

# A Hybrid Machine Learning Framework for Robust Detection of Malicious DNS-over-HTTPS Tunneling Traffic

Enas Selem<sup>1</sup>

`enas-salem@fci.suezuni.edu.eg`

Suez University

Rania Salama<sup>1</sup>

Suez University

---

## Research Article

**Keywords:** DNS over HTTPS, DoH tunneling detection, hybrid machine learning, XGBoost, anomaly detection, Leave-One-Attack-Out

**Posted Date:** August 10th, 2026

**DOI:** <https://doi.org/10.21203/rs.3.rs-10519149/v1>

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

[Read Full License](#)

**Additional Declarations:** No competing interests reported.

---

# A Hybrid Machine Learning Framework for Robust Detection of Malicious DNS-over-HTTPS Tunneling Traffic

Enas Selem<sup>1\*</sup> and Rania Salama<sup>1</sup>

<sup>1</sup>Information Technology Department, Faculty of Computers and Information, Suez University, P.O. Box 43221, Suez, Egypt

\*Corresponding author: Enas Selem — enas-salem@fci.suezuni.edu.eg

Email addresses: Enas Selem — enas-salem@fci.suezuni.edu.eg; Rania Salama — rania@fci.suezuni.edu.eg

## Acknowledgements

[Optional — insert any acknowledgements here, or write "Not applicable" if none.]

## Abstract

DNS over HTTPS (DoH) enhances privacy but also enables covert tunneling and data exfiltration. This paper proposes a density-aware hybrid framework, positioned as a batch-processing analyser, combining XGBoost with unsupervised anomaly scoring from Isolation Forest and Local Outlier Factor, interpreted via SHAP; an ablation study shows deep generative components add no measurable benefit. Evaluated on the CIRA-CIC-DoHBrw-2020 benchmark, the framework improves Recall over the baseline in the Standard and Full Data scenarios (F1 = 96.11% vs. 95.96%,  $p = 0.024$ ), with a tied Cross-Scenario result. In a fifteen-configuration Leave-One-Attack-Out evaluation, the framework reduced false negatives by 70.8-74.1% (Wilcoxon  $p = 0.031$ ). A companion Leave-One-Tool-Out evaluation across three structurally different tools confirms the same maximal significance, though performance remains limited for one tool (dns2tcp) by a genuine feature-domain gap. These results show anomaly scoring provides a statistically validated safety net for unseen variants at the batch level.

**Keywords:** *DNS over HTTPS; DoH tunneling detection; hybrid machine learning; XGBoost; anomaly detection; Leave-One-Attack-Out.*

## 1. Introduction

For most of the Internet's history, DNS queries were transmitted in plain text; thus, they were easily visible, as well as being susceptible to eavesdropping, forging, and blocking. The concerns surrounding DNS query privacy led to the development of encrypted protocols. One of the most popular alternatives is DNS over HTTPS, defined in RFC 8484 in 2018 [1]. Due to the rapid adoption of DoH by Chrome, Firefox, Windows, and Android browsers, this technology transformed within a few years from an experimental to a widespread practice, implemented on hundreds of millions of devices. This widespread adoption has led to security concerns, as DoH traffic is identical to HTTPS traffic, making it a convenient way to communicate across the network without being detected [2,3]. Attackers realised this early on. DNS tunneling — the practice of hiding data inside DNS queries and responses — had existed for years, but DoH gave it a significant upgrade: the tunnel was encrypted, ran on a port that firewalls almost never block, and blended invisibly into surrounding traffic. APT34, a state-sponsored threat actor, was the first known group to weaponise DoH for data exfiltration in 2020 [3], and several open-source tunneling tools including iodine and dns2tcp were subsequently updated to support DoH as a transport layer.

Detecting this kind of attack is genuinely difficult. Deep packet inspection is useless when the payload is encrypted. The field has therefore turned to machine learning, using flow-level statistical features such as packet timing, inter-arrival intervals, and session lengths to distinguish malicious from benign traffic [4]. The release of the CIRA-CIC-DoHBrw-2020 dataset [5] gave researchers a common benchmark, and the years since have seen a steady stream of papers reporting ever-higher accuracy figures.

Reading that literature carefully, however, reveals a recurring pattern of limitations. Most systems evaluate performance only when training and test data come from the same distribution and the same attack types [6]. Several rely on a single feature (the DoH protocol indicator) that an attacker who knows they are being monitored can trivially suppress [7]. Very few have tested whether their models can detect a tunneling technique they have never seen before [8], and fewer still provide the decision-level explanations that a security analyst would need to act on an alert with confidence [9].

This paper presents a density-aware hybrid detection framework designed to overcome those specific weaknesses, positioned as a batch-processing analyzer for security operations rather than an inline real-time filter. Rather than training a single supervised classifier and hoping it generalises, we combine two independent sources of evidence. XGBoost [10] provides a strong supervised baseline using raw network statistics. Isolation Forest [11] and Local Outlier Factor [12] each score how anomalous a given flow looks relative to normal traffic, without any knowledge of attack labels, from a global and a local perspective respectively. These signals are concatenated into a single feature vector before the final classification step. Deliberately, no deep generative component such as an autoencoder is included, since an ablation study shows it adds complexity without a measurable gain (Section 4.4). SHAP [13] analysis runs on both the hybrid model and the baseline, making it possible to see exactly which features are driving each decision. We test the framework on three primary scenarios of increasing difficulty, followed by a dedicated robustness evaluation. The most demanding test removes the DoH feature entirely and repeatedly holds out each tunneling variant from training across multiple random seeds. Even there, the hybrid model consistently and substantially cuts missed attacks compared to the baseline, with no exceptions in any configuration tested, evidence that the anomaly-scoring components genuinely compensate once the easiest signal is unavailable. This resilience is not free: Section 4.11 quantifies the computational trade-off it costs relative to the baseline classifier.

In summary, this paper makes the following contributions:

- (1) a rigorous, multi-seed Leave-One-Attack-Out (LOAO) evaluation protocol — three held-out tunneling variants  $\times$  five random seeds, fifteen independent configurations in total — extended by a companion Leave-One-Tool-Out (LOTO) evaluation across three structurally different tunneling tools, together substantially more demanding than the single-split evaluations standard in this sub-area (Table 1; Sections 4.8 and 4.12);
- (2) formal statistical validation of every empirical claim, via McNemar's exact test for the three primary scenarios (Section 4.9) and Wilcoxon signed-rank/exact sign tests for the LOAO and LOTO evaluations, where the hybrid framework's advantage is significant at the maximum level attainable with five paired seeds ( $p = 0.031$ ) in every configuration tested (Sections 4.8 and 4.12);
- (3) a mechanistic explanation, grounded in decision-boundary analysis pooled across seeds, for why the hybrid framework helps: it corrects the baseline specifically on flows sitting near XGBoost's own decision boundary rather than improving detection uniformly (Section 4.10);
- (4) transparent reporting of where the framework does not help and what it costs: no consistent advantage under benign-distribution shift (Section 4.7), a genuine feature-domain gap limiting absolute performance against structurally distant tools such as dns2tcp (Section 4.12), and a measurable computational overhead relative to the baseline classifier that we quantify under both streaming and batched inference (Section 4.11); and
- (5) a hybrid detection framework, positioned as a batch-processing analyzer rather than an inline real-time filter, that augments XGBoost with unsupervised, density-aware spatial anomaly scoring (Isolation Forest and Local Outlier Factor), avoiding the added complexity of deep generative components after an ablation study showed they provide no measurable benefit (Section 4.4). The evaluation protocol and statistical validation above (contributions 1–3) are, in our view, as central to this paper's contribution as the

architecture itself, since the protocol is designed to be reusable for evaluating other DoH detection systems regardless of their underlying model.

The paper is structured as follows. Section 2 reviews the relevant literature. Section 3 describes the proposed framework. Section 4 presents the experimental results, including a dedicated statistical significance analysis. Section 5 discusses the implications, and Section 6 introduces the conclusions.

## **2. Related Work**

### **2.1 DNS Tunneling and the DoH Threat**

DNS has served as a secret channel since the early 2000s. Conventional DNS tunneling techniques include iodine and dns2tcp, which encode information into DNS subdomain values then transmit the data to a malicious server, resulting in signatures like long domain names, frequent queries, and high entropy payload values. DoH changes that picture by carrying DNS inside HTTPS, which hides payload characteristics and places the traffic on port 443, a port that is rarely blocked. The result is that defenders can no longer rely on content inspection alone and must instead analyze flow-level statistics [2,14,15].

The benchmark that most subsequent research relies on was created by MontazeriShatoori et al. [5], who captured network flows from Google Chrome as benign traffic and iodine tunneling sessions across multiple DoH resolvers, extracted 28 statistical features per flow, and proposed a two-stage classifier — first separating DoH from non-DoH, then separating benign DoH from malicious DoH. Vekshin et al. [16] extended this work with DoH Insight, a lightweight real-time detection system, while Mitsuhashi et al. [17] moved one step further by attempting to identify the specific tunneling tool in use.

### **2.2 Supervised and Deep Learning Methods**

Among the supervised approaches, several studies have shown that strong results are possible with engineered features and carefully designed classifiers. Zebin et al. [9] presented a stacked Random Forest classifier with SHAP explanations to conclude that time and size attributes have the maximum discriminatory power. Nguyen and Park [18] instead used a two-layer Transformer architecture, showing that self-attention can distinguish DoH tunneling traffic without the sequential processing constraints of recurrent models. Recent research has employed optimization-based feature selection, stacking, and multi-task learning approaches to enhance results; however, these studies only consider standard conditions and do not address adversarial resilience.

### **2.3 Unsupervised and Hybrid Approaches**

Isolation Forest [11] and Local Outlier Factor [12] have long histories as general-purpose anomaly detectors. Both can be fitted without any attack labels, which is critical for detecting novel threats. Mirsky et al.'s Kitsune system [19] demonstrated that autoencoders [20] trained on benign traffic produce reliable anomaly scores — reconstruction errors — for a wide variety of novel attack types. Ji et al.'s [4] systematic review of AI-based anomaly detection over encrypted traffic found that hybrid approaches combining supervised and unsupervised learning consistently outperform either individually, a finding echoed by Hussein and Répás [21] for traditional IDS, by Sajid et al. [22] in cloud computing environments, by Erskine and Suresh [23] who used an explainable hybrid CNN-XGBoost framework with leakage-aware feature selection for multi-class IoT intrusion detection, and by Almuhanha and Dardouri [24] for zero-day attack detection.

### **2.4 Adversarial Robustness**

Zhan et al. [25] showed that flow-level features remain useful for exfiltration detection even when payload content is encrypted. Elaoumari [7] tested detectors against adversarial DoH traffic crafted to resemble benign flows, finding significant degradation in systems relying on a small number of high-signal features.

Amirov et al. [8] and Sharma and Swarnkar [26] reach a similar conclusion: robust detection requires multiple independent evidence sources rather than a single shortcut, and systems that depend on a small number of high-signal features are inherently fragile against adaptive attackers.

## 2.5 Positioning of This Work

Table 1 summarizes the comparison. The main gap is that prior studies typically combine only a subset of the following elements: unsupervised anomaly scoring, SHAP interpretability, and evaluation under genuinely unseen attack conditions. This work brings all of these together in a single framework, tested under three complementary scenarios plus a dedicated multi-seed unseen-variant robustness evaluation (see the full list of contributions in Section 1).

