

Network Dominance and the MITM Exception: A Modality-Aware Analysis of IIoT Intrusion Detection on the DataSense Benchmark

Okoi Michael Obeten

michealobeten@auchiipoly.edu.ng

Auchi Polytechnic

Omowumi Hafsat Aliu

Auchi Polytechnic

Research Article

Keywords: Industrial Internet of Things, intrusion detection system, modality analysis, ensemble learning, SHAP, cyber-physical security, DataSense, XGBoost

Posted Date: August 26th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-9889062/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Network Dominance and the MITM Exception: A Modality-Aware Analysis of IIoT Intrusion Detection on the DataSense Benchmark

Okoi Michael Obeten^{1*} and Omowumi Hafsai Aliu^{1†}

^{1*}Computer Science, Auchi Polytechnic, Auchi, Nigeria.

*Corresponding author(s). E-mail(s): michealobeten@auchipoly.edu.ng;

Contributing authors: aliuomowumi@auchipoly.edu.ng;

[†]These authors contributed equally to this work.

Abstract

The Industrial Internet of Things (IIoT) is rapidly expanding the attack surface of critical infrastructure, and intrusion detection systems (IDSs) are increasingly relied upon as the principal automated defence. Recent benchmark datasets such as DataSense provide synchronized network traffic and device telemetry to enable cross-modal IDS research, and a growing body of work assumes that combining these modalities improves detection. However, the per-attack-category contribution of each modality has not been empirically tested on this newest generation of IIoT benchmarks. This paper presents a modality-aware analysis of IIoT intrusion detection on DataSense, partitioning the 84 pre-extracted features into 76 network and 8 telemetry channels (reduced to 64 network and 7 telemetry features after removal of leakage-prone string-valued columns) and constructing three feature configurations: network-only, telemetry-only, and fused. We train Random Forest, XGBoost, LightGBM, and Support Vector Machine classifiers under each configuration for binary, seven-class, and fine-grained eighty-four-class classification, producing 36 experimental conditions. Results show that the network channel alone is nearly sufficient: XGBoost on network-only features achieves 0.9890 macro F1 on binary detection and 0.9819 on the seven-class task, within 0.6 percentage points of the fused configuration in every category. SHAP attribution confirms that network features account for over 99 percent of explanatory weight in six of seven attack categories, with man-in-the-middle (MITM) being the single exception at approximately 10 percent telemetry contribution. The dominance result holds across both 1-second and 10-second window granularities. Practical implication: for cost-constrained IIoT monitoring, the 8

telemetry features as currently engineered in DataSense add minimal value over the network channel, except when MITM-class attacks must be distinguished.

Keywords: Industrial Internet of Things, intrusion detection system, modality analysis, ensemble learning, SHAP, cyber-physical security, DataSense, XGBoost

1 Introduction

The Industrial Internet of Things (IIoT) has become foundational to modern industrial automation, embedding sensors, actuators, edge gateways, and cloud services into manufacturing, energy, transportation, and process control [1, 2]. This pervasive connectivity yields operational and economic benefits but simultaneously broadens the cyber-attack surface in ways that traditional information-technology security mechanisms were not designed to address [3, 4]. The consequences of successful attacks on industrial cyber-physical systems range from data exfiltration and service disruption to physical safety incidents, motivating sustained research into automated defence mechanisms [2, 5].

Intrusion detection systems (IDSs) constitute the principal automated line of defence in this setting, and machine learning (ML) approaches have come to dominate the literature because of their ability to generalize across attack variants without requiring hand-crafted signature databases [6–8]. A substantial body of recent work has applied classical ensemble learners, deep learning architectures, and hybrid combinations to IoT and IIoT benchmark datasets, with reported accuracies routinely exceeding 95 percent on contemporary datasets such as CIC-IoT2023, Edge-IIoTset, and TON-IoT [9–12].

Alongside this methodological progress, benchmark datasets have become richer and more realistic. Early datasets focused exclusively on network traffic captured at routers or gateways [13]. More recent releases such as Edge-IIoTset [14] and CIC-IoT2023 [10] introduced larger device populations and a wider variety of attack scenarios. The most recent contribution in this line, DataSense, released by the Canadian Institute for Cybersecurity in 2025 [1], is positioned as the first benchmark to systematically capture synchronized sensor telemetry and network traffic across a controlled testbed of industrial sensors, IoT devices, and infrastructure equipment, executing 50 distinct attack scenarios across seven categories. The dataset description explicitly motivates this synchronization as enabling cross-modal IDS research where information from both channels can be combined.

A natural and frequently asserted hypothesis follows: that combining network and physical-or-telemetry observations should improve attack detection, particularly for attack categories whose effects are not fully captured at the network layer alone. This assumption is consistent with broader trends in multimodal learning and is taken up by a growing literature on cyber-physical fusion for IoT security [15–18]. However, the per-attack-category contribution of each modality has not, to the authors’ knowledge, been empirically and rigorously tested on the newest generation of synchronized IIoT benchmarks. Existing work tends either to report overall fusion performance without

disaggregation by attack type, or to propose new architectures for fusion without comparison against single-modality baselines.

This paper addresses that gap. We treat the cross-modal premise as a testable empirical claim and characterize, category by category, the contribution of the network and telemetry channels to attack detection on DataSense. We partition the dataset’s 84 pre-extracted features into two cleanly separated modalities by column prefix, drop a small set of high-cardinality string-valued columns to prevent leakage from explicit attacker addresses or ports, and construct three feature configurations: network-only, telemetry-only, and fused. We train four classical learners (Random Forest, XGBoost, LightGBM, and a Support Vector Machine with a radial basis function kernel) under each configuration across three classification granularities (binary, seven-class, and fine-grained eighty-four-class), yielding 36 experimental conditions. We then quantify the per-category contribution of each modality using a modality observability map that reports macro F1 under each configuration alongside the marginal lift of fusion over the better single modality. Finally, we interpret the dominant signal sources via SHAP attribution disaggregated by modality, and we verify robustness with a sensitivity analysis across two temporal aggregation windows (1-second and 10-second).

Our findings depart from the typical narrative in the cross-modal IIoT-IDS literature. The network channel alone is nearly sufficient across all seven attack categories on DataSense. Fusion provides essentially zero macro-F1 lift over network-only features in six of seven categories, with the seventh (web) gaining only 0.6 percentage points. SHAP attribution confirms that network-derived features account for over 99 percent of explanatory weight in six of seven categories. The single exception is the MITM category, where SHAP attributes approximately 10 percent of explanatory weight to the telemetry channel. These results suggest that the implicit assumption underlying the cross-modal premise does not hold uniformly across attack categories on this benchmark, and that for cost-constrained IIoT monitoring the telemetry channel as currently engineered in DataSense adds minimal information beyond what the network channel already carries.

The contributions of this paper are as follows:

- We present the first per-category modality-aware analysis of IIoT intrusion detection on the DataSense benchmark, testing the empirical validity of the cross-modal premise rather than assuming it.
- We introduce a modality observability map methodology that operationalizes the per-category contribution of each modality through controlled feature-configuration experiments and a derived modality-lift metric.
- We empirically demonstrate, across 36 experimental conditions and four model families, that network features alone are nearly sufficient for IIoT intrusion detection on DataSense, with MITM as the single category exhibiting meaningful telemetry contribution.
- We provide a mechanistic interpretation of the dominance pattern and the MITM exception, grounded in SHAP attribution disaggregated by modality and supported by the physical semantics of each attack category.
- We verify that the dominance finding is robust to temporal aggregation, reproducing the modality ordering across 1-second and 10-second window granularities.

The remainder of the paper is structured as follows. Section 2 reviews related work on IIoT intrusion detection datasets, ensemble machine learning approaches, cyber-physical and multimodal fusion, and explainable AI for IDS. Section 3 describes the DataSense dataset, the preprocessing pipeline, the modality partition, and the experimental design. Section 4 reports headline performance, the modality observability map, SHAP attribution, confusion matrices, inference latency, and window-size sensitivity. Section 5 interprets these results and discusses their implications for IIoT monitoring practice and benchmark design. Section 6 acknowledges limitations and identifies future work, and Section 7 concludes.

