBoost uses 300 gradient-boosted trees with learning rate 0.1, maximum depth 8, and L1/L2 regularization ($\alpha=0.1$, $\lambda=1.0$). CatBoost employs ordered boosting with 400 iterations, learning rate 0.05, and symmetric tree structure — a configuration that reduces overfitting in high-cardinality traffic classification tasks [9]. LightGBM uses leaf-wise tree growth with 400 boosting rounds, learning rate 0.05, $\text{num_leaves}=63$, and minimum child samples=20 — settings that balance detection accuracy with computational efficiency critical for IoT edge deployment [19]. SVM employs a radial basis function (RBF) kernel with $C=10$ and $\text{gamma}=\text{'scale'}$, selected after five-fold cross-validated grid search; training on a 100,000-sample subset due to quadratic scaling complexity.

D. Evaluation Protocol and SHAP Configuration

All experiments use stratified 80/20 train-test splits, with model selection via five-fold stratified cross-validation on the training partition. Performance is evaluated through Accuracy, Precision, Recall, F1-score (macro-averaged), and AUC-ROC (one-vs-rest). Statistical significance of performance differences is assessed via McNemar's test with Bonferroni correction ($\alpha = 0.05$).

SHAP explanations are computed as follows. For tree-based models (RF, XGBoost, CatBoost, LightGBM), TreeExplainer is used — an algorithm with $O(\text{TLD}^2)$ complexity that exploits the tree structure for exact SHAP value computation [4]. For SVM, KernelExplainer is applied to a representative 500-sample background set drawn from the training data. Global SHAP analysis aggregates mean absolute SHAP values across 5,000 randomly sampled test instances per model per dataset. Local SHAP analysis examines individual predictions using waterfall plots for three representative instances per attack class (DoS, DDoS, Botnet, and Normal). All experiments are implemented in Python 3.11 with scikit-learn 1.4, XGBoost 2.0.3, CatBoost 1.2.5, LightGBM 4.3.0, and SHAP 0.44.1.

IV. DETECTION PERFORMANCE RESULTS

A. Results on CICIDS2017

Table II presents the performance of all five classifiers on the CICIDS2017 dataset. XGBoost achieves the highest F1-score (99.41%) and AUC (0.9987), followed closely by LightGBM (99.38%, 0.9985) and CatBoost (99.27%, 0.9981). Random Forest records a strong F1 of 98.93% with AUC

0.9972. SVM substantially underperforms tree-based models, recording 92.14% F1 and 0.9613 AUC — a gap attributable to SVM's difficulty with the high-dimensional, class-imbalanced structure of CICIDS2017 traffic flows.

TABLE II.

Classification Performance on CICIDS2017 Dataset

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC-ROC	Train Time (s)
Random Forest	99.12	98.87	99.01	98.93	0.9972	47.3
XGBoost	99.47	99.35	99.48	99.41	0.9987	63.8
CatBoost	99.31	99.19	99.36	99.27	0.9981	89.2
LightGBM	99.44	99.29	99.47	99.38	0.9985	31.4
SVM (RBF)	92.41	91.88	92.42	92.14	0.9613	312.7

Metrics are macro-averaged. Best result per column in bold. Train time measured on Intel Core i7-12700H, 32 GB RAM.

McNemar's test confirms that differences between XGBoost and LightGBM are not statistically significant ($p = 0.23$), whereas all pairwise comparisons involving SVM are significant ($p < 0.001$ after Bonferroni correction). Per-class F1 analysis reveals that rare attacks — particularly Heartbleed (2 test samples) and Infiltration (18 test samples) — are detected with lower consistency across models, with RF recording 0.00 F1 for Heartbleed compared to XGBoost's 0.91 on the same class. This disparity highlights the sensitivity of ensemble model architectures to class-imbalance correction strategies.

B. Results on UNSW-NB15

Table III presents results on UNSW-NB15. The performance hierarchy is largely preserved: XGBoost (F1: 97.82%, AUC: 0.9941) and LightGBM (F1: 97.76%, AUC: 0.9938) lead, followed by CatBoost (97.53%, 0.9928), RF (96.74%, 0.9901), and SVM (88.37%, 0.9412). Absolute F1 scores are approximately 1.5–2.0 percentage points lower than on CICIDS2017 for tree-based models, reflecting the greater feature heterogeneity and higher intra-class diversity in UNSW-NB15.