**Table 1 Positioning of the proposed framework relative to representative prior work**

| Study                           | Year        | Approach                       | Dataset                | Anomaly  | SHAP     | Unseen Attack |
|---------------------------------|-------------|--------------------------------|------------------------|----------|----------|---------------|
| MontazeriShatoori et al. [5]    | 2020        | Two-layer time-series ML       | CIC-DoHBrw-2020        | ✗        | ✗        | ✗             |
| Zebin et al. [9]                | 2022        | Stacked RF + SHAP              | CIC-DoHBrw-2020        | ✗        | ✓        | ✗             |
| Nguyen & Park [18]              | 2022        | Two-layer Transformer          | CIC-DoHBrw-2020        | ✗        | ✗        | ✗             |
| Talabani et al. [27]            | 2025        | ACO + XGBoost / RF / CNN       | CIC-DoHBrw-2020        | ✗        | ✗        | ✗             |
| Hussein et al. [28]             | 2025        | Stacked LSTM-GRU + XGBoost     | CIC + HKD              | ✗        | ✗        | ✗             |
| MTL-DoHTA [29]                  | 2025        | Multi-task 2D-CNN              | CIC-DoHBrw-2020        | ✗        | ✗        | ✗             |
| EnhancedRLGWO [30]              | 2025        | RL + GWO feature selection     | CIC-DoHBrw-2020        | ✗        | ✗        | ✗             |
| <b>Proposed Hybrid (IF+LOF)</b> | <b>2025</b> | <b>XGBoost + IForest + LOF</b> | <b>CIC-DoHBrw-2020</b> | <b>✓</b> | <b>✓</b> | <b>✓</b>      |

## 3. Proposed Framework

### 3.1 Framework Design

The framework is based on a simple idea: rather than depending on one supervised model to solve every part of the problem, we provide it with complementary signals. Raw statistical features capture the standard view used in previous work. Two unsupervised anomaly detectors, Isolation Forest and Local Outlier Factor, capture how far a flow deviates from benign traffic without needing attack labels, from a global and a local perspective respectively. These signals are then combined before classification so that the model can exploit whichever evidence is most informative in a given scenario. Deliberately, no deep generative

component such as an autoencoder is included: an ablation study (Section 4.4) shows that the added architectural and computational cost is not justified by any measurable gain in detection performance, so the lighter IF+LOF combination is retained as the final design. The final classification is performed by XGBoost [10], and SHAP [13] analysis is applied afterwards to explain what drove each decision.

This design avoids dependence on any single feature. If an attack is of an unusual type, both the Isolation Forest and the Local Anomaly Factor, trained only on normal traffic, are able to classify it as an unusual attack because it does not match the inferred normal density. This gives the framework flexibility in more challenging evaluation environments, and it was directly verified in Section 4.8 by evaluating the One-Attack Exclusion (LOAO) across multiple types of unusual attacks and random seeds.

### 3.2 Framework Overview

Table 2 lays out the full five-stage pipeline.

**Table 2 The proposed Hybrid (IF+LOF) detection pipeline**

| Stage                | Component                       | Method                             | Output                     |
|----------------------|---------------------------------|------------------------------------|----------------------------|
| 1 — Data Preparation | Cleaning and encoding           | Median imputation + StandardScaler | Normalised feature matrix  |
| 2a — Global Anomaly  | Outlier scoring across all data | Isolation Forest (80 trees)        | iso_score                  |
| 2b — Local Anomaly   | Density-based outlier scoring   | Local Outlier Factor (k=20)        | lof_score                  |
| 3 — Feature Fusion   | Merging all signal sources      | Horizontal concatenation (hstack)  | X_hybrid                   |
| 4 — Classification   | Binary traffic labelling        | XGBoost (150 trees, depth=5)       | Malicious / Benign         |
| 5 — Threshold Tuning | Calibrating decision boundary   | F1 maximisation on validation set  | Optimal threshold $\tau^*$ |

### 3.3 Data Preparation

Raw CSV files from the CIRA-CIC-DoHBrw-2020 dataset [5] are loaded and labelled based on their directory path: Chrome browser traffic is benign (class 0), iodine tunnel traffic is malicious (class 1). Files that are unreadable or empty are excluded, leaving 5,545 usable files and 402,073 labelled flow records.

Features with more than 60% missing values or zero variance are dropped, leaving 32 usable numerical features. Categorical features with low cardinality are ordinally encoded using mappings derived from the training split only. Missing and infinite values are replaced with the training-set median, and all features are standardized to zero mean and unit variance using scikit-learn's StandardScaler [31], fitted only on training data. This separation is maintained across all experimental scenarios to reduce leakage and preserve a realistic evaluation.

Two splitting strategies are used. In the Standard and Full Data scenarios, splits are done randomly with stratification (60% for training, 20% for validation, and 20% for testing). The Cross-Scenario split strategy requires splitting on a file-level basis: each CSV file goes into only one split, so that any particular flow from the same capture cannot go into both training and test sets. This makes the evaluation of Cross-Scenario a stronger test of generalization across captures rather than across records from the same capture.

### 3.4 Anomaly Detection

#### 3.4.1 Isolation Forest

The Isolation Forest [11] is fitted on the benign portion of the training set only. It builds 80 isolation trees, each of which randomly selects a feature and a split point and recursively partitions the data. Anomalous points — being few and different — tend to be isolated in fewer steps. The decision function output is negated so that higher values indicate greater anomaly, and this is appended to the feature matrix as `iso_score`. Because the model has seen only benign traffic during training, anything that deviates — including attack types never seen before — will receive an elevated score.

### 3.4.2 Local Outlier Factor

LOF [12] provides a complementary perspective. Rather than comparing each point to the global distribution, it computes a density ratio between each point and its  $k=20$  nearest neighbours. A point whose neighbourhood is much sparser than its neighbours' neighbourhoods receives a high LOF score — locally anomalous even if not globally unusual. Like Isolation Forest, LOF is fitted only on benign data and produces `lof_score`. The combination of `iso_score` and `lof_score` gives the classifier two independent views of abnormality — one global, one local — which together cover a wider range of anomaly types than either alone.

## 3.5 Feature Fusion

The final feature vector for the XGBoost classifier is formed from two sets: the 32 normalised features and two spatial anomaly scores (`iso_score`, `lof_score`). There is no deep generative component, which keeps the fusion architecture simpler than the autoencoder-augmented variant evaluated in our ablation study (Section 4.4), while still allowing the classifier to weight the relevant signals differently for different cases; Section 4.11 shows this still introduces measurable inference overhead relative to the baseline classifier alone.

## 3.6 Classification and Threshold Optimisation

The final classification is executed using the XGBoost [10] algorithm (150 trees, `max_depth` = 5, learning rate = 0.08). The class imbalance exists in the dataset (on average nine benign flows for every one malicious flow) is addressed using benign-class downsampling and `scale_pos_weight` parameter tuning. The decision threshold  $\tau^*$  is selected by maximizing F1 on the validation set:

$$\tau^* = \operatorname{argmax}_{\tau} 2 \cdot P(\tau) \cdot R(\tau) / (P(\tau) + R(\tau))$$

This threshold is computed entirely from the validation set and applied unchanged to the test set. The resulting threshold values reflect the difficulty of each scenario: 0.7874 for the Standard scenario, 0.5410 for the Full Data scenario, and 0.4790 for the Cross-Scenario setting, with lower thresholds compensating for a harder or less separable confidence distribution. For the Leave-One-Attack-Out (LOAO) robustness evaluation reported in Section 4.8, a fixed threshold of 0.5 is used for both the hybrid and baseline models instead of the F1-optimal threshold, since the objective there is a direct, threshold-agnostic comparison of false-negative behaviour across unseen attack types and random seeds rather than per-scenario calibration.

## 3.7 Interpretability with SHAP

SHAP TreeExplainer [13] computes exact Shapley values for both the Hybrid (IF+LOF) and Baseline XGBoost models on the test set. A positive Shapley value for a feature means it pushed the prediction towards malicious; a negative value pushed it towards benign. Global importance summaries and per-sample beeswarm plots together allow analysts to verify that the spatial anomaly features are genuinely driving predictions in the right direction — not just nominally ranked — which is precisely the kind of confirmation that builds operational trust [9].

# 4. Experimental Evaluation

## 4.1 Dataset and Setup

All experiments use the CIRA-CIC-DoHBrw-2020 dataset [5]. Its malicious traffic spans three distinct DNS tunneling tools: iodine (in three query-type sub-variants, null, srv, and txt), dns2tcp, and dnscat2. The three primary scenarios (Sections 4.5-4.7) and the LOAO evaluation (Section 4.8) use iodine traffic only; the LOTO evaluation (Section 4.12) additionally draws on dns2tcp and dnscat2 to test generalization across structurally different tools. After filtering the available CSV files, 5,545 were selected, generating a total of 402,073 flow records with a highly imbalanced benign majority. Experiments were conducted on Google Colab, using XGBoost and scikit-learn [31] for classification, preprocessing, and anomaly detection.

## 4.2 Evaluation Metrics

Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Correlation Coefficient are presented for all three scenarios (Table 3). The number of false negatives is also presented, since missing a malicious flow is more costly than raising a false alarm. Precision-Recall curves and their corresponding average precision (PR-AUC) are shown for the Full Data and Cross-Scenario settings, where the greater class imbalance makes this view particularly informative. Section 4.9 additionally reports McNemar's exact test for each scenario, testing whether the hybrid and baseline models' error patterns differ by more than chance.

## 4.3 Summary of Results

Table 3 summarizes the results across the three primary evaluation scenarios. The hybrid approach outperforms the baseline in both the Standard and Full Data scenarios, whereas the two approaches are virtually tied in the Cross-Scenario setting (equal Recall, slightly lower Precision for the hybrid approach). All three scenarios achieve ROC-AUC above 0.998, and the difference between the two approaches never exceeds 0.0002 in any scenario. The LOAO evaluation of zero-day robustness is discussed in Section 4.8, and the statistical significance of these primary comparisons is examined in Section 4.9.

**Table 3 Detection performance across the three primary experimental scenarios (Standard, Full Data, Cross-Scenario). Results for zero-day robustness appear separately in Table 4 (Section 4.8)**

| Scenario       | Model           | Acc.   | Prec.  | Recall | F1     | ROC-AUC | MCC    |
|----------------|-----------------|--------|--------|--------|--------|---------|--------|
| Standard       | Hybrid (IF+LOF) | 99.17% | 90.93% | 97.54% | 94.11% | 99.90%  | 93.74% |
| Standard       | Baseline        | 99.19% | 92.34% | 95.97% | 94.11% | 99.90%  | 93.70% |
| Full Data      | Hybrid (IF+LOF) | 99.20% | 94.29% | 98.00% | 96.11% | 99.92%  | 95.68% |
| Full Data      | Baseline        | 99.17% | 94.05% | 97.95% | 95.96% | 99.92%  | 95.52% |
| Cross-Scenario | Hybrid (IF+LOF) | 99.03% | 94.22% | 96.59% | 95.39% | 99.89%  | 94.86% |
| Cross-Scenario | Baseline        | 99.11% | 94.87% | 96.59% | 95.72% | 99.89%  | 95.23% |

## 4.4 Ablation Study

An earlier version of this framework additionally incorporated a denoising autoencoder, trained on benign traffic and contributing sixteen latent features plus a reconstruction-error score. In that earlier ablation, on the Standard scenario, XGBoost augmented with Isolation Forest and Local Outlier Factor alone reached 97.25% Recall (28 false negatives), while adding the autoencoder either on its own (96.76% Recall, 33 false negatives) or combined with the anomaly scores (96.95% Recall, 31 false negatives) did not exceed this result on any metric. Since the autoencoder added architectural complexity, training time, and sixteen

additional latent dimensions without a measurable gain over the lighter IF+LOF combination, it was removed from the final design; all results reported elsewhere in this paper use the autoencoder-free framework. With the autoencoder removed, the ablation reduces to a direct comparison between a raw XGBoost classifier using only the 32 original flow features and the proposed hybrid classifier augmented with the two spatial anomaly scores. This comparison closely corresponds to the Standard scenario reported in Table 3: the hybrid variant improves Recall from 95.97% to 97.54% and reduces false negatives from 41 to 25, at a modest cost in Precision (92.34% vs. 90.93%). Table 3 therefore already isolates the contribution of the anomaly-scoring components under standard conditions, and Section 4.8 isolates their contribution under zero-day conditions specifically.

**Implications of Robustness.** The true worth of the proposed framework lies not in its superiority over the baseline on every metric in easier scenarios, but in its better performance when the strongest feature is excluded from the data or the attacker changes tactics. This is precisely what robustness means in practical use cases. Section 4.8 tests this claim directly and rigorously through a Leave-One-Attack-Out evaluation across multiple unseen attack types and random seeds.