2 Related Work

2.1 IIoT Intrusion Detection Datasets

The development of realistic benchmark datasets has been a central enabler of progress in IoT and IIoT intrusion detection. Earlier benchmarks such as KDD-Cup99 and NSL-KDD were derived from non-IoT environments and have been widely criticized as failing to reflect contemporary IoT traffic patterns and protocol use [13]. More recent releases have addressed these shortcomings with progressively more realistic testbeds and attack scenarios. Ferrag et al. introduced Edge-IIoTset, a comprehensive IoT/IIoT dataset spanning 14 attack types organized into five threat categories and combining attacks against MQTT, CoAP, and HTTP/HTTPS protocols [14]. Neto et al. released CIC-IoT2023, a large-scale dataset capturing traffic from over 100 IoT devices with 33 distinct attack types organized into seven categories including Mirai variants, DDoS, DoS, reconnaissance, spoofing, web-based attacks, and brute force [10]. Belarbi et al. introduced Gotham Dataset 2025, focused on reproducibility through an emulated testbed of 78 IoT devices and a Tshark-based feature extraction pipeline [19].

In the IIoT-specific space, DataSense [1] was released by the Canadian Institute for Cybersecurity in 2025 with a distinctive emphasis on synchronized capture of sensor telemetry and network traffic in an industrial testbed. The dataset includes diverse industrial sensors (steam, vibration, weather, soil), commodity IoT devices (cameras), and infrastructure (broker, router, access point, edge gateway), and executes 50 attack scenarios spanning reconnaissance, DoS, DDoS, MITM, malware, web-based attacks, and brute force. Features are pre-extracted into time-windowed CSV files at granularities from one to ten seconds, with naming conventions that distinguish network-derived (**network_***) from log-derived (**log_***) features. This property is what makes DataSense uniquely suited to per-modality analysis without requiring custom feature engineering.

A subsequent contribution by the same authors, DeepSense [20], proposed an adaptive scalable ensemble framework on DataSense; however, that work focused on a hierarchical RuleSense/NeuroSense detection architecture rather than on quantifying the per-category contribution of the two modalities. The asymmetric-contribution question, which the present paper addresses, remains open in the literature.

2.2 Machine Learning Approaches for IIoT Intrusion Detection

Classical ensemble learners remain competitive with, and often outperform, deep neural alternatives on tabular IoT/IIoT IDS benchmarks. Almotairi et al. proposed a heterogeneous stacking classifier with embedded feature selection on IoT data and reported XGBoost as the dominant base learner in both binary and multi-class settings [6]. The 2024 study by Akif et al., reported in [21], compared multiple classifiers within a PySpark-based real-time IDS pipeline on the IoT-23 dataset and found XGBoost achieved the highest accuracy at 98.89 percent while Random Forest delivered the lowest inference latency at 31 milliseconds. Ibrahim and Al-Wadi evaluated Random Forest, XGBoost, AdaBoost, multi-layer perceptron, logistic regression, and k-nearest neighbours as base models for an ensemble Internet of Medical Things (IoMT) IDS and again identified XGBoost as the strongest individual learner [22]. Chaib et al. compared XGBoost, LightGBM, and Random Forest under Top-K feature selection on CIC-IoMT2024 and found Random Forest provided the best balance of accuracy and computational efficiency [23]. A recent IoT device security audit framework by Shaikhanova et al. [24] benchmarked Random Forest, LightGBM, XGBoost, and logistic regression on TON-IoT with both binary and multiclass settings, with ensemble models reaching 99.8 to 99.9 percent binary accuracy and sub-15-millisecond inference latency per 1000 flows.

Deep learning approaches have also been widely applied. Saadouni et al. proposed CNN-LSTM and CNN-GRU architectures for IIoT, with CNN-LSTM particularly effective for MITM detection [25]. Ullah et al. introduced a multi-head-attention-based gated recurrent unit (MAGRU) addressing class imbalance on Edge-IIoTset and MQTTset [26]. More recent work has explored vision transformer and bi-LSTM hybrids, including the SE-Enhanced ViT and BiLSTM intrusion detection model evaluated on CIC-IoT2023 and other benchmarks [27]. These deep architectures often achieve marginal gains over well-tuned ensembles at substantially higher training and inference cost, a trade-off that motivates the continued empirical strength of classical methods particularly when interpretability is required.

2.3 Multimodal and Cyber-Physical Fusion for IoT Security

A growing literature explores combining heterogeneous data sources for IoT intrusion detection. Ahmad and Alsmadi provide a comprehensive survey of data-fusion approaches for network intrusion detection, classifying methods by fusion level (data, feature, decision) and identifying open challenges in heterogeneous-source integration [5]. Ullah et al. proposed a multimodal big-data representation for IoT IDS that combined network bytes converted to images with text features, evaluated on Edge-IIoTset and CIC-IoT 2022/2023 with 98.2 percent accuracy [28]. Xu et al. presented a multimodal few-shot intrusion detection framework using feature fusion enhanced by gradient importance [29]. A recent 2026 study introduced a CNN-RNN-Transformer hybrid for fusing acceleration, gyroscope, microphone, and temperature sensor signals for cyber-physical threat detection, reporting an F1 score of 0.94 and 23-millisecond inference latency on edge hardware [16].

The general thrust of this literature is that fusion improves detection. However, very few studies disaggregate this claim by attack category, and fewer still compare against carefully matched single-modality baselines on the same benchmark. The present paper provides such a per-category empirical test on the most recent purpose-built synchronized IIoT benchmark.

2.4 Explainable AI for Intrusion Detection

Explainable AI methods, particularly SHAP (SHapley Additive exPlanations) [30], have become a standard tool for interpreting machine-learning-based IDSs [31, 32]. A recent study evaluated XGBoost, Random Forest, CatBoost, and a multi-layer perceptron on UNSW-NB15 with SHAP, LIME, and ELI5 explanations, demonstrating that XGBoost combined with SHAP delivers a strong combination of performance and transparency [32]. Khediri et al. integrated SHAP with large-language-model-generated natural-language descriptions to improve operator-facing interpretability of IDS decisions [33]. Khan et al. proposed an explainable deep learning framework for cyber threat discovery in IIoT networks, using SHAP-derived global feature importances to characterize attack signatures [12]. Recent systematic reviews of explainable AI for Industry-5.0 IDSs identify SHAP and LIME as the dominant techniques and note that per-class or per-category attribution analysis remains less common than aggregate global importance [34].

The use of SHAP in this paper differs from these prior applications in two ways. First, we apply SHAP not merely for ranking features but for quantifying the share of explanatory weight carried by each modality on a per-category basis. Second, we use SHAP attribution as confirmatory evidence alongside the classification-performance-based observability map, providing a two-perspective view of modality contribution.

2.5 MITM Detection in IoT

MITM attacks, frequently realized through ARP spoofing in IoT contexts, are a recurring focus in IoT IDS work because of their stealth and their direct impact on the integrity of device-broker communications [35, 36]. Alsaaidah et al. evaluated machine learning models for ARP-spoofing detection on IoTID20 with attention to the accuracy-versus-complexity trade-off [35]. Rostami and Goli-Bidgoli, working within an SDN-enabled smart-home MQTT setting, investigated detection of MITM attacks at the broker layer [37]. Rahman et al. introduced ARProof, a cross-protocol approach to detect and mitigate ARP-spoofing attacks in smart home networks, building on a dataset assembled from CIC-IoT, UQ-IoT-IDS, IoT Network Intrusion, and ARP PCAP sources [36]. Recent work by Yang et al. examined the role of time-window selection for ARP-spoofing detection and identified time-window length as an important hyperparameter when the spoofing event spans only a short interval [38]. These studies collectively establish MITM as a category where finer-grained protocol-aware features are often required, motivating examination of whether device-level telemetry provides additional information at the broker side. Our results provide direct evidence on this point.

3 Materials and Methods

Figure 1 presents the overall architecture of the proposed pipeline. The pipeline takes the DataSense pre-extracted feature CSVs as input, applies a sequence of preprocessing and modality-partitioning steps, evaluates three feature configurations against four classical machine-learning models on three classification tasks, and produces three categories of output: standard performance metrics, a per-category modality-observability map (the paper’s central novel contribution), and SHAP-based interpretability disaggregated by modality. A parallel sensitivity branch replicates the binary task at the 1-second window granularity to test robustness of the central finding.