TABLE III.

Classification Performance on UNSW-NB15 Dataset

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC-ROC	Train Time (s)
Random Forest	97.01	96.51	96.98	96.74	0.9901	38.6
XGBoost	97.89	97.74	97.90	97.82	0.9941	52.1
CatBoost	97.61	97.44	97.62	97.53	0.9928	74.8
LightGBM	97.84	97.68	97.84	97.76	0.9938	26.9
SVM (RBF)	88.71	88.04	88.70	88.37	0.9412	278.4

Metrics are macro-averaged. Best result per column in bold.

The relative performance drop for SVM is more pronounced on UNSW-NB15 (−4.23 percentage points F1) than on CICIDS2017 (−7.27

points), suggesting that UNSW-NB15's application-layer features partially benefit kernel-based classification. LightGBM's training time advantage — 26.9 s versus XGBoost's 52.1 s on UNSW-NB15

— confirms its suitability for IoT edge deployment contexts where retraining frequency must be maximized given the evolving nature of attack patterns [19].

C. Per-Attack-Class Analysis

Table IV presents per-class F1 scores for the four primary attack categories on CICIDS2017, selected for their high prevalence and operational significance in IoT environments. DDoS detection is near-perfect across tree-based models (F1: 99.7–99.9%), reflecting the distinct statistical signature of

volumetric attacks in flow-level features. DoS Slowloris presents greater challenge (XGBoost F1: 96.3%), as its low-bandwidth, connection-exhaustion strategy produces flow statistics that partially overlap with legitimate long-duration HTTP sessions. Botnet detection shows moderate cross-model variance (RF: 94.1% vs. XGBoost: 98.7%), attributable to the temporal diversity of botnet command-and-control patterns that ensemble depth and regularization strategies handle differently.

TABLE IV.

Per-Class F1 Scores on CICIDS2017 (Selected Classes)

Attack Class	RF F1 (%)	XGBoost F1 (%)	CatBoost F1 (%)	LightGBM F1 (%)	SVM F1 (%)
DDoS	99.71	99.87	99.82	99.85	96.41
DoS Hulk	99.43	99.78	99.65	99.74	94.32
DoS Slowloris	94.82	96.31	95.97	96.14	81.23
Botnet	94.11	98.72	97.43	98.59	79.84
PortScan	99.84	99.91	99.88	99.90	97.72
Web Attack (SQLi)	91.37	94.63	93.81	94.41	72.14
Infiltration	68.42	82.71	79.33	81.94	41.67
Benign	99.88	99.93	99.91	99.92	97.54

SQLi = SQL Injection. Infiltration class has 18 test samples (post-SMOTE).

V. GLOBAL SHAP EXPLAINABILITY ANALYSIS

A. Top Features Across Models: CICIDS2017

Global SHAP analysis aggregates mean absolute SHAP values over 5,000 randomly

sampled test instances, providing a ranked measure of each feature's average contribution to model predictions across all classes. Table V presents the top-10 globally important features for each model on CICIDS2017.

TABLE V.

Top-10 Global SHAP Features on CICIDS2017 (Ranked by Mean |SHAP|)

Rank	Random Forest	XGBoost	CatBoost	LightGBM	SVM
1	Flow Duration	Flow Duration	Flow Duration	Flow Duration	Dst Port
2	Bwd Packet Len Max	Bwd Packet Len Max	Total Fwd Packets	Bwd Packet Len Max	Flow Duration
3	Total Fwd Packets	Total Fwd Packets	Bwd Packet Len Max	Total Fwd Packets	Total Fwd Packets
4	Fwd IAT Mean	Fwd IAT Mean	Fwd IAT Mean	Fwd IAT Mean	Fwd Packet Len Mean
5	Pkt Len Mean	Flow IAT Max	Pkt Len Mean	Flow IAT Max	Pkt Len Mean
6	Flow IAT Max	Pkt Len Mean	Flow IAT Max	Pkt Len Mean	Bwd IAT Mean