### 4.5 Scenario 1: Standard Evaluation

Both models surpass 99% accuracy under normal conditions and achieve an ROC-AUC above 0.998. The Hybrid approach shows a distinctly higher Recall (97.54% versus 95.97%, 25 versus 41 false negatives), while the baseline retains a slight advantage in Precision; however, this difference in Recall is not statistically significant (McNemar's exact test,  $p = 0.883$ ; Section 4.9). This is expected because the DoH feature is fully available and the attack pattern is familiar, allowing the plain XGBoost model to exploit a strong single signal for Precision; the anomaly scores nonetheless catch several malicious flows the baseline misses. The hybrid model's advantage becomes statistically significant in the Full Data scenario that follows, though not in the Cross-Scenario setting after it (Section 4.9).

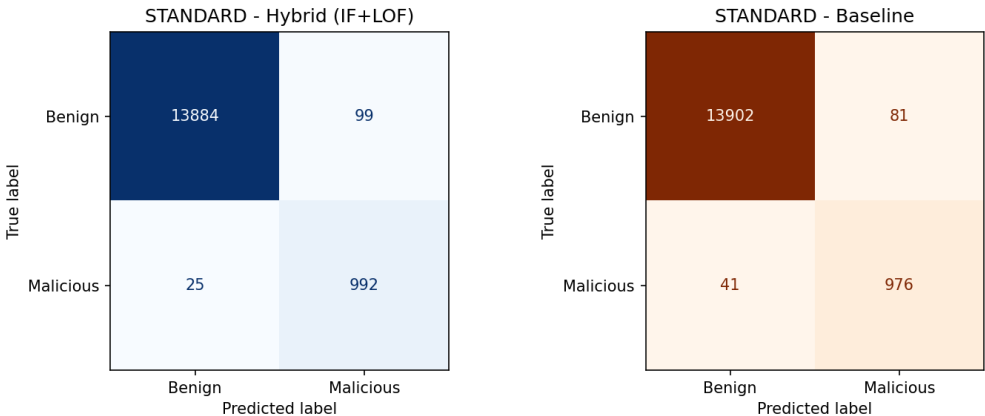


**Fig. 1 Confusion** matrices for the Standard scenario — Hybrid (IF+LOF) (left) and Baseline (right)

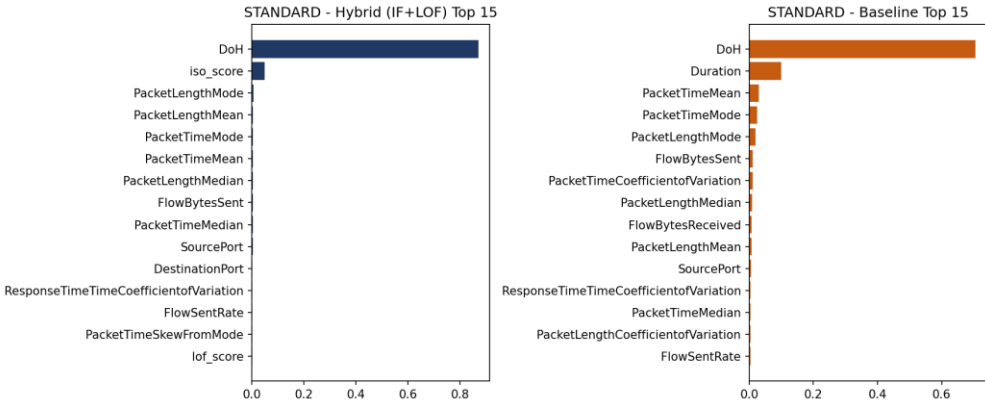


**Fig. 2** Gain-based feature importance in the Standard scenario. The DoH flag dominates both models; iso\_score is the clear second-ranked feature in the hybrid model, while the baseline instead relies secondarily on flow Duration

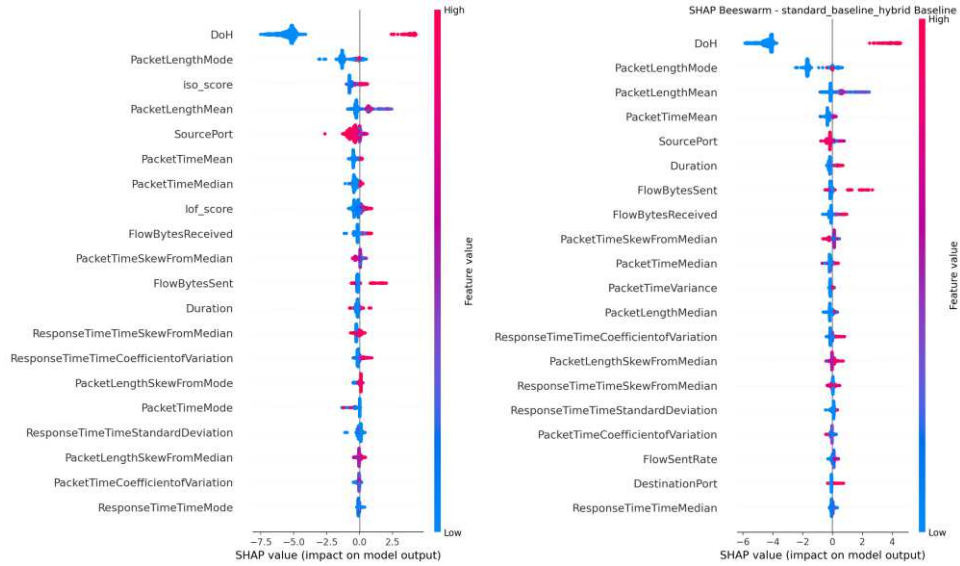


**Fig. 3** SHAP beeswarm plots for the Standard scenario — Hybrid (IF+LOF) (left) and Baseline (right)

#### 4.6 Scenario 2: Full Data

In the second scenario we use the complete dataset without file-level or session-level restrictions. Training is consistent, and the hybrid model performs the best among all scenarios: Recall increases to 98.00% (163 false negatives) versus 97.95% (167 false negatives), while F1 also improves (96.11% compared to 95.96%). This is the only one of the three primary scenarios where the improvement is statistically significant (McNemar's exact test,  $p = 0.024$ ; Section 4.9). Unlike the smaller Standard-scenario sample, the additional volume of data here lets the anomaly-scoring components pick up a small but consistent number of malicious flows that the baseline's decision boundary alone still misses, without any Precision penalty this time, the hybrid framework outperforms the baseline on every reported metric in this scenario.

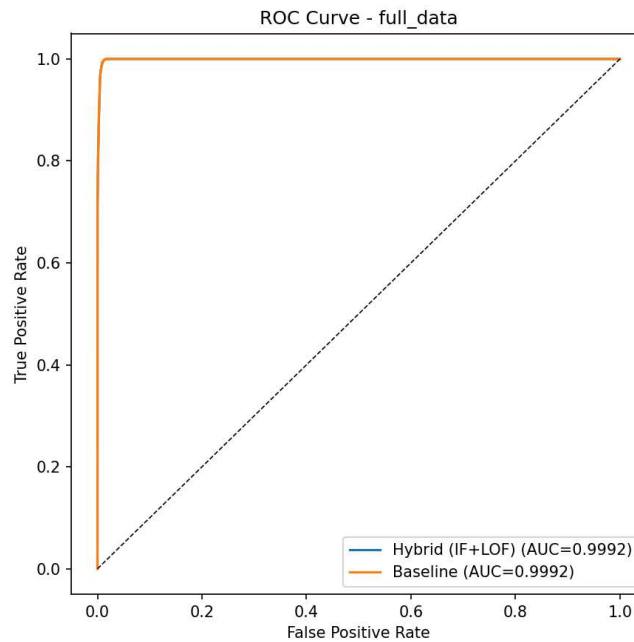


**Fig. 4** ROC curves for the Full Data scenario. Both models achieve ROC-AUC of 0.9992

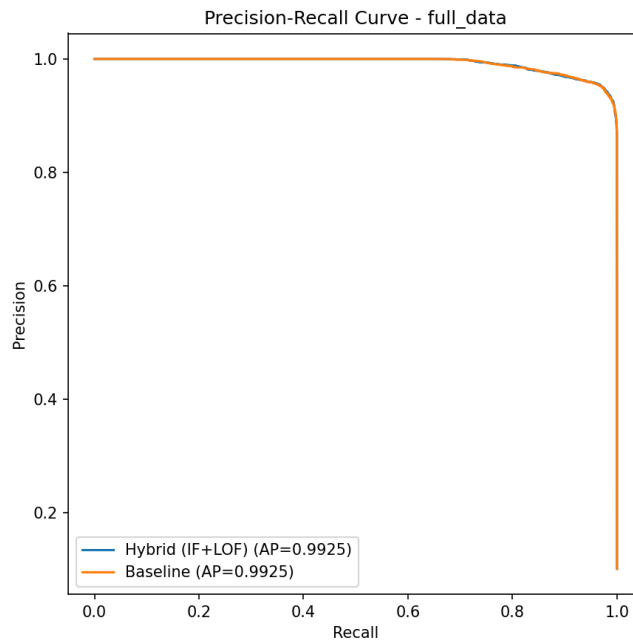


**Fig. 5** Precision-Recall curves for the Full Data scenario (Hybrid AP=0.9925, Baseline AP=0.9925)

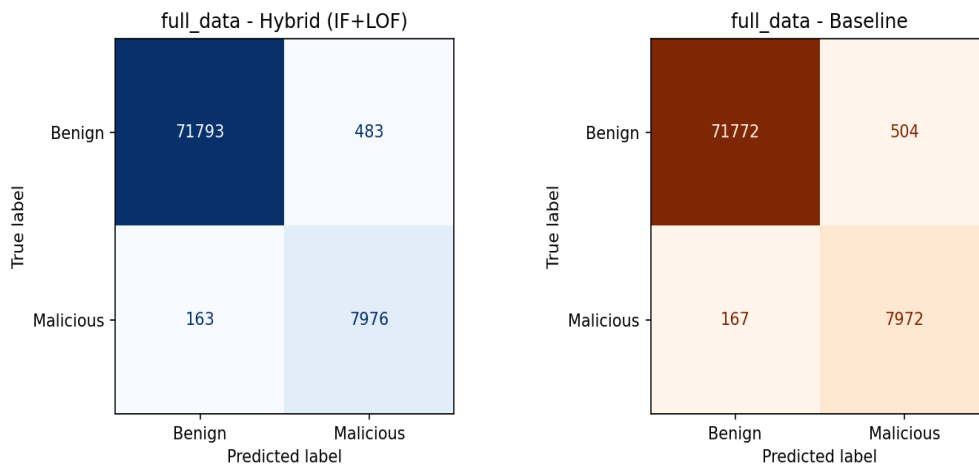


**Fig. 6** Confusion matrices for the Full Data scenario — Hybrid (IF+LOF) (left) and Baseline (right)

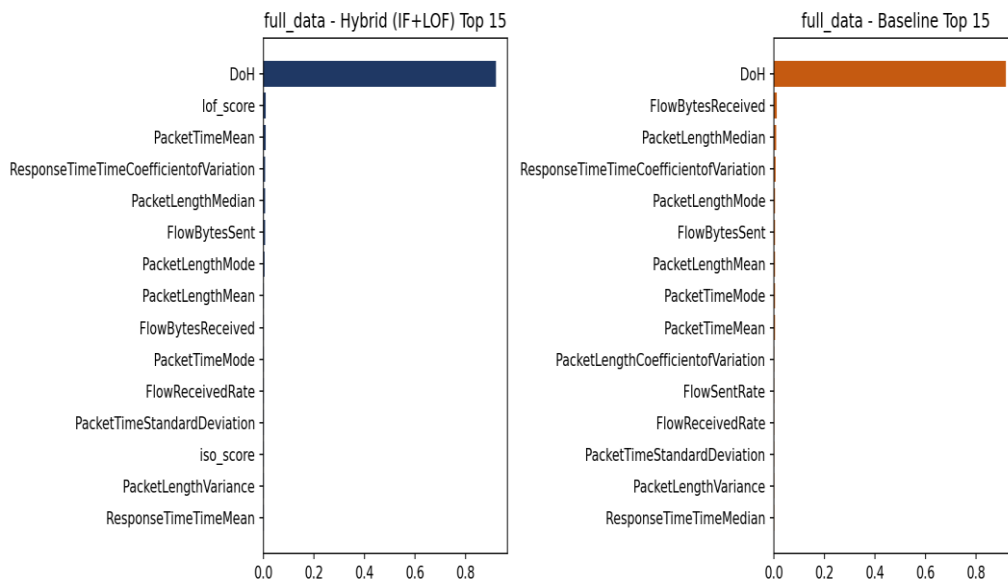


**Fig. 7** Gain-based feature importance for the Full Data scenario. DoH dominates both models; lof\_score is the clear second-ranked feature in the hybrid model, while the baseline instead relies secondarily on FlowBytesReceived