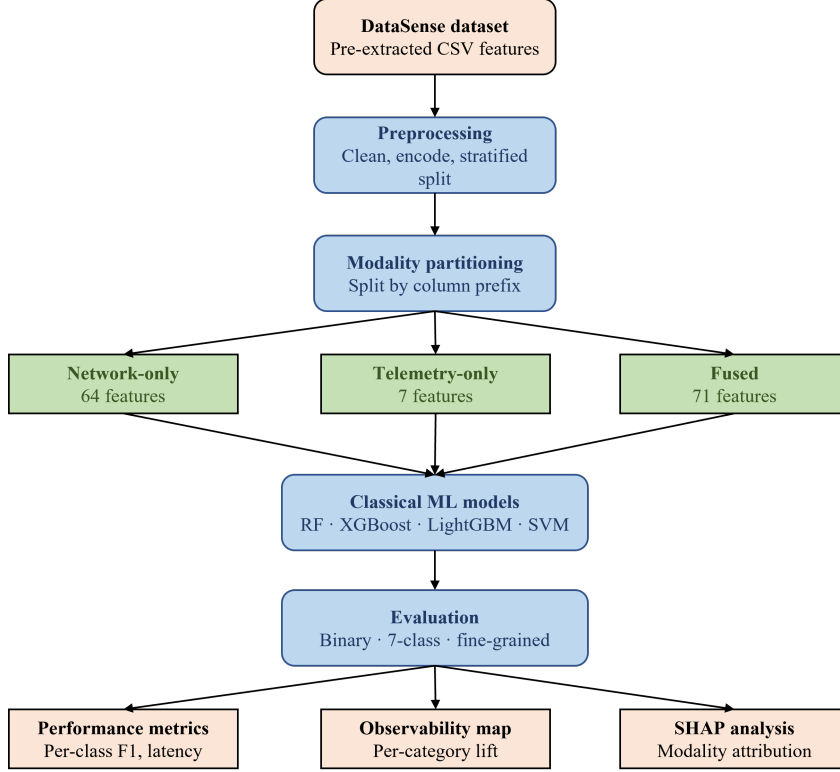


Fig. 1 Proposed pipeline architecture for the modality-aware IIoT intrusion detection system

3.1 Dataset and Working Corpus

We use the DataSense benchmark [1] in its pre-extracted, time-windowed CSV release. The dataset is structured into matched attack and benign feature files for time-window granularities ranging from 1 to 10 seconds. For the principal analysis we use the 10-second files, containing 16,350 attack windows and 13,680 benign windows for a combined working corpus of 30,030 instances. For the sensitivity analysis (Section 4.6)

we additionally use the 1-second files, which contain 90,391 attack windows and 136,800 benign windows.

Each row carries 94 columns: 84 features (76 prefixed **network_*** and 8 prefixed **log_***), 4 hierarchical attack-label columns (**label1** through **label4**), 1 device identifier (**device_name**), 1 MAC address, 1 full-label concatenation, and 3 timestamp columns. Attack labels at the category level (**label2**) span seven classes: reconnaissance, DDoS, DoS, MITM, malware, web, and brute force, with reconnaissance the most prevalent (6,005 windows) and brute force the least (364 windows) in our 10-second working corpus. Attack labels at the fine-grained level (**label4**) distinguish 84 attack scenarios because some categories are split by target port or device.

3.2 Preprocessing

Three preprocessing operations are applied before any modelling. First, benign label completion: benign rows in DataSense may have null entries in the attack-label columns, which we fill with the literal string “benign” so that downstream label encoding is well defined. Second, removal of string-valued columns: twelve network features and one telemetry feature in DataSense are encoded as variable-length strings (**network_ips_all**, **network_ips_dst**, **network_ips_src**, **network_macs_all**, **network_macs_dst**, **network_macs_src**, **network_ports_all**, **network_ports_dst**, **network_ports_src**, **network_protocols_all**, **network_protocols_dst**, **network_protocols_src**, and **log_data-types**). Each of these has a numeric **_count** companion already included in the feature set. We drop the string variants in favour of their **_count** companions for two reasons: the string variants encode high-cardinality identifiers whose presence in the training set could permit a classifier to memorize attacker fingerprints rather than learn attack-class patterns, and the **_count** companions already capture the cardinality information in a model-ready numeric form. After this step, the feature space comprises 64 **network_*** features and 7 **log_*** features for a total of 71 modelling features. Third, defensive median-imputation handling of missing values: in practice, no missing values were observed in the 10-second working corpus, so the imputation step had no effect; we retain it as a documented defensive measure.

3.3 Modality Partition and Feature Configurations

The feature partition is mechanical, driven by the column prefix in DataSense. The network modality comprises 64 features prefixed **network_***, capturing packet-size statistics, IP and port and MAC cardinalities, TCP flag counts, header and payload length statistics, time-delta statistics, TTL statistics, window-size statistics, and fragmentation indicators. The telemetry modality comprises 7 features prefixed **log_***, capturing log message count, inter-message interval, data-types count, and data-range statistics (average, maximum, minimum, standard deviation). From this partition we construct three feature configurations: C1 (Network-only) using the 64 **network_*** features, C2 (Telemetry-only) using the 7 **log_*** features, and C3 (Fused) using the 71-feature concatenation. This three-way partition is the operational basis of the per-category modality contribution analysis.

3.4 Experimental Design

A full factorial design crosses the three feature configurations with three classification granularities and four model families, producing 36 experimental conditions. The classification tasks are binary (attack versus benign), 7-class (`label12`), and fine-grained 84-class (`label14`). The model families are Random Forest [39], XGBoost [40], LightGBM [41], and Support Vector Machine with a radial basis function kernel [42]. The fixed-seed condition (`random_state = 42`) is used throughout for reproducibility. All experiments are conducted in Python 3.12 with scikit-learn, xgboost, lightgbm, and shap on a Google Colab CPU runtime.

3.5 Train, Validation, and Test Split

We use a stratified 70/15/15 train/validation/test split, stratified on the 7-class label so that every attack category appears in all three partitions. A stratified random split is chosen over a temporal-anchored split because attack scenarios in DataSense are captured as discrete sessions rather than as a continuous timeline; a temporal split would shuffle whole attack types between partitions rather than enforce temporal leakage protection. The resulting partition sizes are 21,021 training, 4,504 validation, and 4,505 test instances.

3.6 Models and Hyperparameters

We use sensible-default hyperparameters that are competitive in published IoT-IDS benchmark studies [10, 22, 23], deferring large-grid hyperparameter optimization to future work to keep the focus on modality contribution rather than per-model fine tuning. Random Forest is configured with 300 trees, no maximum depth limit, minimum two samples per leaf, balanced class weights, and parallel training. XGBoost is configured with 400 trees, maximum depth 8, learning rate 0.1, subsample 0.9, column subsample 0.9, the histogram tree method, the multi-class softmax objective for the seven-class and fine-grained tasks, and the binary logistic objective for the binary task. LightGBM is configured with 400 trees, 63 leaves, learning rate 0.1, minimum ten samples per leaf, feature fraction 0.9, and balanced class weights. The Support Vector Machine uses an RBF kernel with $C = 1.0$, $\gamma = \text{scale}$, and balanced class weights. Because RBF-SVM training scales quadratically with the number of training instances and becomes impractical above a few tens of thousands of rows [43], SVM training for multi-class tasks is performed on a stratified 8,000-row subsample of the training partition. This is a standard concession in IDS benchmarking work [24]. Standardization (zero-mean, unit-variance) is applied prior to SVM training only; tree-based ensembles are scale-invariant and do not require it.

3.7 Evaluation Metrics

For each of the 36 experimental conditions we compute accuracy, macro F1, weighted F1, macro precision, macro recall, training time, and inference latency per sample. Macro F1 is used as the primary comparison metric because it weights each class equally and is therefore robust to the class imbalance present in both the 7-class and

fine-grained tasks. For the seven-class task we additionally compute the per-category F1 score under each configuration, which is the basis of the modality observability map.