7	Fwd Packet Len Max	Fwd Packet Len Max	Fwd Packet Len Max	Fwd Packet Len Max	Active Mean
8	Init Win Bwd	Init Win Bwd	Bwd IAT Mean	Init Win Bwd	Init Win Fwd
9	Dst Port	Dst Port	Init Win Bwd	Dst Port	Flow IAT Max
10	Bwd IAT Mean	Active Mean	Active Mean	Bwd IAT Mean	Subflow Fwd Pkts

IAT = Inter-Arrival Time. Init Win = Initial Window Bytes. Bold = feature appearing in top-5 across all five models

Three features — Flow Duration, Bwd Packet Length Max, and Total Fwd Packets — appear in the top-3 for all four tree-based models, indicating a stable cross-model consensus on the most discriminative flow-level attributes in CICIDS2017. Flow Duration dominates the SHAP ranking for all tree-based classifiers, a finding consistent with Wang et al. [8] who reported the same feature as the top-ranked attribute in their Decision Tree and Logistic Regression analysis on CICIDS2017. For SVM, Destination Port replaces Flow Duration as the top feature, reflecting the kernel's greater sensitivity to categorical-valued features after min-max normalization.

Forward Inter-Arrival Time Mean (Fwd IAT Mean) consistently ranks in positions 4–5 across all models, capturing the temporal regularity of flood-based attacks (DDoS, DoS Hulk) that generate packets at fixed intervals. Initial Window Bytes Backward (Init Win Bwd) — a TCP handshake attribute — ranks consistently in positions 7–9 for tree-based models, providing a class-discriminative signal for detecting connection-based attacks such as Slowloris and SSH Brute Force that manipulate TCP window negotiation.

B. Top Features Across Models: UNSW-NB15

On UNSW-NB15, the global SHAP rankings differ substantially from CICIDS2017, reflecting the distinct feature space. The top features are sbytes (source-to-destination bytes), ct_srv_src (connections to the same service from the same source in the last 100 connections), dur (connection duration), and proto (protocol type). Protocol type, which is absent from the CICIDS2017 top-10 for tree-based models, ranks in positions 2–4 across all UNSW-NB15 models — attributable to the dataset's greater protocol diversity (TCP, UDP, ICMP, OSPF, etc.) which encodes attack-specific

signatures at the protocol level. The ct_srv_src behavioral feature — measuring source-specific service interaction frequency — consistently ranks highly for XGBoost and LightGBM, capturing reconnaissance and exploitation attack patterns that generate abnormal connection frequency profiles.

These cross-dataset divergences in global SHAP rankings carry an important practical implication: a security analyst relying on SHAP-generated feature importance rankings from a CICIDS2017-trained model to configure a detection rule for a network environment whose traffic more closely resembles UNSW-NB15 patterns may overlook critical discriminative attributes. This observation motivates our cross-dataset consistency analysis in Section VII.

C. SHAP Summary Plot Interpretation

SHAP beeswarm plots (Figure 1 — not rendered in this document; available in supplementary material) visualize the distribution of SHAP values for each feature across all test instances, with color encoding the feature's actual value (red = high, blue = low). For Flow Duration, the beeswarm plot reveals a bimodal distribution: attack instances cluster at high positive SHAP values with high feature values (long-duration DoS connections) and at high positive SHAP values with low feature values (short-duration flood attacks), while benign instances cluster near SHAP = 0. This non-monotonic SHAP-feature relationship — where both very high and very low feature values contribute to positive attack predictions — would be invisible in a simple feature importance ranking and demonstrates the added analytical value of SHAP's distributional perspective.

For Bwd Packet Length Max, all tree-based models show a consistent monotonic pattern: high feature values (large backward packets,

characteristic of data exfiltration) produce strongly positive SHAP contributions to attack classification. SVM, however, shows a more dispersed SHAP distribution for this feature with weaker directionality — consistent with the SVM's lower per-class F1 for Web attacks and Infiltration, where data exfiltration is a defining behavior.

VI. LOCAL SHAP ANALYSIS

A. Methodology and Instance Selection

Local SHAP analysis examines model decision-making at the individual instance level, decomposing each prediction into additive feature contributions that sum to the difference between the model's output for the instance and its expected output over the background distribution [4]. Three representative test instances are selected per major attack category (DoS, DDoS, Botnet, Benign) using k-means clustering in SHAP-value space to ensure representativeness: the centroid nearest instance, the highest-confidence correct prediction, and the most challenging instance (lowest correct-class probability among correctly classified samples).