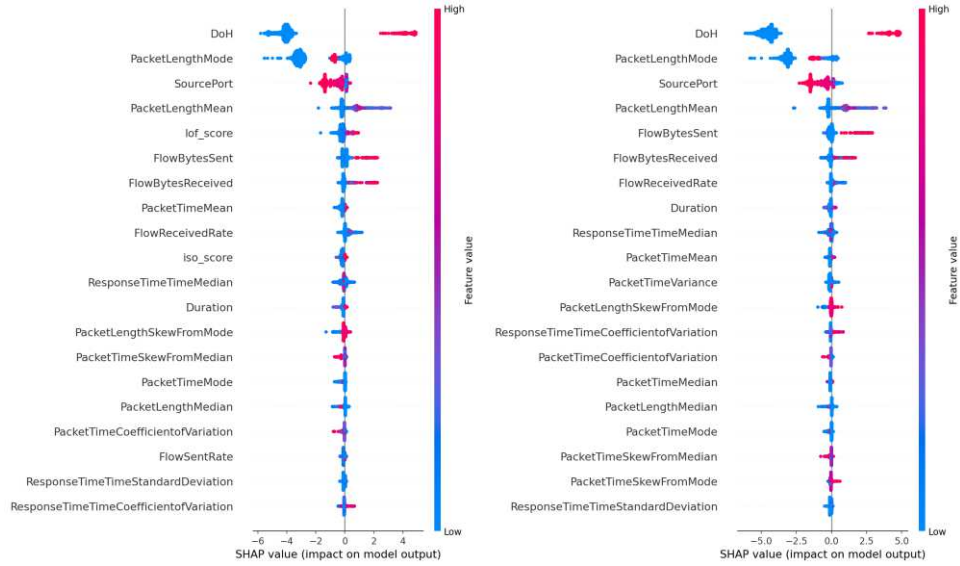


**Fig. 8** SHAP beeswarm plots for the Full Data scenario — Hybrid (IF+LOF) (left) and Baseline (right). `lof_score` is the fifth-ranked feature by mean  $|\text{SHAP}|$  in the hybrid model, ahead of `PacketTimeMean`

#### 4.7 Scenario 3: Cross-Scenario Generalization

The Cross-Scenario test is more rigorous because each source CSV file is assigned to exactly one split, eliminating any possibility that a flow from the same capture session appears in both training and test. Both models achieve ROC-AUC above 0.998, providing encouraging evidence that their learned patterns transfer across different captures. The two models are tied on Recall (96.59%, 53 false negatives each), while the baseline outperforms on Precision (94.87% vs. 94.22%) and F1 (95.72% vs. 95.39%); this small difference is not statistically significant (McNemar's exact test,  $p = 0.090$ ; Section 4.9). At the level of individual predictions, the two models disagree on 35 of the roughly 15,000 test flows, but their false-negative counts coincide; the hybrid model's extra false positives (92 vs. 81) account for its slightly lower Precision. When the benign traffic distribution shifts between training and test files, Isolation Forest and Local Outlier Factor — both fitted only on the training-split benign traffic — appear to flag some shifted-but-still-benign traffic as anomalous, adding false positives without a compensating Recall gain. The additional anomaly signals therefore offer no measurable benefit here, and a small cost in Precision, without weakening the case for the framework overall: it delineates the conditions under which the framework helps (Section 4.8) from those under which it is, at best, neutral.

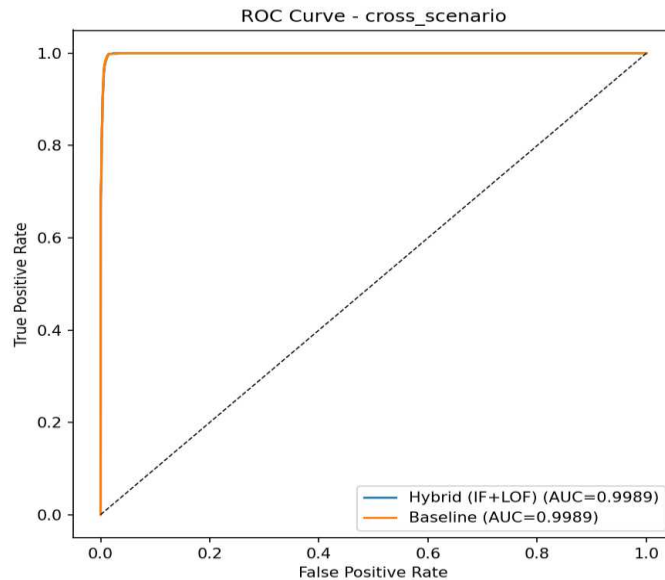


**Fig. 9** ROC curves for the Cross-Scenario setting (Hybrid 0.9989, Baseline 0.9989)

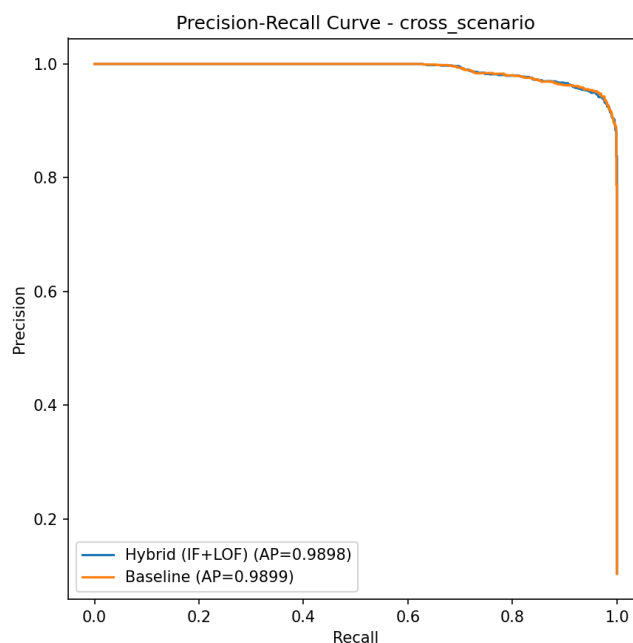


**Fig. 10** Precision-Recall curves for the Cross-Scenario setting (Hybrid AP=0.9898, Baseline AP=0.9899)

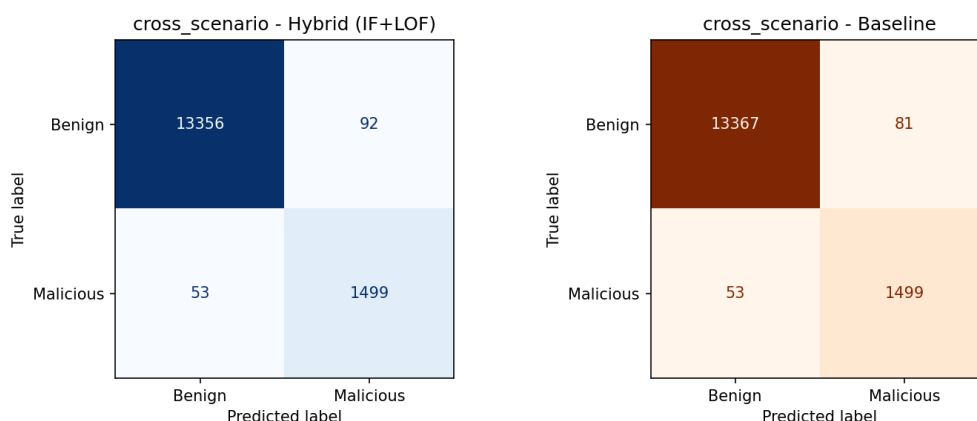


**Fig. 11** Confusion matrices for the Cross-Scenario setting — Hybrid (IF+LOF) (left) and Baseline (right). The two models share an identical false-negative count (53) but differ in false positives (92 vs. 81)

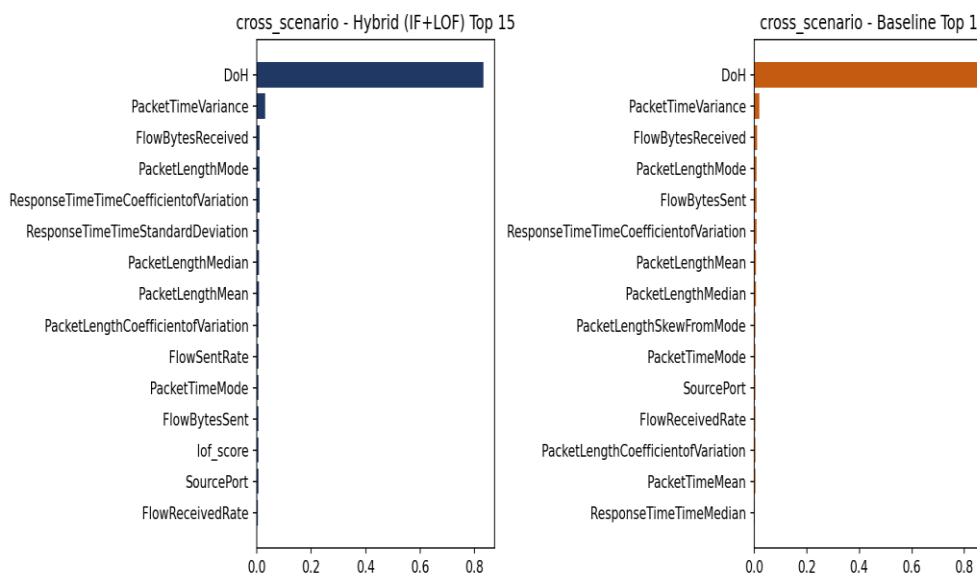


**Fig. 12** Gain-based feature importance for the Cross-Scenario setting. DoH again dominates both models; Packet time variance is the distant second-ranked feature for both, with lof\_score contributing only marginally in the hybrid model

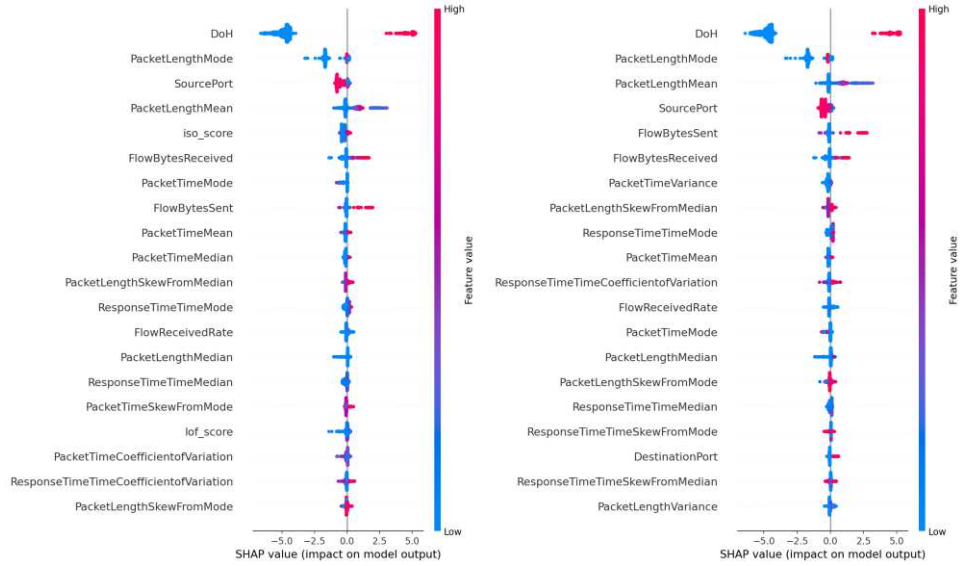


**Fig. 13** SHAP beeswarm plots for the Cross-Scenario setting — Hybrid (IF+LOF) (left) and Baseline (right). `iso_score` ranks fifth by mean  $|\text{SHAP}|$  in the hybrid model, its weakest showing of the three scenarios

#### 4.8 Unseen Attack-Type Robustness: Leave-One-Attack-Out (LOAO) Evaluation

The three preceding scenarios all evaluate detection when the attack type present at test time was also present, in some form, during training. To test resilience against genuinely unseen tunneling variants, we adopt a Leave-One-Attack-Out (LOAO) protocol: for each of the three iodine attack sub-types available in the dataset (null, srv, txt), the model is trained on the remaining two and evaluated exclusively on the held-out type, with the DoH indicator removed throughout to prevent the classifier from relying on it as a shortcut. To rule out the possibility that any single random split favours the hybrid framework, each of the three LOAO configurations is repeated across five random seeds (42, 123, 500, 777, 999), each with an independently re-drawn benign train/validation/test split, for 15 independent evaluations in total. Both models use a fixed decision threshold of 0.5 in this evaluation (Section 3.6), so that the comparison isolates the discriminative value of the anomaly-scoring components rather than any per-scenario threshold tuning.