3.8 Modality Observability Map

The modality observability map is the operational core of the per-category contribution analysis. For each attack category c and each feature configuration k in {network, telemetry, fused}, we compute the per-class F1 score using the best-performing model on the seven-class task (XGBoost in our experiments). From these we derive two summary measures. First, modality lift, defined as the F1 under the fused configuration minus the maximum of the F1 under network-only and the F1 under telemetry-only, which measures the marginal contribution of fusion over the better single modality (positive values indicate complementary information; values at or near zero indicate redundancy). Second, dominant modality, defined as the argmax over network-only and telemetry-only of per-class F1, which identifies which modality carries more of the detection signal in isolation. Together, modality lift and dominant modality provide a categorical-level view of the cross-modal premise as it operates on DataSense.

3.9 SHAP Attribution Analysis

We complement the performance-based observability map with a SHAP-based attribution analysis [30]. SHAP TreeExplainer is applied to the best fused model on the seven-class task (XGBoost on C3) using a 1,000-instance random subsample of the test partition. From the resulting SHAP values we compute two quantities. First, per-feature mean absolute SHAP, averaged over samples and target classes, providing a global feature-importance ranking. Second, per-category modality share: for each attack category c , the proportion of total absolute SHAP attributed to network features versus telemetry features, computed over instances whose true label is c . The modality-share view is particularly useful because it operates on the fused model alone, eliminating any concern that the lift result is an artifact of model selection or training-data leakage between modalities.

3.10 Window-Size Sensitivity

To verify that the per-category observability map is not an artifact of the 10-second aggregation choice, we replicate the binary-task experiments on the 1-second window release of DataSense for Random Forest and LightGBM under all three feature configurations. The Random Forest and LightGBM models are selected for the sensitivity analysis as well-represented exemplars of the two ensemble families that dominate the main results, with SVM omitted owing to compute cost on the larger 1-second corpus and XGBoost omitted owing to its near-equivalence to LightGBM in the main results.

3.11 Reproducibility

The full implementation, including preprocessing, training, evaluation, and figure-generation code, is implemented in Python with fixed random seeds. Trained model

artifacts, the consolidated metrics CSV, the modality observability map, the SHAP attribution tables, and all figures in PNG and PDF formats are produced as outputs of the pipeline.

4 Results

4.1 Headline Performance

Table 1 reports macro F1 by task, model, and feature configuration. Three observations dominate. First, both network-only and fused configurations achieve very high performance on the binary and seven-class tasks (macro F1 in the range 0.98 to 0.99), with XGBoost the strongest individual learner on both. Random Forest is the strongest learner on the fine-grained 84-class task at 0.807 macro F1 on network-only features. Second, the telemetry-only configuration is substantially weaker across every task and every model. Best telemetry-only macro F1 is 0.61 on binary, 0.22 on the seven-class task, and 0.06 on the fine-grained task. Third, the gap between network-only and fused is essentially zero. On the binary task the best fused macro F1 (XGBoost) is 0.9884 versus 0.9890 for network-only. On the seven-class task it is 0.9827 versus 0.9819. On the fine-grained task it is 0.7928 versus 0.7937. The fused configuration neither materially improves nor materially harms detection relative to network-only across all three tasks.

Table 1 Macro F1 by task, model, and feature configuration on DataSense (10-second windows)

Task	Model	Network-only	Telemetry-only	Fused
Binary	LightGBM	0.9888	0.6040	0.9881
Binary	Random Forest	0.9859	0.6096	0.9859
Binary	SVM	0.9313	0.5708	0.9349
Binary	XGBoost	0.9890	0.6039	0.9884
7-class	LightGBM	0.9791	0.1816	0.9790
7-class	Random Forest	0.9727	0.1867	0.9749
7-class	SVM	0.8039	0.1151	0.8127
7-class	XGBoost	0.9819	0.2162	0.9827
Fine (84)	LightGBM	0.2661	0.0541	0.1958
Fine (84)	Random Forest	0.8070	0.0514	0.7975
Fine (84)	SVM	0.6048	0.0159	0.5978
Fine (84)	XGBoost	0.7937	0.0586	0.7928

LightGBM underperforms on the fine-grained 84-class task (0.20 to 0.27 macro F1 versus 0.79 to 0.81 for Random Forest and XGBoost). This is consistent with known sensitivity of LightGBM’s leaf-wise growth strategy to severe class imbalance with many classes, particularly under default-parameter settings, and is not a property of the modality configurations under test; Random Forest and XGBoost behave consistently across the three configurations on the same task.

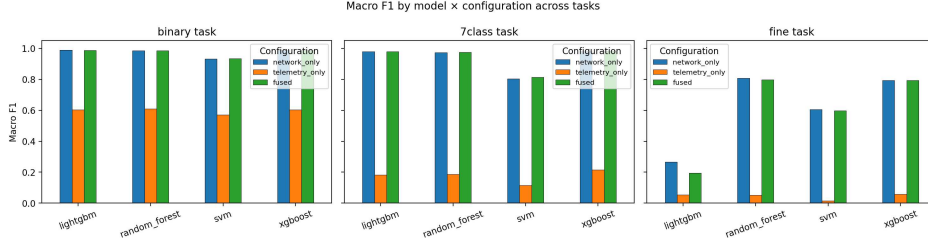


Fig. 2 Macro F1 by model and feature configuration across the binary, 7-class, and fine-grained tasks

4.2 Modality Observability Map

Table 2 presents the modality observability map: per-category F1 scores under each configuration, the best single-modality F1, the modality lift, and the dominant modality. The data confirm and refine the picture from Table 1 at the level of individual attack categories.

Table 2 Modality observability map on the seven-class task (XGBoost classifier)

Category	Net-only	Tele-only	Fused	Best single	Lift	Dominant
benign	0.9872	0.6610	0.9869	0.9872	-0.0003	network
bruteforce	0.9720	0.0000	0.9720	0.9720	0.0000	network
ddos	0.9907	0.1590	0.9907	0.9907	0.0000	network
dos	0.9836	0.1499	0.9836	0.9836	0.0000	network
malware	0.9789	0.0609	0.9789	0.9789	0.0000	network
mitm	0.9798	0.3060	0.9798	0.9798	0.0000	network
recon	0.9748	0.3926	0.9759	0.9748	0.0011	network
web	0.9881	0.0000	0.9940	0.9881	0.0059	network

The network channel is the dominant modality for every one of the seven attack categories plus benign. Modality lift is at or near zero in six of seven attack categories; the maximum positive lift is observed on the web category at 0.0059, and the recon category shows a minor positive lift of 0.0011. The bruteforce, ddos, dos, malware, and mitm categories all show exactly zero lift to four decimal places. The telemetry-only column itself shows a wider range of values (0.00 to 0.39), indicating that the seven telemetry features do carry some category-discriminative signal in isolation, but in every case the network channel already contains that signal and substantially more.

4.3 SHAP Attribution by Modality

Figure 4 (Top-15 SHAP features) and Figure 5 (per-category modality share) summarize the SHAP-based view. The top-15 features by mean absolute SHAP are all network-derived. The single most influential feature is `network_time-delta_min` (mean $|\text{SHAP}| = 0.834$), followed by `network_time-delta_avg` (0.630), `network_packets_dst_count` (0.370), `network_time-delta_std.deviation`

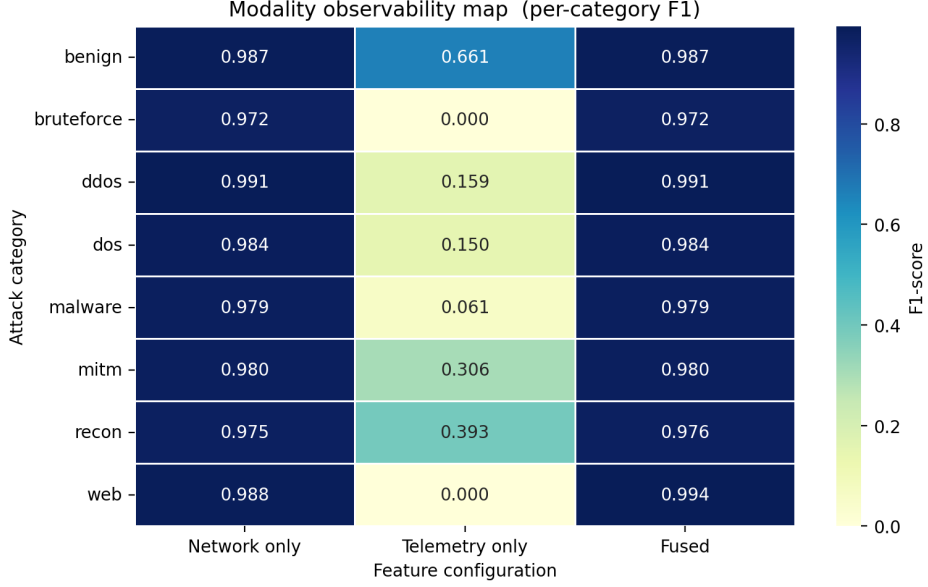


Fig. 3 Modality observability map: per-category F1 scores under each of the three feature configurations

(0.368), `network_packets_all_count` (0.343), `network_ttl_min` (0.326), and `network_packet-size_min` (0.324). Inter-packet timing statistics and packet-count statistics dominate the top of the ranking, consistent with the volumetric and rate-based signatures characteristic of DoS, DDoS, and reconnaissance attacks [44, 45].