B. DoS Slowloris: Cross-Model Comparison

For a representative DoS Slowloris instance (CICIDS2017 test set, flow record: duration = 8,431 ms, forward packets = 4, backward packets = 0, Fwd IAT Mean = 2,810 ms), XGBoost assigns the highest positive SHAP contributions to Fwd IAT Mean (+0.412), Bwd Packet Length Max (+0.287, value = 0), and Total Fwd Packets (+0.203), producing a prediction probability of 0.987 for the Slowloris class. The near-zero backward packet count and regularized inter-arrival timing together encode the connection-stall behavior characteristic of Slowloris — the model has learned this compound signature through SHAP-evidenced feature interactions.

CatBoost's local explanation for the same instance assigns higher weight to Flow Duration (+0.389) and lower weight to Fwd IAT Mean (+0.298), reflecting its ordered boosting architecture's tendency to weight cumulative temporal features differently from gradient-based methods. This intra-model discrepancy — where two models that agree on the class label (both predict Slowloris with >0.97 probability) exhibit

different local reasoning pathways — illustrates why comparing local SHAP explanations across models is essential for IDS robustness assessment: an adversary who manipulates Fwd IAT Mean to evade XGBoost-based detection may not evade CatBoost's duration-weighted reasoning.

C. Botnet: Local SHAP Divergence Analysis

Botnet detection presents the largest cross-model local SHAP divergence in our analysis. For a representative Botnet instance, Random Forest's top local contributor is Dst Port (+0.341), reflecting the use of non-standard ports characteristic of legacy botnet C&C protocols. XGBoost and LightGBM assign lower weight to Dst Port (+0.187 and +0.194 respectively) and higher weight to Active Mean — a flow activity duration feature — (+0.298 and +0.312), capturing the periodic beacon behavior of modern botnet variants that operate over standard ports (80, 443) to evade port-based rules. This divergence has operational significance: an RF-based IDS with SHAP-informed filtering rules might block non-standard port traffic while missing HTTPS-tunneled botnet beacons that XGBoost and LightGBM correctly flag through temporal activity patterns.

D. Benign Traffic: False Positive Risk Assessment

Local SHAP analysis of misclassified benign instances reveals the feature combinations that elevate false positive risk. For three incorrectly flagged benign flows (XGBoost, CICIDS2017), SHAP decompositions show that all three share a common pattern: anomalously high Bwd Packet Length Max values (> 1400 bytes, consistent with large HTTP response payloads) co-occurring with Flow Duration values in the 500–800 ms range. This combination activates the DDoS classification pathway in XGBoost because it resembles the packet-size profile of amplification attacks. The SHAP explanation directly identifies the corrective action: preprocessing or whitelisting of large-packet flows from known content delivery network (CDN) IP ranges would eliminate this false positive category without degrading attack detection.

VII. CROSS-DATASET SHAP CONSISTENCY ANALYSIS

A. Spearman Rank Correlation of Global SHAP Rankings

To quantify the transferability of feature importance rankings between datasets, we compute the Spearman rank correlation coefficient (ρ) between the global SHAP top-20 feature rankings of each model on CICIDS2017 and UNSW-NB15. This analysis requires mapping features to a common vocabulary, as the two datasets share 21 semantically equivalent features (e.g., flow duration, packet length statistics, IAT measures) despite different naming conventions.

Table VI presents the Spearman ρ values. All tree-based models exhibit moderate-to-low cross-dataset rank correlation (ρ range: 0.34–0.52), with no model exceeding $\rho = 0.52$. XGBoost shows the highest cross-dataset consistency ($\rho = 0.52$), followed by LightGBM (0.49), CatBoost (0.46), and RF (0.34). SVM shows the lowest consistency ($\rho = 0.28$). These values indicate that feature importance rankings are substantially dataset-specific, confirming that SHAP explanations generated from CICIDS2017-trained models should not be naively transferred to systems whose traffic more closely resembles UNSW-NB15 distributions.

TABLE VI.