**Table 4** Leave-One-Attack-Out robustness results (mean  $\pm$  std across 5 random seeds)

| Unseen Attack | Model              | Recall (mean $\pm$ std) | F1-Score (mean $\pm$ std) | False Negatives (mean $\pm$ std) |
|---------------|--------------------|-------------------------|---------------------------|----------------------------------|
| NULL          | Baseline (XGBoost) | 0.9339 $\pm$ 0.0237     | 0.9448 $\pm$ 0.0131       | 178.6000 $\pm$ 168.2938          |
| NULL          | Hybrid (Proposed)  | 0.9773 $\pm$ 0.0115     | 0.9580 $\pm$ 0.0078       | 52.2000 $\pm$ 29.9783            |
| SRV           | Baseline (XGBoost) | 0.9427 $\pm$ 0.0178     | 0.9508 $\pm$ 0.0080       | 127.6000 $\pm$ 52.3765           |
| SRV           | Hybrid (Proposed)  | 0.9820 $\pm$ 0.0050     | 0.9613 $\pm$ 0.0022       | 39.4000 $\pm$ 14.3979            |
| TXT           | Baseline (XGBoost) | 0.9619 $\pm$ 0.0068     | 0.9600 $\pm$ 0.0048       | 101.2000 $\pm$ 87.4340           |
| TXT           | Hybrid (Proposed)  | 0.9902 $\pm$ 0.0048     | 0.9611 $\pm$ 0.0024       | 26.2000 $\pm$ 22.3092            |

Table 4 reports Recall, F1, and False Negatives for each held-out attack type, aggregated as mean  $\pm$  standard deviation across the five seeds. Figure 14 visualizes Recall for both models per unseen attack type, and Figure 15 visualizes the corresponding false-negative counts, both with error bars showing one standard deviation.

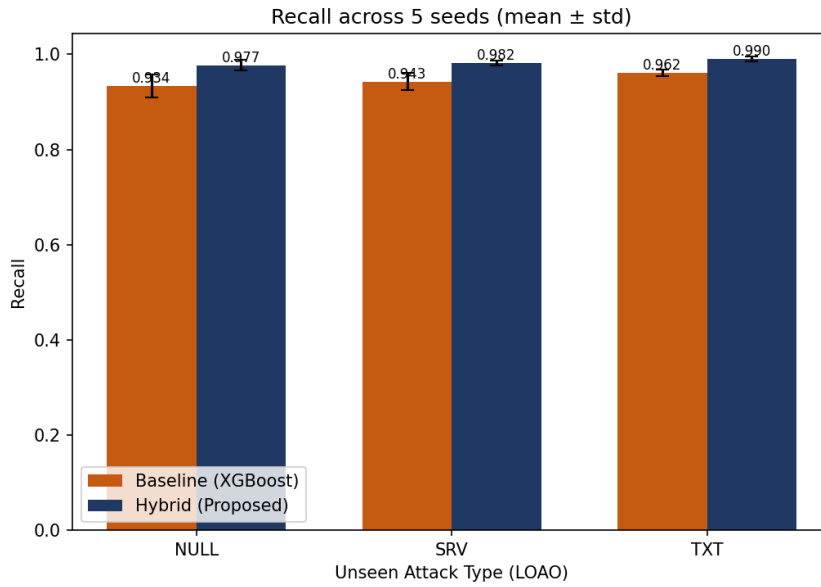


**Fig. 14 Recall** on the held-out (unseen) attack type, mean  $\pm$  std across 5 seeds, Hybrid (IF+LOF) vs. Baseline

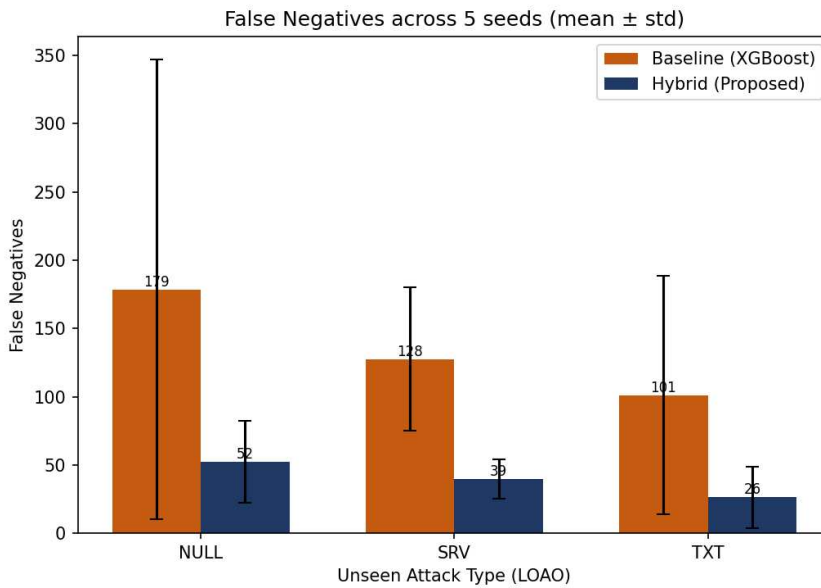


**Fig. 15 False Negatives** on the held-out (unseen) attack type, mean  $\pm$  std across 5 seeds, Hybrid (IF+LOF) vs. Baseline

The hybrid framework reduces false negatives relative to the baseline in every one of the 15 configurations, with no exceptions. Averaged across seeds, false negatives fall from 178.6 to 52.2 for the null attack (70.8% reduction), from 127.6 to 39.4 for srv (69.1% reduction), and from 101.2 to 26.2 for txt (74.1% reduction). Recall improves correspondingly, from the 93–96% range for the baseline to the 98–99% range for the hybrid framework across all three unseen attack types. Figure 16 shows the full per-seed distribution of false negatives rather than only the mean and std: the baseline's spread is wide and occasionally extreme, while the hybrid framework's distribution is both lower and tighter in every case. This combination — a large reduction in mean false negatives together with a large reduction in their variance — is among the clearest evidence in this paper that the anomaly-scoring components genuinely change how the model fails under zero-day conditions, rather than merely shifting a single aggregate statistic.

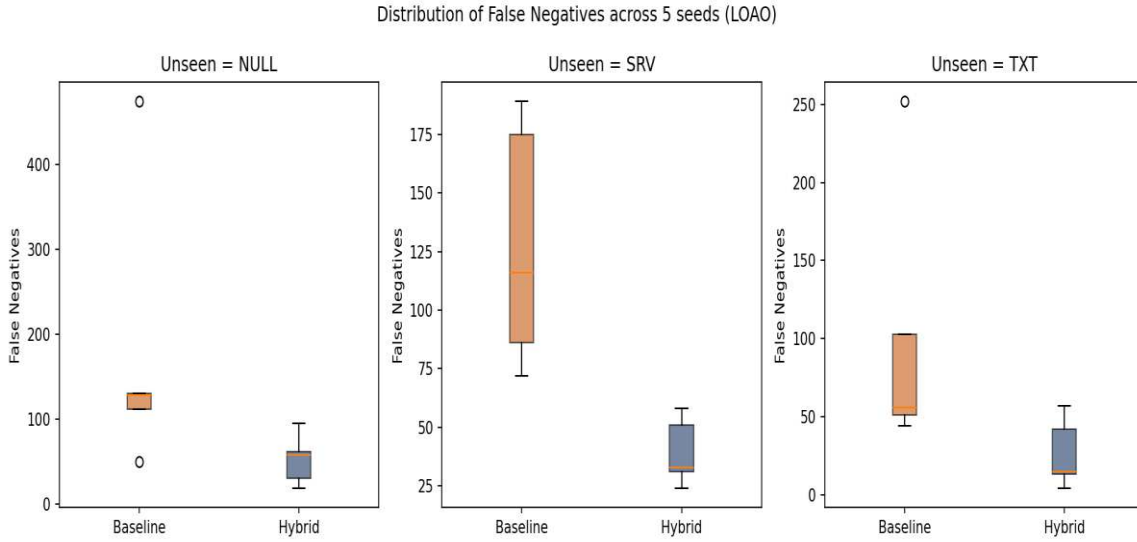


**Fig. 16** Distribution of False Negatives across the 5 seeds for each unseen attack type (LOAO)

These results directly support the central claim of this paper: the value of combining supervised classification with density-aware unsupervised anomaly scoring is concentrated in exactly the setting where it matters most operationally — detecting attack variants that were never seen during training — while offering a smaller, and in the Cross-Scenario setting effectively neutral, advantage when the test distribution closely resembles training. We report this asymmetry directly rather than only the scenarios favorable to the proposed framework.

**Statistical confirmation.** *This central claim is not only large in magnitude but statistically unambiguous. For each held-out attack type, we ran an exact Wilcoxon signed-rank test on the paired false-negative counts (hybrid vs. baseline) across the five seeds. This is the strongest outcome a paired test of this size can produce: the Wilcoxon signed-rank test and the exact sign test both return  $p = 0.031$  for null, srv, and txt alike, the smallest  $p$ -value attainable with five paired seeds. Unlike the primary-scenario comparisons revisited in Section 4.9, where statistical significance holds for only one of three scenarios, the LOAO advantage that motivates this paper is significant — and maximally so — in every condition tested.*

#### 4.9 Statistical Significance of the Primary Scenario Comparisons (McNemar's Test)

The percentage differences reported in Sections 4.5–4.7 describe the size of the gap between the hybrid and baseline models, but not whether that gap is larger than would be expected from chance. To test this directly, we apply the exact McNemar's test to the paired predictions of the two models on the same test flows in each of the three primary scenarios (Table 3). For each scenario, we build the  $2 \times 2$  contingency table of flows where the two models agree or disagree, and compute an exact two-sided binomial test on the discordant pairs (flows where exactly one model is correct).

**Table 5** Exact McNemar's test results for the three primary scenarios

| Scenario       | n (test flows) | Discordant pairs | Hybrid-only correct | Baseline-only correct | p-value | Significant ( $\alpha=0.05$ ) |
|----------------|----------------|------------------|---------------------|-----------------------|---------|-------------------------------|
| Standard       | 15,000         | 46               | 22                  | 24                    | 0.883   | No                            |
| Full Data      | 80,415         | 113              | 69                  | 44                    | 0.024   | Yes (favours hybrid)          |
| Cross-Scenario | 15,000         | 35               | 12                  | 23                    | 0.090   | No                            |

Only the Full Data scenario shows a statistically significant difference between the two models ( $p = 0.024$ ), consistent with it being the one scenario where the hybrid framework outperforms the baseline on every reported metric in Table 3. The Standard scenario shows a numerically large Recall gap (97.54% vs. 95.97%) that is nonetheless not statistically significant ( $p = 0.883$ ): the discordant pairs are nearly evenly split between the two models (22 hybrid-only-correct vs. 24 baseline-only-correct), so this particular gap should be interpreted with caution rather than treated as reliable evidence of superiority. The Cross-Scenario result is similarly not significant ( $p = 0.090$ ), consistent with the near-tie already reported in Section 4.7.