The per-category modality share strengthens this picture. Table 3 reports the fraction of total absolute SHAP attribution carried by each modality on instances of each true class. The MITM category is a clear and isolated exception: telemetry features account for approximately 10 percent of explanatory weight when MITM instances are the target, an order of magnitude more than for any other category (the next highest is recon at 0.6 percent). The bruteforce and web categories show zero detectable telemetry contribution.

Table 3 Per-category SHAP modality share (XGBoost on fused features, seven-class task)

Category	Network share	Telemetry share
benign	0.9993	0.0007
bruteforce	1.0000	0.0000
ddos	0.9988	0.0012
dos	0.9957	0.0043
malware	0.9960	0.0040
mitm	0.8996	0.1004
recon	0.9942	0.0058
web	1.0000	0.0000

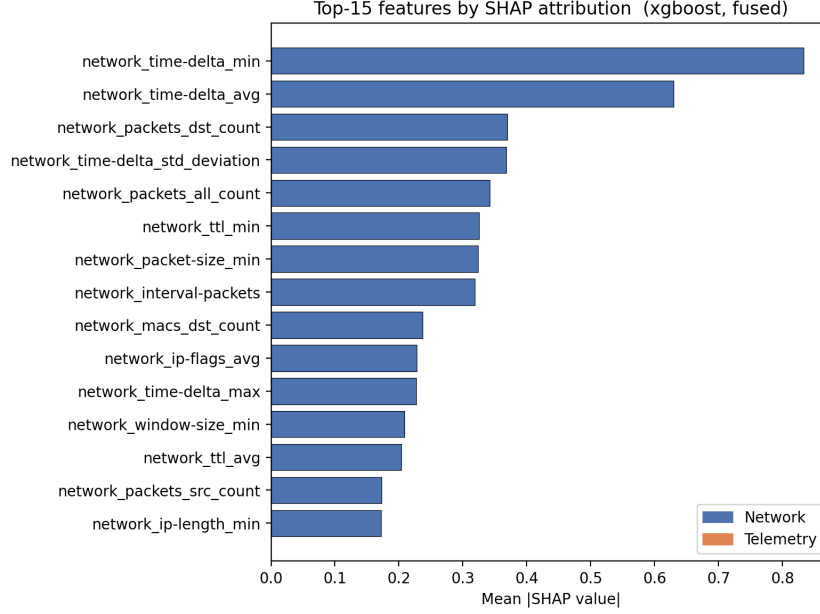


Fig. 4 Top-15 features by mean absolute SHAP attribution for the XGBoost fused-configuration classifier on the seven-class task. All features in the top ranking are network-derived

4.4 Confusion Matrix Analysis

The per-configuration confusion matrices on the seven-class task confirm the patterns above. Under network-only and fused configurations the diagonal dominates uniformly: per-class true-positive rates range from 0.96 (bruteforce, malware) to 1.00 (web), with off-diagonal mass concentrated in a small number of plausible confusions (recon-benign at 0.03, malware-benign at 0.02). Under the telemetry-only configuration the confusion matrix collapses almost entirely onto the benign column: 0.98 of true benign, 1.00 of true bruteforce, 0.86 of true DDoS, 0.89 of true DoS, 0.96 of true malware, 0.76 of true MITM, 0.73 of true recon, and 1.00 of true web are classified as benign. The telemetry channel in isolation provides essentially no useful discrimination between attack categories with the exception of partial recon (0.26 correctly classified) and partial MITM (0.19 correctly classified).

4.5 Inference Latency

Inference latency under network-only XGBoost on the binary task is 0.012 milliseconds per sample, which corresponds to an effective throughput on the order of 80,000 windows per second on a single CPU thread, an order of magnitude faster than published edge-deployment latency targets [16, 24]. Across the full set of 36 conditions, fused-configuration inference is uniformly within a factor of two of network-only inference, indicating that the fused configuration does not impose meaningful operational cost; the question of whether to deploy it is therefore decided on detection benefit rather than on cost, and the detection benefit is shown above to be effectively zero.

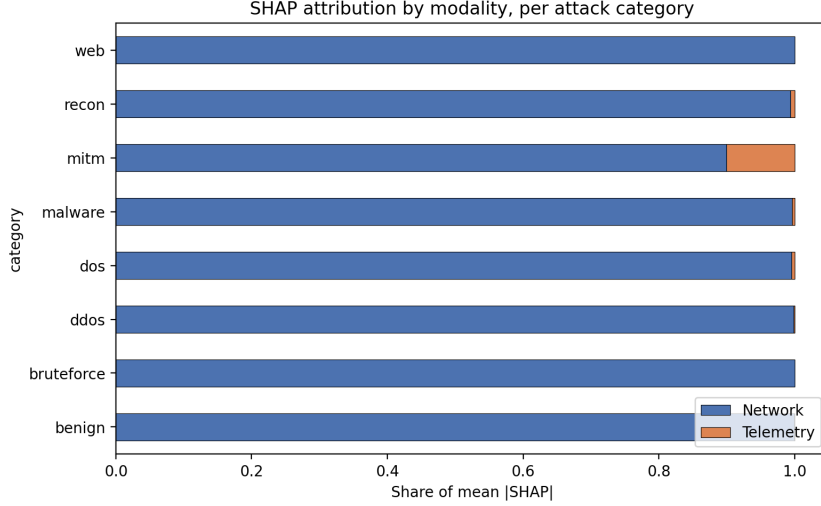


Fig. 5 SHAP attribution share by modality for each attack category. MITM is the single category with non-trivial telemetry contribution

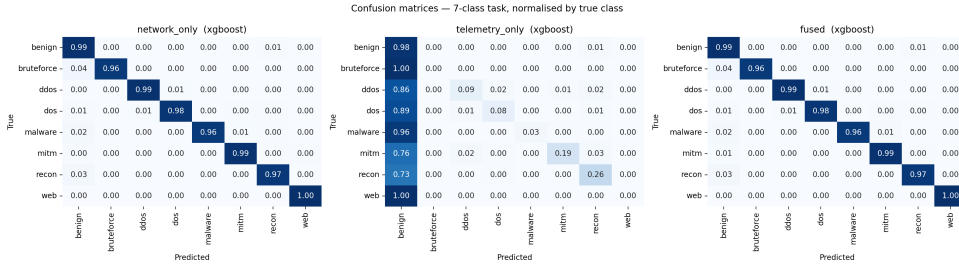


Fig. 6 Normalised confusion matrices for the XGBoost classifier under each of the three feature configurations on the seven-class task

4.6 Window-Size Sensitivity

Table 4 reports the sensitivity analysis. The pattern observed at 10-second windows is preserved at 1-second windows: network-only and fused configurations achieve near-identical macro F1 at both granularities (approximately 0.94 at 1-second and 0.99 at 10-second), while telemetry-only is substantially weaker (approximately 0.48 at 1-second and 0.60 at 10-second). All configurations degrade at finer granularity, but the telemetry channel degrades disproportionately (a roughly 12 percentage-point drop versus a roughly 5 percentage-point drop for network and fused). This is mechanistically consistent: MQTT log messages are sparser than network packets, and shorter windows often contain too few log entries for the 7 aggregated telemetry features to remain informative.

The central finding that network features alone are nearly sufficient for IIoT intrusion detection on DataSense holds independently of the window granularity at which features are aggregated.