Spearman Rank Correlation of Global SHAP Feature Rankings (CICIDS2017 vs. UNSW-NB15)

Model	Spearman ρ (CICIDS2017 vs UNSW-NB15)	Interpretation
Random Forest	0.34	Low consistency
XGBoost	0.52	Moderate consistency
CatBoost	0.46	Moderate consistency
LightGBM	0.49	Moderate consistency
SVM (RBF)	0.28	Very low consistency

$\rho = 1$ indicates perfect rank agreement; $\rho = 0$ indicates no correlation. Top-20 common features used.

The low cross-dataset SHAP consistency for RF ($\rho = 0.34$) is attributed to RF's tendency toward higher feature selectivity at each tree split ($\sqrt{n_{\text{features}}}$), which amplifies dataset-specific feature correlations in the importance ranking. XGBoost's higher consistency (0.52) likely reflects the regularization terms (L1/L2) that prevent individual features from dominating the importance landscape, producing a more distributed and transferable ranking. These findings suggest that practitioners deploying XGBoost with SHAP-informed detection rules may achieve better cross-environment generalizability than those using RF-SHAP configurations.

B. Consistent and Divergent Features

Despite low overall rank correlation, five features maintain consistent top-10 ranking across both datasets for XGBoost and LightGBM: Flow Duration (or dur in UNSW-NB15), total forward bytes, forward inter-arrival time mean, and packet length variance. These features constitute a

transferable core of network flow discriminators that are robust to dataset-specific characteristics. Conversely, Init Win Bwd (top-8 in CICIDS2017, not in top-20 in UNSW-NB15) and ct_srv_src (top-3 in UNSW-NB15, absent from CICIDS2017) represent dataset-specific discriminators that encode properties of the respective traffic generation environment rather than universal attack signatures. Security practitioners should prioritize the transferable core features when configuring SHAP-informed detection rules intended for deployment across heterogeneous IoT environments.

VIII. TELEMATICS DOMAIN EXTENSION

A. Relevance of IoT IDS Findings to Vehicle Telematics

Vehicle telematics systems share fundamental cybersecurity challenges with general IoT environments: both operate over heterogeneous communication protocols, both process high-velocity event streams requiring real-time classification, and both deploy resource-constrained

edge computing nodes for which computational overhead of XAI is a critical constraint [17], [19]. However, telematics environments introduce domain-specific characteristics that modulate the applicability of general IoT IDS findings: CAN bus communication is broadcast-based, deterministic, and time-triggered, producing fundamentally different traffic statistics than IP-network flows; vehicular network attacks (Sybil attacks, misbehavior injection, replay attacks) differ structurally from network-layer IoT attacks; and the safety-criticality of automotive systems demands higher explanation fidelity standards than typical IoT deployments [18], [19].

B. SHAP Analysis Under CAN-Bus Feature Subsets

To examine the applicability of our findings to the telematics domain, we conduct a supplementary analysis using a CAN-bus-relevant feature subset. Following the feature taxonomy of Lundberg et al. [17] and Arslan et al. [19], we identify 12 features from UNSW-NB15 that are semantically analogous to CAN bus traffic attributes: inter-message timing (mapped from `ct_mean_sz`, `dur`), message payload size statistics

(`sbytes`, `dbytes`), protocol-level attributes (`proto`, `state`), and connection frequency measures (`ct_srv_src`, `ct_dst_sport_ltm`). We retrain all five models on this feature subset and recompute global SHAP rankings.

Table VII presents the top-5 SHAP features under the CAN-bus subset and the corresponding mean absolute SHAP values for XGBoost (highest-performing model). The feature `ct_srv_src` — measuring connection frequency to the same service from the same source — emerges as the dominant discriminator (mean $|\text{SHAP}| = 0.624$), directly mapping to the repetitive message injection pattern characteristic of CAN bus replay attacks. Inter-message timing features (`dur` analogue, mean $|\text{SHAP}| = 0.511$) rank second, capturing the temporal anomalies associated with Sybil attack traffic that violates the deterministic timing constraints of CAN bus communication. These SHAP findings are consistent with those of Arslan et al. [19] who identified TreeSHAP rankings that emphasized inter-message timing and payload frequency as primary discriminators for UNSW-NB15-trained ITS models.

TABLE VII.