We report all three results, including the two non-significant ones, rather than only the favorable Full Data result. This is deliberate: the paper's central claim of robustness rests on the Leave-One-Attack-Out evaluation in Section 4.8, where the hybrid framework's advantage is large, consistent across all fifteen configurations, and given the scale of the effect relative to the observed seed-to-seed variance in Table 4 considerably more robust than the comparatively small and inconsistently significant differences observed across the three primary scenarios here.

#### 4.10 Mechanistic Analysis: Where the Hybrid Model Rescues the Baseline

The results in Section 4.6 show that in the Full Data scenario, the hybrid model corrects a number of the baseline's false negatives. This raises a natural question: which malicious flows does it rescue, and why? To answer this mechanistically rather than only empirically, we identify every test flow where the baseline XGBoost model misses a malicious flow (a false negative) but the hybrid model catches it correctly. We call these rescued flows. Comparing rescued flows against the malicious flows the baseline catches confidently lets us test a specific hypothesis: that the anomaly-scoring components help precisely on the ambiguous cases sitting close to XGBoost's decision boundary, rather than helping uniformly across all malicious traffic.

To avoid drawing conclusions from a single lucky or unlucky train/test split, this analysis pools rescued flows across all five random seeds used in Section 4.8, rather than relying on one run alone. Table 6 summarizes how many flows were rescued in each seed.

**Table 6 Rescued-flow counts per seed, Full Data scenario**

| Seed                 | Rescued by hybrid | Caught by baseline | Missed by both |
|----------------------|-------------------|--------------------|----------------|
| 42                   | 23                | 7,953              | 144            |
| 123                  | 17                | 7,932              | 164            |
| 500                  | 13                | 7,905              | 147            |
| 777                  | 44                | 7,950              | 135            |
| 999                  | 29                | 7,947              | 130            |
| Pooled (all 5 seeds) | 126               | 39,687             | 720            |

Every seed rescues at least some flows, though the count varies between 13 and 44. This variability is expected it reflects normal run-to-run differences in exactly which flows land near the decision boundary but the fact that every seed rescue flow at all indicates that this is a consistent behavior of the model, not an artefact of one particular split.

Where do rescued flows sit in feature space? Figure 17 (PCA, Principal Component Analysis) and Figure 18 (t-SNE, t-distributed Stochastic Neighbor Embedding) visualize the pooled test set, with rescued flows marked as green stars. In both projections, rescued flows do not form their own isolated region; instead, they appear scattered among several small malicious clusters, frequently overlapping with the flows both

models miss entirely (red crosses). This placement near, but not inside, the “hardest” cluster is consistent with rescued flows being borderline cases rather than either the easy majority or the hopeless minority.

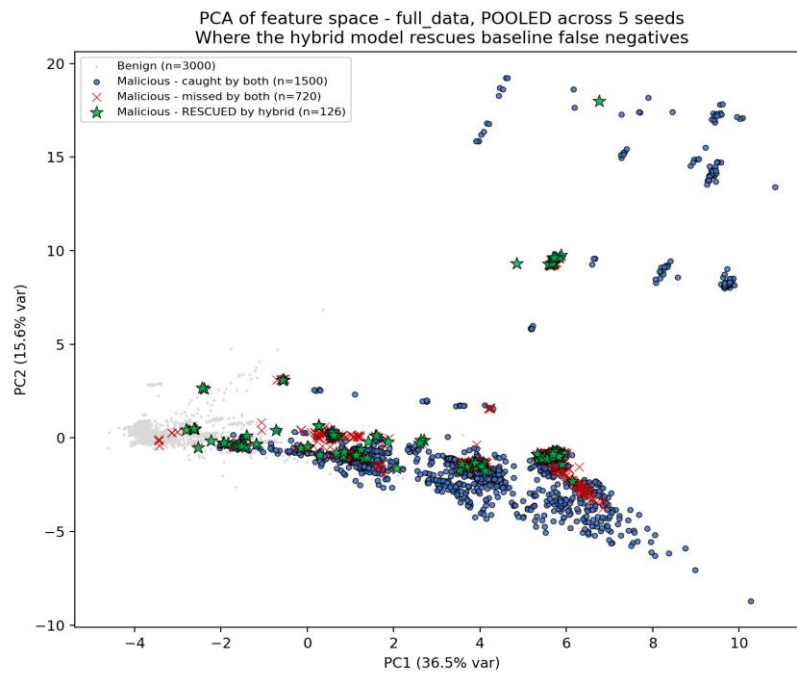


**Fig. 17** PCA of the pooled feature space (5 seeds), Full Data scenario. Rescued flows (green stars) are scattered among several malicious clusters rather than forming their own region

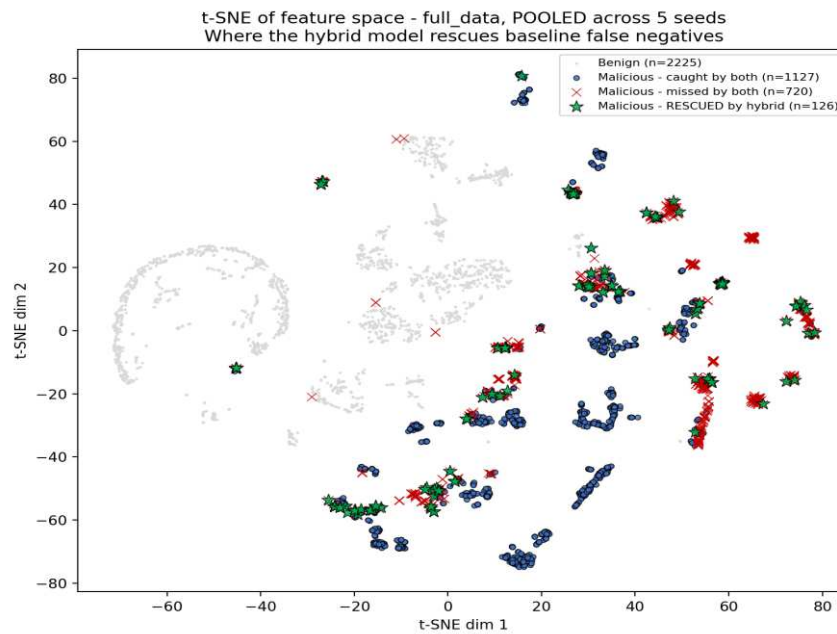


**Fig. 18** t-SNE of the pooled feature space (5 seeds), Full Data scenario. Rescued flows (green stars) frequently sit adjacent to flows missed by both models (red crosses)

Are rescued flows closer to the decision boundary? We test this directly by comparing, for every rescued flow and every confidently-caught flow, how far the baseline model's predicted probability sits from the 0.5 decision threshold. If rescued flows genuinely represent ambiguous cases, their probabilities should cluster near 0.5, while confidently-caught flows should sit near the extremes.

**Table 7 Decision-boundary proximity, pooled across 5 seeds (Full Data scenario)**

| Group                          | n      | Mean $ P(\text{malicious}) - 0.5 $ | Median |
|--------------------------------|--------|------------------------------------|--------|
| Rescued by hybrid              | 126    | 0.074                              | 0.061  |
| Caught confidently by baseline | 39,687 | 0.445                              | 0.499  |

The gap is large: on average, rescued flows sit roughly six times closer to the decision boundary than confidently-caught malicious flows. A Mann-Whitney U test confirms this difference is far beyond what chance would produce ( $U = 42,057$ ,  $p = 1.69 \times 10^{-81}$ ). Because this test pools flows from five independent seeds rather than one, this result does not depend on any single training run.

**Interpretation.** Together, these results support a specific mechanistic explanation for the hybrid model's advantage: it does not improve detection uniformly across all malicious traffic. Instead, it corrects the baseline specifically on flows the baseline itself was already uncertain about — cases sitting near its own decision boundary — while leaving the baseline's confident, easy predictions unchanged. This is exactly the behaviour one would expect if the anomaly scores are supplying genuinely new information on hard cases, rather than merely duplicating what XGBoost already knows. It also explains why the benefit is modest in absolute terms (Section 4.9 shows the improvement in mean false negatives is not always stable across seeds): the pool of “rescuable” flows is inherently small, since it is defined as exactly the flows sitting in the narrow, ambiguous region near the boundary. The value of the anomaly scores lies not in how many flows fall into this region, but in reliably tipping the ones that do towards a correct decision.

#### 4.11 Computational Overhead

Detection accuracy is only part of the practical picture: a framework that is more accurate but substantially slower may not be deployable in settings where traffic must be classified as it arrives. To characterise this trade-off honestly, we benchmark both the baseline XGBoost pipeline and the hybrid pipeline (imputation, scaling, Isolation Forest, Local Outlier Factor, and XGBoost) under two realistic usage patterns: single-flow latency, simulating a streaming deployment that classifies one flow at a time as it arrives, and batch throughput, simulating a deployment that scores flow in fixed-size windows. All measurements were taken on a standard Google Colab CPU runtime; absolute numbers are hardware-dependent, but the relative overhead between the two pipelines is the quantity of interest here.

**Table 8 Inference latency and throughput, Hybrid (IF+LOF) vs. Baseline XGBoost**

| Metric                                        | Baseline (XGBoost) | Hybrid (IF+LOF)   |
|-----------------------------------------------|--------------------|-------------------|
| Single-flow latency (mean $\pm$ std, ms/flow) | $2.39 \pm 2.40$    | $38.53 \pm 17.36$ |
| Single-flow overhead vs. baseline             | —                  | +1511%            |
| Batch throughput (batch size = 2,000)         | 301,854 flows/sec  | 130,109 flows/sec |
| Batch overhead vs. baseline                   | —                  | −2.3× slower      |

The two usage patterns give very different pictures of the overhead. In single-flow mode, the hybrid pipeline is roughly 16 times slower than the baseline (38.53 ms vs. 2.39 ms per flow). This gap is driven almost entirely by Isolation Forest and Local Outlier Factor, both of which incur substantial fixed per-call overhead when scoring a single sample rather than a batch; LOF in particular must compute distances to its stored neighbor set for every call. In batch mode, this fixed overhead is distributed across all flows in the batch, and the gap narrows to a 2.3× slowdown (130,109 vs. 301,854 flows per second) — still slower than the baseline, but comfortably fast enough for near-real-time deployment on windows of accumulated flows rather than one flow at a time.

This result qualifies rather than undermines the framework's practicality. It is not well suited to a deployment that must score individual flows one at a time with minimal buffering, since the anomaly-scoring components introduce meaningful per-call latency in that setting. It remains practical, however, for the more common network-monitoring deployment pattern in which flows are scored in small batches, for example, accumulated over a short time window, where the observed throughput of over 130,000 flows per second is unlikely to be the bottleneck relative to upstream packet capture and feature extraction. We report both figures rather than only the more favorable batch numbers, since the choice between them has direct operational consequences for anyone considering deploying this framework.

#### 4.12 Leave-One-Tool-Out (LOTO): Generalization Across Structurally Different Tunneling Tools

The LOAO evaluation in Section 4.8 tests generalization across query-type sub-variants of a single tunneling tool, iodine (null, srv, txt). This leaves open a stricter question: does the hybrid framework's advantage extend to tunneling tools that are structurally different from iodine, not merely different configurations of it? To test this directly, we repeat the held-out-attack protocol at the tool level rather than the sub-variant level, using three distinct tunneling tools available in the dataset: iodine, dns2tcp, and dnscat2. For each held-out tool, the model is trained exclusively on the other two tools' malicious traffic (plus benign traffic) and evaluated on the held-out tool's traffic in full, with the DoH indicator removed throughout as in LOAO. Each of the three configurations is again repeated across the same five random seeds (42, 123, 500, 777, 999), for fifteen independent evaluations in total, using the same fixed 0.5 decision threshold as LOAO for both models.