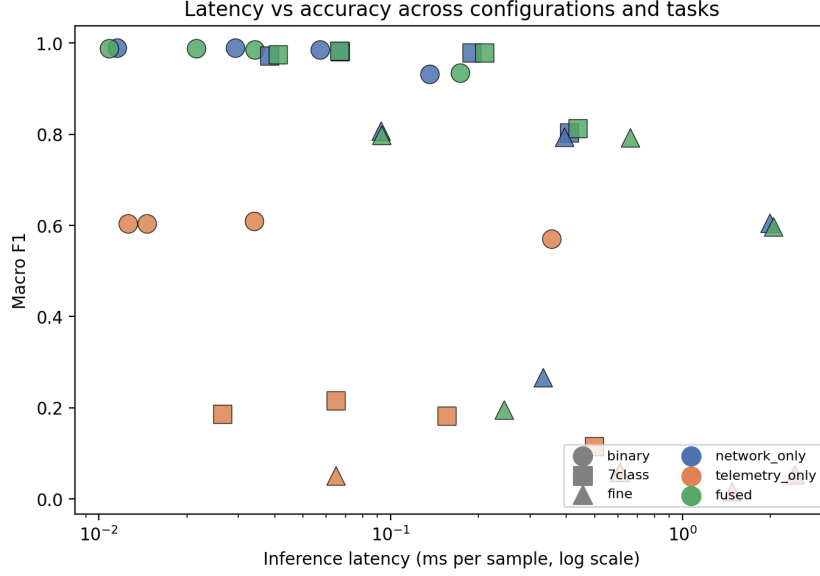


Fig. 7 Inference latency versus macro F1 across the full 36-condition experiment matrix. Marker shape encodes the classification task; marker colour encodes the feature configuration

Table 4 Window-size sensitivity on the binary task (macro F1)

Configuration	Model	10-second	1-second	Delta
Network-only	LightGBM	0.9888	0.9419	-0.047
Network-only	Random Forest	0.9859	0.9410	-0.045
Fused	LightGBM	0.9881	0.9425	-0.046
Fused	Random Forest	0.9859	0.9419	-0.044
Telemetry-only	LightGBM	0.6040	0.4831	-0.121
Telemetry-only	Random Forest	0.6096	0.4833	-0.126

5 Discussion

5.1 Why Network Dominates

The dominance of network features across all seven attack categories on DataSense admits a clear mechanistic explanation. Volumetric attacks (DoS, DDoS, malware-derived flooding) leave direct, high-amplitude signatures in packet-rate, packet-size, and inter-packet-timing statistics that the network features capture with very high fidelity [44, 45]. Reconnaissance attacks (port scans, host discovery scans, OS scans, vulnerability scans) similarly produce characteristic patterns in source-address cardinality, port cardinality, TCP-flag distributions, and inter-packet timing that the network features encode directly [46]. Brute-force attacks, on this dataset realized as repeated authentication attempts against the broker, generate distinctive packet-rate patterns at attack-target ports. Web-based attacks generate atypical payload-length

distributions and HTTP-method-related TCP flag patterns. In each case the attack mechanism operates at the network layer or through it, and the network features capture the relevant signal directly.

The 7 telemetry features in DataSense, by contrast, are aggregated statistics over MQTT log messages: message count, inter-message interval, data-type count, and data-range statistics. These are second-order summaries of device behaviour that capture the effect of attacks on application-layer message patterns rather than the attack signal itself. For attack categories whose effects on application messages are either invisible (because the device is offline during a flood) or redundant with what the network channel already shows (because abnormal packet rates imply abnormal message rates), the telemetry features carry no marginal information.

5.2 The MITM Exception Explained

The MITM exception is consistent with the physical semantics of ARP-spoofing and IP-spoofing attacks as implemented in DataSense [1]. In an ARP-spoofing attack the attacker poisons the ARP cache of the broker (or of a device) so that subsequent traffic intended for one device is redirected to the attacker, who may then optionally forward it on. At the network layer this produces relatively subtle anomalies (changes in MAC-IP binding cardinality, small timing perturbations) that are still detectable but less dramatic than the volumetric signatures of DoS or DDoS. At the broker layer, however, the rerouting can alter which devices respond to messages and how quickly they do so, producing measurable deviations in the rate, regularity, and content of MQTT log messages. The 10 percent SHAP share for the telemetry channel on MITM instances captures this effect.

The MITM category is therefore the one category on DataSense where the cross-modal premise empirically holds, albeit weakly. Notably, the telemetry contribution shows up in SHAP attribution but not in modality lift on the seven-class macro-F1 view, because the network channel by itself already achieves a high MITM F1 of 0.98. This is not a contradiction: SHAP attribution captures how the fused model uses information, while modality lift captures the gain relative to the best single-modality baseline. The two metrics together suggest that telemetry information is genuinely informative for MITM detection but is largely redundant in the sense that the network channel already extracts what is needed to achieve high macro F1. This redundancy interpretation aligns with prior work on ARP-spoofing detection where network-level features alone reach high accuracy [35, 36].

5.3 Implications for IIoT Monitoring Practice

The practical implication for IIoT operators is clear and actionable: for the attack categories represented in DataSense, a monitoring stack that ingests and processes only network features can achieve macro F1 within 0.6 percentage points of a more elaborate fused stack on every category. This matters because telemetry collection from heterogeneous industrial devices typically requires custom agents, broker-level instrumentation, and downstream log-shipping infrastructure that adds operational complexity and cost. Our results suggest that this additional complexity is, at least at

the level of feature aggregation provided by DataSense, not repaid by detection benefit for six of seven attack categories. Operators concerned specifically with MITM-class attacks may justifiably invest in broker-level telemetry, but operators concerned with general-purpose attack detection should focus their effort on network-layer monitoring quality. This is also consistent with the operational position of the recent DeepSense framework [20], which treats network features as the primary detection surface.

5.4 Implications for Benchmark Design

A second implication is for benchmark design. The DataSense feature release is structured to enable cross-modal research, but the level of aggregation at which the telemetry channel is exposed (seven highly-summarized features) may limit the cross-modal premise from playing out empirically. Richer telemetry features, for example per-sensor numeric readings, per-topic message-rate time series, or message-content distributional features, may carry information that aggregated message counts and data ranges do not. Future iterations of DataSense and related benchmarks could explore exposing telemetry at finer granularities to test whether the dominance pattern reported here persists. A related point concerns labelling. The fine-grained 84-class task is essentially defined by `label14` distinctions that split categories by target port or device; these distinctions are network-observable by construction, which further reinforces the network channel as the dominant signal source. A benchmark that exposed attack categories distinguished by device-internal effects rather than by network-target metadata would provide a sharper test of telemetry value.

5.5 Comparison with Prior Work

Our headline accuracy numbers are competitive with state-of-the-art classical-ML results reported on adjacent IoT benchmarks. The 0.989 macro F1 we obtain on the DataSense binary task is comparable to the 99.8 to 99.9 percent binary accuracies reported on TON-IoT by Shaikhanova et al. [24] using a similar four-model ensemble panel, and to the 98.89 percent accuracy reported by Akif et al. on IoT-23 with XGBoost [21]. Our 0.982 macro F1 on the seven-class task is in line with multi-class results reported on CIC-IoT2023 with similar classical ensemble methods [22, 23]. Where our work departs from these comparators is in the per-category disaggregation by modality, which to our knowledge has not been performed on DataSense.

Our results also nuance the narrative in recent cross-modal IDS literature. The multimodal CNN-RNN-Transformer system of [16] reports $F1 = 0.94$ on a heterogeneous sensor-and-network setting; the multimodal big-data representation of Ullah et al. [28] reports 98.2 percent accuracy on combined Edge-IIoTset and CIC-IoT 2022/2023 corpora. These works demonstrate that fusion can work at high overall accuracy levels, but they do not test per-category whether the fusion is in fact carrying complementary signal or whether the dominant modality is doing essentially all of the work. Our per-category and SHAP-disaggregated results suggest that the latter possibility deserves serious attention in future cross-modal benchmarking.

6 Limitations and Future Work

Several limitations should be noted. First, our analysis is restricted to one benchmark dataset. While DataSense is the most recent and most thoroughly designed synchronized IIoT IDS dataset to date, the generalizability of the network-dominance finding to other multimodal benchmarks (existing or future) is an empirical question. Replicating the per-category modality-aware analysis on other datasets where synchronized non-network signals are available is an immediate avenue for further work.