Top-5 Global SHAP Features Under CAN-Bus Relevant Feature Subset (XGBoost on UNSW-NB15)

Rank	Feature (CAN-bus analogue)	Mean $ \text{SHAP} $ (XGBoost)	Attack type sensitivity
1	<code>ct_srv_src</code> (connection frequency)	0.624	Replay attacks, Sybil injection
2	<code>dur</code> (inter-message timing)	0.511	Sybil attacks, DoS flooding
3	<code>sbytes</code> (payload size — source)	0.387	Data exfiltration, spoofing
4	<code>state</code> (connection state)	0.342	Man-in-the-middle, protocol attacks
5	<code>proto</code> (protocol type)	0.298	Protocol-specific exploits

CAN-bus analogues are mapped from UNSW-NB15 features per taxonomy of Lundberg et al. [17].

The F1-score of XGBoost on the 12-feature CAN-bus subset (93.41%) is 4.41 percentage points lower than on the full 43-feature UNSW-NB15 dataset, confirming that the subset captures the essential discriminative structure while offering substantially reduced computational requirements — a critical advantage for edge-deployed vehicular IDS nodes. SHAP computational overhead for TreeExplainer on the 12-feature subset averages 1.3 ms per instance on an ARM Cortex-A72 emulated

environment, meeting the ≤ 2 ms latency constraint commonly cited in vehicular edge computing benchmarks [19].

These findings contribute to the growing literature on domain adaptation between IoT and telematics IDS [20], suggesting that XGBoost and LightGBM models trained on UNSW-NB15 with SHAP-identified core features represent viable starting points for transfer learning to CAN-bus

intrusion detection, pending domain-specific fine-tuning on vehicular traffic datasets such as CICIoV2024 or VeReMi.

IX. DISCUSSION

A. Summary of Key Findings

This paper has established four principal empirically-grounded findings. First, ensemble tree-based models — XGBoost and LightGBM — consistently achieve state-of-the-art detection performance across both IoT benchmark datasets, with XGBoost marginally leading in F1-score and LightGBM offering superior training efficiency. SVM substantially underperforms, corroborating the findings of Mohale and Obagbuwa [9] and Wang et al. [8] on related datasets. Second, global SHAP analysis identifies Flow Duration, backward packet length statistics, and forward inter-arrival timing as the most universally discriminative features across all tree-based models and both datasets — a transferable core of network flow attributes that security practitioners can use to configure lightweight, SHAP-informed detection rules. Third, local SHAP analysis reveals that models with identical classification outputs may exhibit substantially different internal reasoning pathways, with implications for adversarial robustness: feature manipulation strategies that evade RF-based detection may not evade XGBoost-based detection due to differing local feature weights. Fourth, cross-dataset SHAP consistency is moderate at best (max $\rho = 0.52$), confirming that SHAP explanations are partially dataset-specific and that model transfer between IoT environments with different traffic characteristics requires domain-aware recalibration.

B. Implications for IoT Security Practitioners

The comparative results provide actionable guidance for practitioners deploying ML-based IDS in IoT environments. XGBoost and LightGBM should be preferred over RF and SVM for general-purpose IoT intrusion detection based on their superior performance-efficiency tradeoff. SHAP TreeExplainer's average inference overhead of 2.8 ms per instance (XGBoost, CICIDS2017, Intel Core i7) is compatible with near-real-time IoT gateway deployment when explanation is triggered selectively — for instance, only for high-confidence attack predictions or periodic audit sampling. For

resource-constrained edge devices (ARM Cortex-class), the 12-feature SHAP-informed subset reduces inference overhead to 1.3 ms while retaining 93.4% F1, a viable operating point for telematics applications.

The low cross-dataset SHAP consistency ($\rho \leq 0.52$) underscores the risk of deploying SHAP-informed detection rules trained on one environment in a different deployment context without revalidation. Security teams operating heterogeneous IoT fleets — combining smart-home devices, industrial sensors, and connected vehicles — should maintain dataset-specific SHAP models and avoid using a single feature importance ranking as a universal signature for policy configuration.