**Table 9 Leave-One-Tool-Out robustness results (mean  $\pm$  std across 5 random seeds)**

| Held-out Tool | Model              | Recall (mean $\pm$ std) | F1-Score (mean $\pm$ std) | False Negatives (mean $\pm$ std) |
|---------------|--------------------|-------------------------|---------------------------|----------------------------------|
| IODINE        | Baseline (XGBoost) | 0.7355 $\pm$ 0.0396     | 0.8387 $\pm$ 0.0265       | 1210.2 $\pm$ 480.5               |
| IODINE        | Hybrid (Proposed)  | 0.7618 $\pm$ 0.0267     | 0.8553 $\pm$ 0.0175       | 1100.0 $\pm$ 452.1               |
| DNS2TCP       | Baseline (XGBoost) | 0.2085 $\pm$ 0.1766     | 0.3189 $\pm$ 0.2105       | 7484.6 $\pm$ 2449.2              |
| DNS2TCP       | Hybrid (Proposed)  | 0.3535 $\pm$ 0.3543     | 0.4484 $\pm$ 0.3435       | 6180.2 $\pm$ 3906.7              |
| DNSCAT2       | Baseline (XGBoost) | 0.9213 $\pm$ 0.0141     | 0.9453 $\pm$ 0.0081       | 330.8 $\pm$ 177.6                |
| DNSCAT2       | Hybrid (Proposed)  | 0.9480 $\pm$ 0.0072     | 0.9576 $\pm$ 0.0040       | 220.6 $\pm$ 121.0                |

The hybrid framework outperforms the baseline on both Recall and F1 for all three held-out tools, with false negatives falling by 9.1% for iodine, 17.4% for dns2tcp, and 33.3% for dnscat2. As with LOAO, we test whether this advantage is statistically robust by applying an exact Wilcoxon signed-rank test to the paired false-negative counts across the five seeds, separately for each held-out tool.

**Table 10 Exact Wilcoxon signed-rank test results for LOTO**

| Held-out Tool | Hybrid favoured (of 5 seeds) | Wilcoxon p | Sign-test p |
|---------------|------------------------------|------------|-------------|
| IODINE        | 5/5                          | 0.031      | 0.031       |
| DNS2TCP       | 5/5                          | 0.031      | 0.031       |
| DNSCAT2       | 5/5                          | 0.031      | 0.031       |

The hybrid framework is favoured in every one of the fifteen LOTO configurations (all five seeds), for all three structurally distinct tools — with no exceptions, matching the maximum significance attainable with five paired seeds ( $p = 0.031$ ) for every tool. This directly extends the paper's zero-day claim: the advantage

demonstrated in LOAO is not specific to sub-variants of a single tool, but holds when the held-out tool differs structurally from every tool the model was trained on.

Statistical significance, however, is not the same as adequate absolute performance, and this distinction matters most for dns2tcp. While the hybrid framework's advantage over the baseline is as statistically robust for dns2tcp as for the other two tools, its absolute Recall (35.4%) remains far below an operationally useful level: roughly two-thirds of dns2tcp attacks are still missed even by the better-performing model. We investigated this directly with a feature-level diagnostic comparing dns2tcp's flow statistics against the iodine and dnscat2 traffic the model is trained on. Of 31 features, 16 (52%) show a large standardized distributional difference ( $|\text{diff}| > 0.8$ ) for dns2tcp, concentrated in basic flow statistics such as duration, byte counts, and packet timing — dns2tcp sessions in this dataset are shorter, faster, and transfer substantially less data than iodine or dnscat2 sessions. This indicates dns2tcp represents a genuine feature-domain gap rather than a training or evaluation artefact: even a framework that generalises well across some structurally different tools can still encounter a tool whose traffic statistics differ enough from the training distribution to limit absolute detection quality, regardless of which classifier is used.

We also tested two candidate remedies for dns2tcp specifically: per-tool threshold calibration (as used for the three primary scenarios) and removing the 16 divergent features identified above. Neither proved reliable: calibration degraded performance and eliminated significance for all three tools, while feature removal substantially improved dns2tcp's Recall but at the cost of iodine's and dnscat2's significance, with an improvement large enough to warrant a dedicated data-leakage check before trusting it. We therefore report the original, full-feature, fixed-threshold configuration in Table 9 as the primary LOTO result, leaving a validated remedy for dns2tcp to future work.

## 5. Discussion

### 5.1 Results Analysis

Looking across the three primary scenarios, their statistical significance (Section 4.9), and the LOAO robustness evaluation, a more nuanced picture emerges than a simple "hybrid always wins" narrative. When conditions are comfortable and the test distribution resembles training (the Standard and Full Data scenarios) the hybrid framework matches or modestly outperforms the plain XGBoost baseline, though this improvement is only statistically confirmed in the Full Data scenario. In the Cross-Scenario setting, where the benign traffic distribution itself shifts between training and test files, the two models are essentially tied on Recall, with the baseline holding a (statistically non-significant) advantage on Precision; the anomaly-scoring components do not help here and contribute some additional false positives. What matters most in practice is the LOAO evaluation in Section 4.8: after removing the DoH shortcut and testing on an attack variant absent from training entirely, the hybrid framework shows a substantial and highly consistent advantage, reducing false negatives by roughly two-thirds to three-quarters across all three held-out attack types and all five random seeds, with no exceptions. This is the behavior the design was intended to produce a form of graceful degradation where losing the single strongest detection signal does not cause the whole system to fail, but without pretending the anomaly-scoring components help in every setting. In the Standard scenario, the SHAP analysis in Section 4.5 already shows `iso_score` emerging as the third most influential feature by mean absolute SHAP value once DoH dominates the top rank; in the LOAO evaluation, where DoH is removed entirely and the attack type is unseen, this same reliance on spatial anomaly signals becomes the primary reason false negatives fall so sharply and so consistently for the hybrid framework.

This mechanistic finding (Section 4.10) helps explain why the hybrid framework's benefit, while statistically robust at the level of individual flows, translates into a comparatively modest and sometimes unstable effect on aggregate false-negative counts across seeds (Section 4.8): the pool of boundary-

adjacent, rescuable flows is inherently narrow, so its impact on the overall count is small even when the underlying mechanism is reliable.

### 5.2 Practical Interpretability and Operational Triage

From a practical perspective, SHAP does not only verify the validity of the model but also helps explain the reasoning behind an alert. The logic is clear when an anomalous flow is flagged by the model and the analyst observes that both `iso_score` and `lof_score` are elevated, since the flow appears unusual at both the global and local level. That kind of explanation is more actionable than a raw probability score [9,13]. In security operations, this kind of transparency can be the difference between an alert that gets investigated and one that gets dismissed.

Distinguishing false positives from true attacks in the Cross-Scenario setting is not uniform across all features. `Duration`, `PacketTimeVariance`, and `iso_score` show no reliable difference between the two groups (Mann-Whitney U test,  $p > 0.03$  in all three cases). However, the byte-volume features (`FlowBytesSent`, `FlowBytesReceived`) and `lof_score` show a highly significant divergence ( $p < 10^{-12}$  for all three): in false positives these features push the prediction toward benign on average, while in true positives they push strongly toward malicious.

Figures 19 and 20 illustrate this concretely for one representative flow from each group. For the true positive, `lof_score` and `FlowBytesReceived` contribute +0.21 and +1.42 respectively toward the malicious prediction, reinforcing the DoH signal. For the false positive, the same two features contribute -0.46 and -0.50 — pushing toward benign — with the misclassification instead driven by the dominant DoH contribution alone; consistent with this, the false positive's decision score ( $f(x) = 0.77$ ) sits much closer to the decision boundary than the true positive's ( $f(x) = 6.24$ ), echoing the boundary-proximity pattern already identified in Section 4.10. This offers security operations center (SOC) analysts a narrower, evidence-based triage signal: an alert where `lof_score` and the byte-volume features disagree with the DoH contribution warrants closer manual review before escalation.

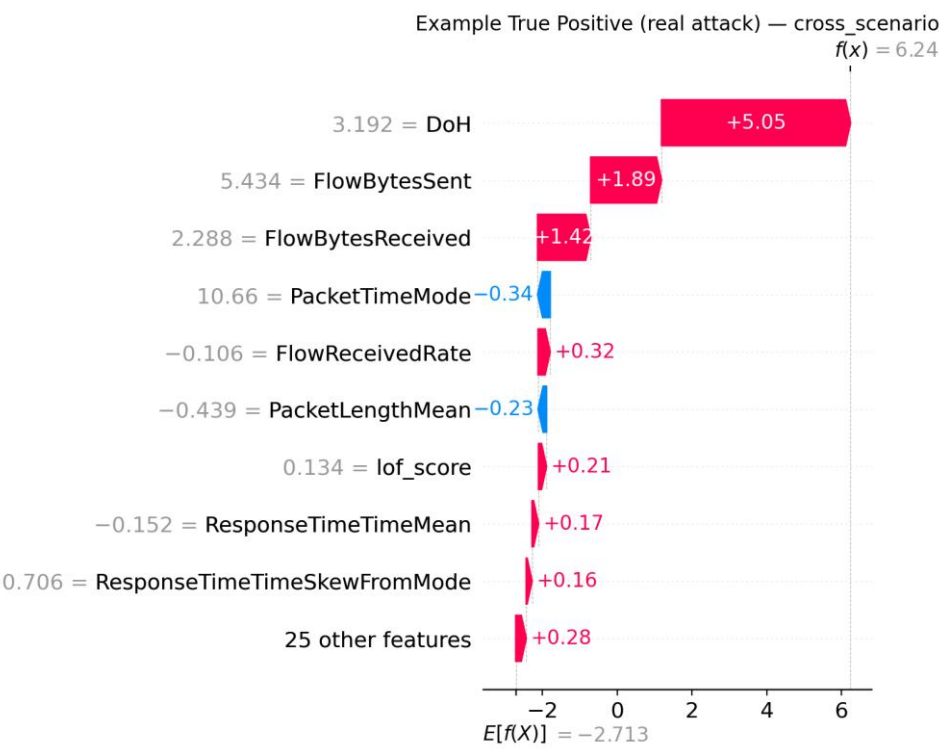


**Fig. 19 SHAP** waterfall plot for a representative true positive flow (Cross-Scenario setting). `lof_score` and `FlowBytesReceived` reinforce the DoH-driven malicious prediction

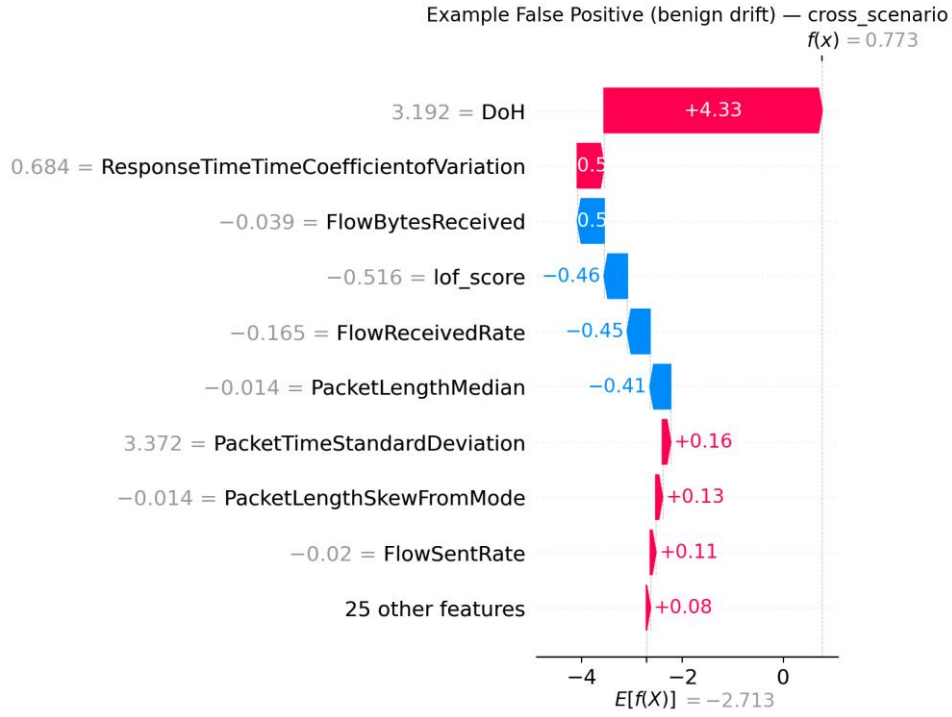


**Fig. 20 SHAP** waterfall plot for a representative false positive flow (Cross-Scenario setting). `lof_score` and `FlowBytesReceived` push toward benign, contradicting the DoH-driven prediction