Second, we restrict ourselves to classical machine learning models (Random Forest, XGBoost, LightGBM, SVM). Deep learning architectures with explicit cross-modal fusion (cross-attention, joint encoders, hybrid CNN-LSTM-Transformer models) might extract complementary information from the telemetry channel that classical learners cannot. Empirical evaluation of cross-modal deep models against rigorously matched single-modality baselines on DataSense would be a valuable extension.

Third, the telemetry features in DataSense are aggregated at a relatively coarse level (counts, averages, ranges over a time window). It is plausible that finer-grained telemetry features (raw time series of sensor readings, per-topic message statistics, content-level distributional features) carry information that the eight aggregated features do not. Re-engineering richer telemetry features from the underlying log streams and re-running the modality contribution analysis is a natural next step.

Fourth, we use sensible-default rather than fully optimized hyperparameters. A more thorough hyperparameter search could improve absolute accuracy figures, particularly on the fine-grained task, but is unlikely to alter the relative ordering between configurations. Fifth, our SHAP analysis is computed on a 1,000-instance subsample for tractability. Computing SHAP on the full test set could surface secondary attribution patterns that the subsample misses, although prior work suggests that 1,000-instance subsamples are typically sufficient to stabilize global attribution patterns [30]. Sixth, the LightGBM under-performance on the fine-grained 84-class task is unaddressed in the present analysis and could likely be improved with explicit class-imbalance handling. We retain it as observed because the focus of the paper is on modality contribution rather than on absolute model ranking.

Future work in addition to the above includes extending the analysis to device-class-conditioned modality contribution (does the network/telemetry asymmetry differ between industrial sensors and IP cameras?), examining open-set detection in which entire attack types are held out from training, and integrating the resulting observability map into operator-facing decision-support tools.

7 Conclusion

This paper presented a modality-aware analysis of IIoT intrusion detection on the DataSense benchmark, testing rather than assuming the empirical validity of the cross-modal premise. We partitioned the dataset’s 84 pre-extracted features into 64 network and 7 telemetry modelling features, constructed three feature configurations (network-only, telemetry-only, fused), and conducted 36 experimental conditions across three classification granularities and four classical machine learning families. We

then characterized the per-category contribution of each modality through a modality observability map and through SHAP attribution disaggregated by modality, and verified the robustness of our findings to temporal aggregation window size.

The principal finding is that network features alone are nearly sufficient for IIoT intrusion detection on DataSense. Across all seven attack categories, the network modality is dominant, fusion lift is at or near zero (maximum 0.6 percentage points on the web category), and SHAP attribution assigns the network channel over 99 percent of explanatory weight in six of seven categories. The single exception is the MITM category, where the telemetry channel contributes approximately 10 percent of SHAP attribution, consistent with the broker-layer effects of ARP-spoofing-based MITM attacks. The dominance pattern holds at both 1-second and 10-second window granularities.

These results have direct implications for IIoT monitoring practice (network-layer monitoring is operationally sufficient for the attack categories represented on DataSense, with broker-layer telemetry adding marginal value primarily for MITM detection) and for IIoT benchmark design (richer telemetry features may be required for the cross-modal premise to play out empirically). They also caution against assuming, in the absence of per-category empirical testing, that multimodal fusion in IIoT IDS is universally beneficial.

Declarations

Funding. The authors declare that no funding was received for this research.

Competing interests. The author has no relevant financial or non-financial interests to disclose.

Ethics approval and consent to participate. Not applicable.

Consent for publication. Not applicable.

Data availability. This study uses the publicly available DataSense benchmark dataset [1] released by the Canadian Institute for Cybersecurity. No new data were generated in this study.

Materials availability. Not applicable.

Code availability. The implementation code is available from the author on reasonable request.

References

- [1] Firouzi, A., Dadkhah, S., Maret, S.A., Ghorbani, A.A.: DataSense: A real-time sensor-based benchmark dataset for attack analysis in IIoT with multi-objective feature selection. *Electronics* **14**(20), 4095 (2025) <https://doi.org/10.3390/electronics14204095>

- [2] Sharma, S.B., Bairwa, A.K.: Leveraging AI for intrusion detection in IoT ecosystems: A comprehensive study. *IEEE Access* **13**, 66290–66317 (2025) <https://doi.org/10.1109/ACCESS.2025.3550392>
- [3] Saheed, Y.K., Omole, A.I., Sabit, M.O.: GA-mADAM-IIoT: A new lightweight threats detection in the industrial IoT via genetic algorithm with attention mechanism and LSTM on multivariate time series sensor data. *Sensors International* **6**, 100297 (2025) <https://doi.org/10.1016/j.sintl.2024.100297>
- [4] Khan, I.A., Keshk, M., Pi, D., Khan, N., Hussain, Y., Soliman, H.: Enhancing IIoT networks protection: A robust security model for attack detection in internet industrial control systems. *Ad Hoc Networks* **134**, 102930 (2022) <https://doi.org/10.1016/j.adhoc.2022.102930>
- [5] Ahmad, R., Alsmadi, I.: Data fusion and network intrusion detection systems. *Cluster Computing* **27**(6), 7493–7519 (2024) <https://doi.org/10.1007/s10586-024-04365-y>
- [6] Almotairi, A., Atawneh, S., Khashan, O.A., Khafajah, N.M.: Enhancing intrusion detection in IoT networks using machine learning-based feature selection and ensemble models. *Systems Science & Control Engineering* **12**(1), 2321381 (2024) <https://doi.org/10.1080/21642583.2024.2321381>
- [7] Azam, Z., Islam, M.M., Huda, M.N.: Comparative analysis of intrusion detection systems and machine learning-based model analysis through decision tree. *IEEE Access* **11**, 80348–80391 (2023) <https://doi.org/10.1109/ACCESS.2023.3296444>
- [8] Alashjaee, A.M., Alqahtani, F.: Enhanced intrusion detection system IoT network security model by feed forward neural network and machine learning. *Scientific Reports* **15**(1), 36085 (2025) <https://doi.org/10.1038/s41598-025-20047-0>
- [9] Jony, A.I., Arnob, A.K.B.: A long short-term memory based approach for detecting cyber attacks in IoT using CIC-IoT2023 dataset. *Journal of Edge Computing* **3**(1), 28–42 (2024) <https://doi.org/10.55056/jec.648>
- [10] Neto, E.C.P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., Ghorbani, A.A.: CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. *Sensors* **23**(13), 5941 (2023) <https://doi.org/10.3390/s23135941>
- [11] Herzalla, D., Lunardi, W.T., Andreoni, M.: TII-SSRC-23 dataset: Typological exploration of diverse traffic patterns for intrusion detection. *IEEE Access* **11**, 118577–118594 (2023) <https://doi.org/10.1109/ACCESS.2023.3319213>
- [12] Khan, M.A., *et al.*: A deep learning-based intrusion detection system for MQTT enabled IoT. *Sensors* **21**(21), 7016 (2021) <https://doi.org/10.3390/s21217016>
- [13] Kasongo, S.M., Sun, Y.: Performance analysis of intrusion detection systems using