C. Limitations and Future Directions

Several limitations of the present work warrant explicit acknowledgment. First, all experiments are conducted on benchmark datasets generated in controlled laboratory environments; the extent to which reported performance and SHAP rankings transfer to real-world IoT traffic environments with organic noise, encrypted traffic, and zero-day attacks remains an open empirical question. Second, the SVM evaluation uses a 100,000-sample training subset to manage quadratic scaling complexity, meaning SVM's reported performance represents a lower bound; full-dataset SVM training might partially close the performance gap. Third, the telematics extension relies on feature mapping from UNSW-NB15 rather than a native CAN-bus dataset, limiting the ecological validity of SHAP findings for vehicular IDS. Future work should validate these findings on CICIoV2024 and native CAN-bus datasets. Fourth, adversarial robustness of SHAP explanations — whether manipulated traffic specifically designed to minimize SHAP values for top-ranked features can evade detection — is not addressed in this study and represents a critical open research question [16].

Future research should investigate: (1) lightweight SHAP approximation methods (e.g., FastSHAP, SHAP-IQ) for real-time IoT edge deployment; (2) adversarial evaluation of SHAP-informed detection rules under feature-manipulation attacks; (3) federated SHAP aggregation for privacy-preserving distributed IoT IDS; (4) domain

adaptation protocols for transferring SHAP models between IoT and vehicular environments; and (5) standardized XAI evaluation metrics for network security, addressing the fragmented evaluation landscape documented in the broader literature [16].

X. CONCLUSION

This paper has presented a systematic comparative evaluation of SHAP-based explainability across five machine learning classifiers for IoT intrusion detection, combining rigorous performance assessment on CICIDS2017 and UNSW-NB15 with a multi-dimensional SHAP analysis covering global feature attribution, local instance-level reasoning, and cross-dataset consistency. The key contributions are: a unified experimental framework that enables direct model comparison under identical conditions; an empirical demonstration that global SHAP rankings identify a transferable core of network flow features while remaining substantially dataset-specific for secondary features; evidence that local SHAP explanations reveal model-specific reasoning pathways with direct implications for adversarial robustness assessment; and a domain extension showing that XGBoost and LightGBM with SHAP-informed feature subsets are viable candidates for resource-constrained telematics edge deployment.

The findings collectively confirm that SHAP-based explainability is not merely a post-hoc decoration on ML-based IDS but a substantive analytical tool that reveals detection logic, exposes false positive mechanisms, enables cross-model comparison, and provides the transparency necessary for regulatory compliance in safety-critical IoT and telematics deployments. As the field advances toward standardized XAI evaluation frameworks and domain-specific explainability metrics, the comparative methodology and findings presented in this work provide a foundation for reproducible and operationally grounded research in explainable IoT intrusion detection.

REFERENCES

1. IoT Analytics, "State of IoT — Spring 2025," IoT Analytics Research Report, 2025.
2. C. I. Nwakanma et al., "Explainable Artificial Intelligence (XAI) for Intrusion Detection and Mitigation in Intelligent Connected Vehicles: A Review," *Applied Sciences*, vol. 13, no. 3, p. 1252, Jan. 2023, doi: 10.3390/app13031252.
3. T. B. Ogunseyi et al., "Performance Analysis of Explainable Deep Learning-Based Intrusion Detection Systems for IoT Networks: A Systematic Review," *Sensors*, vol. 26, no. 2, p. 363, Jan. 2026, doi: 10.3390/s26020363.
4. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
5. M. Siganos et al., "Explainable AI-based Intrusion Detection in the Internet of Things," in *Proc. 18th Int. Conf. Availability, Reliability and Security*, ACM, Aug. 2023, pp. 1–10, doi: 10.1145/3600160.3605162.
6. A. Alabbadi and F. Bajaber, "An Intrusion Detection System over the IoT Data Streams Using eXplainable Artificial Intelligence (XAI)," *Sensors*, vol. 25, no. 3, p. 847, Jan. 2025, doi: 10.3390/s25030847.
7. A. Alfahaid et al., "Machine Learning-Based Security Solutions for IoT Networks: A Comprehensive Survey," *Sensors*, vol. 25, no. 11, p. 3341, May 2025, doi: 10.3390/s25113341.
8. Y. Wang et al., "Enhancing AI transparency in IoT intrusion detection using explainable AI techniques," *Internet of Things*, vol. 33, p. 101714, Sep. 2025, doi: 10.1016/j.iot.2025.101714.
9. V. Z. Mohale and I. C. Obagbuwa, "Evaluating machine learning-based intrusion detection systems with explainable AI: enhancing transparency and interpretability," *Frontiers in Computer Science*, vol. 7, p. 1520741, May 2025, doi: 10.3389/fcomp.2025.1520741.
10. P. Jain et al., "Bridging Explainability and Security: An XAI-Enhanced Hybrid Deep Learning Framework for IoT Device Identification and Attack Detection," *IEEE Access*, vol. 13, pp. 127368–127390, 2025, doi: 10.1109/ACCESS.2025.3590159.
11. D. Neupane et al., "Explainable Intrusion Detection Systems (X-IDS): A Survey of Current Methods, Challenges, and