- a feature selection method on the UNSW-NB15 dataset. *Journal of Big Data* **7**(1), 105 (2020) <https://doi.org/10.1186/s40537-020-00379-6>
- [14] Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H.: 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) <https://doi.org/10.1109/ACCESS.2022.3165809>
 - [15] Uppal, M., Gulzar, Y., Gupta, D., Uppal, J., Kumar, M., Saini, S.: Enhancing accuracy through ensemble based machine learning for intrusion detection and privacy preservation over the network of smart cities. *Discover Internet of Things* **5**(1), 11 (2025) <https://doi.org/10.1007/s43926-025-00101-z>
 - [16] Latif, M., *et al.*: AI-based intelligent sensing detection of cybersecurity threats using multimodal sensor data in smart devices. *Scientific Reports* **16**(1), 11091 (2026) <https://doi.org/10.1038/s41598-026-40614-3>
 - [17] Zhou, Z.: Multimodal fusion anomaly detection model for agricultural wireless sensors. *Engineering Reports* **6**(12), 13021 (2024) <https://doi.org/10.1002/eng2.13021>
 - [18] Khan, I.A., *et al.*: Fed-Inforce-Fusion: A federated reinforcement-based fusion model for security and privacy protection of IoMT networks against cyber-attacks. [Online]. Accessed: May 30, 2026 (2024). https://dro.deakin.edu.au/articles/journal_contribution/Fed-Inforce-Fusion_A_federated_reinforcement-based_fusion_model_for_security_and_privacy_protection_of_IoMT_networks_against_cyber-attacks/24257167/1
 - [19] Belarbi, O., Spyridopoulos, T., Anthi, E., Rana, O., Carnelli, P., Khan, A.: Gotham Dataset 2025: A Reproducible Large-Scale IoT Network Dataset for Intrusion Detection and Security Research. *arXiv:2502.03134* (2025). <https://doi.org/10.48550/arXiv.2502.03134>
 - [20] Firouzi, A., Ghorbani, A.A.: DeepSense: An adaptive scalable ensemble framework for industrial IoT anomaly detection. *Sensors* **26**(9), 2662 (2026) <https://doi.org/10.3390/s26092662>
 - [21] Akif, M.A., Butun, I., Williams, A., Mahgoub, I.: Hybrid Machine Learning Models for Intrusion Detection in IoT: Leveraging a Real-World IoT Dataset. *arXiv:2502.12382* (2025). <https://doi.org/10.48550/arXiv.2502.12382>
 - [22] Ibrahim, M., Al-Wadi, A.: Enhancing IoMT network security using ensemble learning-based intrusion detection systems. *Journal of Engineering Research* **13**(4), 3166–3175 (2025) <https://doi.org/10.1016/j.jer.2024.12.003>
 - [23] Kouassi, B.M., Ballo, A.B., Ayikpa, K.J., Mamadou, D., Coulibaly, M.Z.J.: Top-K

- feature selection for IoT intrusion detection: Contributions of XGBoost, LightGBM, and random forest. *Future Internet* **17**(11), 529 (2025) <https://doi.org/10.3390/fi17110529>
- [24] Shaikhanova, A., Kuznetsov, O., Tokkuliyeva, A., Ayapbergenov, K., Olzhas, S., Danir, T.: Security audit of IoT device networks: A reproducible machine learning framework for threat detection and performance benchmarking. *Sensors* **25**(24), 7519 (2025) <https://doi.org/10.3390/s25247519>
 - [25] Saadouni, R., Khacha, A., Harbi, Y., Gherbi, C., Harous, S., Aliouat, Z.: Secure IIoT networks with hybrid CNN-GRU model using Edge-IIoTset. In: 2023 15th International Conference on Innovations in Information Technology (IIT), pp. 150–155 (2023). <https://doi.org/10.1109/IIT59782.2023.10366486>
 - [26] Ullah, S., Boulila, W., Koubâa, A., Ahmad, J.: MAGRU-IDS: A multi-head attention-based gated recurrent unit for intrusion detection in IIoT networks. *IEEE Access* **11**, 114590–114601 (2023) <https://doi.org/10.1109/ACCESS.2023.3324657>
 - [27] Gueriani, A., Kheddar, H., Mazari, A.C., Sagioglu, S., Ceran, O.: SE-enhanced ViT and BiLSTM-based intrusion detection for secure IIoT and IoMT environments. In: 2025 18th International Conference on Information Security and Cryptology (ISCTürkiye), pp. 1–6 (2025). <https://doi.org/10.1109/ISCTrkiye68593.2025.11224819>
 - [28] Ullah, F., Turab, A., Ullah, S., Cacciagrano, D., Zhao, Y.: Enhanced network intrusion detection system for internet of things security using multimodal big data representation with transfer learning and game theory. *Sensors* **24**(13), 4152 (2024) <https://doi.org/10.3390/s24134152>
 - [29] Xu, C., Zhan, Y., Wang, Z., Yang, J.: Multimodal fusion based few-shot network intrusion detection system. *Scientific Reports* **15**(1), 21986 (2025) <https://doi.org/10.1038/s41598-025-05217-4>
 - [30] Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS’17), pp. 4768–4777. Curran Associates Inc., Red Hook, NY, USA (2017)
 - [31] Gaspar, D., Silva, P., Silva, C.: Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron. *IEEE Access* **12**, 30164–30175 (2024) <https://doi.org/10.1109/ACCESS.2024.3368377>
 - [32] Mohale, V.Z., Obagbuwa, I.C.: Evaluating machine learning-based intrusion detection systems with explainable AI: enhancing transparency and interpretability. *Frontiers in Computer Science* **7** (2025) <https://doi.org/10.3389/fcomp.2025.1520741>

- [33] Khediri, A., Slimi, H., Yahiaoui, A., Derdour, M., Bendjenna, H., Ghenai, C.E.: Enhancing machine learning model interpretability in intrusion detection systems through SHAP explanations and LLM-generated descriptions. In: 2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS), pp. 1–6 (2024). <https://doi.org/10.1109/PAIS62114.2024.10541168>
- [34] Khan, N., Ahmad, K., Al Tamimi, A., Alani, M.M., Bermak, A., Khalil, I.: Explainable AI-based intrusion detection systems for industry 5.0 and adversarial XAI: A systematic review. *Information* **16**(12), 1036 (2025) <https://doi.org/10.3390/info16121036>
- [35] Alsaaidah, A., Almomani, O., Abu-Shareha, A.A., Abualhaj, M.M., Achuthan, A.: ARP spoofing attack detection model in IoT network using machine learning: Complexity vs. accuracy. *Journal of Applied Data Sciences* **5**(4), 1850–1860 (2024) <https://doi.org/10.47738/jads.v5i4.374>
- [36] Rahman, M.M., Bouhafs, F., Hoseini, S.A., Hartog, F.: ARProof: A cross-protocol approach to detect and mitigate ARP-spoofing attacks in smart home networks. *Journal of Network and Computer Applications* **246**, 104396 (2026) <https://doi.org/10.1016/j.jnca.2025.104396>
- [37] Karmous, N., Ben Dhiab, Y., Ould-Elhassen Aouileyine, M., Youssef, N., Bouallegue, R., Yazidi, A.: Deep learning approaches for protecting IoT devices in smart homes from MitM attacks. *Frontiers in Computer Science* **6** (2024) <https://doi.org/10.3389/fcomp.2024.1477501>
- [38] Yang, Y.-M., Chang, K.-C., Luo, J.-N.: A lightweight training approach for MITM detection in IoT networks: Time-window selection and generalization. *Applied Sciences* **15**(22), 12147 (2025) <https://doi.org/10.3390/app152212147>
- [39] Breiman, L.: Random forests. *Machine Learning* **45**(1), 5–32 (2001) <https://doi.org/10.1023/A:1010933404324>
- [40] Chen, T., Guestrin, C.: XGBoost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM, San Francisco, CA, USA (2016). <https://doi.org/10.1145/2939672.2939785>
- [41] Ke, G., *et al.*: LightGBM: A highly efficient gradient boosting decision tree. In: Advances in Neural Information Processing Systems. Curran Associates, Inc., ??? (2017). Accessed: May 30, 2026. <https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html>
- [42] Cortes, C., Vapnik, V.: Support-vector networks. *Machine Learning* **20**(3), 273–297 (1995) <https://doi.org/10.1007/BF00994018>

- [43] Chang, C.-C., Lin, C.-J.: LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* **2**(3), 1–27 (2011) <https://doi.org/10.1145/1961189.1961199>
- [44] Gelgi, M., Guan, Y., Arunachala, S., Samba Siva Rao, M., Dragoni, N.: Systematic literature review of IoT botnet DDOS attacks and evaluation of detection techniques. *Sensors* **24**(11), 3571 (2024) <https://doi.org/10.3390/s24113571>
- [45] Doshi, R., Apthorpe, N., Feamster, N.: Machine learning DDoS detection for consumer internet of things devices. In: 2018 IEEE Security and Privacy Workshops (SPW), pp. 29–35 (2018). <https://doi.org/10.1109/SPW.2018.00013>
- [46] Hindy, H., Bayne, E., Bures, M., Atkinson, R., Tachtatzis, C., Bellekens, X.: Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset). In: Ghita, B., Shiaele, S. (eds.) *Selected Papers from the 12th International Networking Conference*, pp. 73–84. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-64758-2_6