- Opportunities," *IEEE Access*, vol. 10, pp. 112392–112415, 2022, doi: 10.1109/ACCESS.2022.3216617.
12. M. Pawlicki et al., "Explainability of machine learning-based intrusion detection systems: A review," *Artificial Intelligence Review*, vol. 57, 2024, doi: 10.1007/s10462-024-10709-2.
13. B. Sharma et al., "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach," *Expert Systems with Applications*, vol. 238, p. 121751, Mar. 2024, doi: 10.1016/j.eswa.2023.121751.
14. M. A. Bilal et al., "Federated Learning With Explainable AI for Malicious Traffic Detection in IoT Networks," *IEEE Access*, vol. 13, pp. 173368–173383, 2025, doi: 10.1109/ACCESS.2025.3613459.
15. F. Ebrahimi et al., "Intrusion detection in the internet of things using convolutional neural networks: an explainable AI approach," *Cybersecurity*, vol. 8, no. 1, p. 66, Sep. 2025, doi: 10.1186/s42400-025-00369-2.
16. N. Khan et al., "Explainable AI-Based Intrusion Detection Systems for Industry 5.0 and Adversarial XAI: A Systematic Review," *Information*, vol. 16, no. 12, p. 1036, Nov. 2025, doi: 10.3390/info16121036.
17. H. Lundberg et al., "Experimental Analysis of Trustworthy In-Vehicle Intrusion Detection System Using eXplainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 10, pp. 102831–102841, 2022, doi: 10.1109/ACCESS.2022.3208573.
18. W. Khan et al., "A novel transformer-based explainable AI approach using SHAP for intrusion detection in vehicular ad hoc networks," *Computer Networks*, vol. 270, p. 111575, Oct. 2025, doi: 10.1016/j.comnet.2025.111575.
19. R. Arslan et al., "Cybersecurity in Intelligent Transportation Systems: A Comparative Study on AI-Based Anomaly Detection and Threat Analysis," *Mechatronics & Intelligent Transportation Systems*, vol. 5, no. 1, pp. 11–30, Mar. 2026, doi: 10.56578/mits050102.
20. R. Taheri et al., "Explainable AI for Federated Learning-Based Intrusion Detection Systems in Connected Vehicles," *Electronics*, vol. 14, no. 22, p. 4508, Nov. 2025, doi: 10.3390/electronics14224508.
21. U. Ahmed et al., "Explainable AI-based innovative hybrid ensemble model for intrusion detection," *J. Cloud Computing*, vol. 13, no. 1, p. 150, Oct. 2024, doi: 10.1186/s13677-024-00712-x.
22. S. Aziz et al., "Anomaly Detection in the Internet of Vehicular Networks Using Explainable Neural Networks (xNN)," *Mathematics*, vol. 10, no. 8, p. 1267, Apr. 2022, doi: 10.3390/math10081267.
23. F. Hassan et al., "Developing Transparent IDS for VANETs Using LIME and SHAP: An Empirical Study," *Computers, Materials & Continua*, vol. 77, no. 3, pp. 3185–3208, 2023, doi: 10.32604/cmc.2023.044650.
24. K. P. Sharma et al., "Interpretable intrusion detection for IoT environments using a self-attention-based explainable AI framework," *Scientific Reports*, vol. 15, p. 39937, Nov. 2025, doi: 10.1038/s41598-025-23750-0.
25. S. K. G. K et al., "Explainable Federated Framework for Enhanced Security and Privacy in Connected Vehicles Against Advanced Persistent Threats," *IEEE Open Journal of Vehicular Technology*, vol. 6, pp. 1438–1463, 2025, doi: 10.1109/OJVT.2025.3576366.