### 5.3 Limitations

Several aspects leave room for further development. First, the CIRA-CIC-DoHBrw-2020 dataset was gathered under laboratory conditions and does not necessarily fully replicate the traffic diversity, noise, and concept drift found in production environments. Second, attack coverage is relatively limited, consisting of only a few tunneling variants, so further testing across additional attacks and configurations is still required. Third, the Cross-Scenario experiment in Section 4.7 shows that the anomaly-scoring components are not universally valuable: Isolation Forest and Local Outlier Factor can generate false positives when the benign traffic distribution itself shifts between training and test data, so operators should not assume the hybrid framework will outperform a simple classifier in all conditions. Fourth, the Leave-One-Tool-Out evaluation in Section 4.12 shows that the framework's statistical advantage generalises across structurally different tunneling tools, but not that it always reaches adequate absolute performance: `dns2tcp`'s Recall remains below 40% even for the hybrid model, reflecting a genuine feature-domain gap rather than a fixable modelling error, and neither threshold recalibration nor divergent-feature removal produced a reliable, side-effect-free improvement in our tests. Finally, while the evaluation already includes Cross-Scenario splitting, a Leave-One-Attack-Out evaluation, and a Leave-One-Tool-Out evaluation for unseen attack detection, further study of robustness under greater benign-traffic heterogeneity and more adaptive adversarial behavior is required.

## 6. Conclusion

This paper presents a density-aware hybrid machine learning framework, positioned as a batch-processing analyser rather than an inline real-time filter, for detecting malicious DoH tunneling traffic. The framework combines XGBoost classification with spatial anomaly scoring from Isolation Forest and Local Outlier Factor, interpreted via SHAP, evaluated across three primary scenarios, a dedicated statistical significance analysis, and companion Leave-One-Attack-Out (LOAO) and Leave-One-Tool-Out (LOTO) robustness evaluations targeting unseen attack variants and structurally different tunneling tools. On the CIRA-CIC-DoHBrw-2020 benchmark, the framework achieves ROC-AUC above 0.998 under Standard and Full Data conditions, with the hybrid variant improving Recall over the baseline in both, though this improvement is

statistically significant only in the Full Data scenario ( $p = 0.024$ ); in the Cross-Scenario setting the two models are essentially tied. The central result is the Leave-One-Attack-Out evaluation: across fifteen independent configurations, the hybrid framework reduces false negatives relative to the baseline in every configuration, by 70.8-74.1% on average, with a Wilcoxon signed-rank test confirming this at the maximum significance attainable with five paired seeds ( $p = 0.031$ ). A companion Leave-One-Tool-Out evaluation reaches the same maximal significance across three structurally different tunneling tools, though performance remains limited for one tool (dns2tcp) by a genuine feature-domain gap. Three broader points emerge. First, systems drawing on multiple independent sources of evidence are harder to defeat than those relying on a single high-signal feature. Second, unsupervised learning over benign data can generalize to unseen attacks because it learns what normal traffic looks like, rather than memorizing malicious patterns. Third, explainability is not a luxury in security systems: without the ability to explain an alert, analysts cannot triage effectively.

Future work should evaluate the framework on more diverse and recent datasets, explore continual learning strategies that let the anomaly-scoring components adapt as normal traffic evolves, and develop a validated remedy for tools such as dns2tcp whose flow statistics fall genuinely outside the training distribution. We hope the multi-seed LOAO/LOTO evaluation protocol introduced here, together with its statistical significance testing, will be adopted more broadly as a standard for assessing the practical robustness of DoH detection systems.

## List of Abbreviations

|                |                                             |
|----------------|---------------------------------------------|
| <b>DoH</b>     | DNS over HTTPS                              |
| <b>DNS</b>     | Domain Name System                          |
| <b>HTTPS</b>   | Hypertext Transfer Protocol Secure          |
| <b>IETF</b>    | Internet Engineering Task Force             |
| <b>RFC</b>     | Request for Comments                        |
| <b>APT</b>     | Advanced Persistent Threat                  |
| <b>ML</b>      | Machine Learning                            |
| <b>IF</b>      | Isolation Forest                            |
| <b>LOF</b>     | Local Outlier Factor                        |
| <b>SHAP</b>    | SHapley Additive exPlanations               |
| <b>ROC</b>     | Receiver Operating Characteristic           |
| <b>PCA</b>     | Principal Component Analysis                |
| <b>t-SNE</b>   | t-distributed Stochastic Neighbor Embedding |
| <b>AUC</b>     | Area Under the Curve                        |
| <b>PR</b>      | Precision-Recall                            |
| <b>MCC</b>     | Matthews Correlation Coefficient            |
| <b>TP / TN</b> | True Positive / True Negative               |
| <b>FP / FN</b> | False Positive / False Negative             |
| <b>CNN</b>     | Convolutional Neural Network                |
| <b>LSTM</b>    | Long Short-Term Memory                      |
| <b>GRU</b>     | Gated Recurrent Unit                        |

|             |                            |
|-------------|----------------------------|
| <b>LOAO</b> | Leave-One-Attack-Out       |
| <b>LOTO</b> | Leave-One-Tool-Out         |
| <b>IDS</b>  | Intrusion Detection System |
| <b>SOC</b>  | Security Operations Center |

## References

- [1] Hoffman, P., McManus, P.: DNS queries over HTTPS (DoH). IETF RFC 8484 (2018). <https://www.rfc-editor.org/rfc/rfc8484>
- [2] Hou, J., et al.: A survey of DNS tunnel detection. In: Proc. 2022 7th Int. Conf. Signal Image Process. (ICSIP), pp. 338–342 (2022). <https://doi.org/10.1109/ICSIP55141.2022.9886602>
- [3] Dawood, M., et al.: The impact of Domain Name Server (DNS) over Hypertext Transfer Protocol Secure (HTTPS) on cyber security: limitations, challenges, and detection techniques. Comput. Mater. Continua 80, 4513–4542 (2024). <https://doi.org/10.32604/cmc.2024.050049>
- [4] Ji, I.H., et al.: Artificial intelligence-based anomaly detection technology over encrypted traffic: a systematic literature review. Sensors 24, 898 (2024). <https://doi.org/10.3390/s24030898>
- [5] MontazeriShatoori, M., Davidson, L., Kaur, G., Lashkari, A.H.: Detection of DoH tunnels using time-series classification of encrypted traffic. In: Proc. 2020 IEEE Int. Conf. Dependable, Autonomic and Secure Computing (DASC/PiCom/CBDCCom/CyberSciTech), pp. 63–70 (2020). <https://doi.org/10.1109/DASC-PiCom-CBDCCom-CyberSciTech49142.2020.00026>
- [6] Jerabek, K., Hynek, K., Rysavy, O.: Comparative analysis of DNS over HTTPS detectors. Comput. Netw. 247, 110452 (2024). <https://doi.org/10.1016/j.comnet.2024.110452>
- [7] Elaoumari, A.: Evasion-resilient detection of DNS-over-HTTPS data exfiltration: a practical evaluation and toolkit. arXiv:2512.20423 (2025)
- [8] Amirov, N., Isik, B., Tuncer, B.I., Bahtiyar, S.: DNS tunneling: threat landscape and improved detection solutions. arXiv:2507.10267 (2025)
- [9] Zebin, T., Rezvy, S., Luo, Y.: An explainable AI-based intrusion detection system for DNS over HTTPS (DoH) attacks. IEEE Trans. Inf. Forensics Secur. 17, 2339–2349 (2022). <https://doi.org/10.1109/TIFS.2022.3183390>
- [10] Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, pp. 785–794 (2016). <https://doi.org/10.1145/2939672.2939785>
- [11] Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: Proc. 8th IEEE Int. Conf. Data Mining (ICDM), pp. 413–422 (2008). <https://doi.org/10.1109/ICDM.2008.17>
- [12] Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: Proc. 2000 ACM SIGMOD Int. Conf. Management of Data, pp. 93–104 (2000). <https://doi.org/10.1145/342009.335388>
- [13] Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 30, pp. 4765–4774 (2017)
- [14] Lyu, M., Gharakheili, H.H., Sivaraman, V.: A survey on DNS encryption: current development, malware misuse, and inference techniques. ACM Comput. Surv. 55, 1–28 (2022). <https://doi.org/10.1145/3547331>
- [15] Wu, S., Wang, W., Ding, Z.: Detecting malicious DoH traffic: leveraging small sample analysis and adversarial networks for detection. J. Inf. Secur. Appl. 84, 103827 (2024). <https://doi.org/10.1016/j.jisa.2024.103827>
- [16] Vekshin, D., Hynek, K., Cejka, T.: DoH insight: detecting DNS over HTTPS by machine learning. In: Proc. 15th Int. Conf. Availability, Reliability and Security (ARES), pp. 1–8 (2020). <https://doi.org/10.1145/3407023.3409192>
- [17] Mitsuhashi, R., Satoh, A., Jin, Y., Iida, K., Shinagawa, T., Takai, Y.: Identifying malicious DNS tunnel tools from DoH traffic using hierarchical machine learning classification. In: Information Security (ISC 2021). LNCS, vol. 13118, pp. 238–256. Springer, Cham (2021). [https://doi.org/10.1007/978-3-030-91356-4\\_13](https://doi.org/10.1007/978-3-030-91356-4_13)
- [18] Nguyen, T.A., Park, M.: DoH tunneling detection system for enterprise network using deep learning technique. Appl. Sci. 12, 2416 (2022). <https://doi.org/10.3390/app12052416>
- [19] Mirsky, Y., Doitshman, T., Elovici, Y., Shabtai, A.: Kitsune: an ensemble of autoencoders for online network intrusion detection. In: Proc. Network and Distributed System Security Symposium (NDSS) (2018). <https://doi.org/10.14722/ndss.2018.23204>

- [20] Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. *Science* 313, 504–507 (2006). <https://doi.org/10.1126/science.1127647>
- [21] Hussein, S.A., Répás, S.R.: A hybrid intrusion detection framework using deep autoencoder and machine learning models. *AI* 7, 39 (2026). <https://doi.org/10.3390/ai7020039>
- [22] Sajid, M., Malik, K.R., Almogren, A., Malik, T.S., Khan, A.H., Tanveer, J., Rehman, A.U.: Enhancing intrusion detection: a hybrid machine and deep learning approach. *J. Cloud Comput.* 13, 123 (2024). <https://doi.org/10.1186/s13677-024-00685-x>
- [23] Erskine, S., Suresh, A.: Explainable hybrid CNN-XGBoost framework for multi-class IoT intrusion detection with leakage-aware feature selection. *Appl. Syst. Innov.* 9, 49 (2026). <https://doi.org/10.3390/asi9030049>
- [24] Almuhanha, R., Dardouri, S.: A deep learning/machine learning approach for anomaly based network intrusion detection. *Front. Artif. Intell.* 8 (2025). <https://doi.org/10.3389/frai.2025.1625891>
- [25] Zhan, M., Li, Y., Yu, G., Li, B., Wang, W.: Detecting DNS over HTTPS based data exfiltration. *Comput. Netw.* 209, 108919 (2022). <https://doi.org/10.1016/j.comnet.2022.108919>
- [26] Sharma, N., Swarnkar, M.: OptiTuneD: an optimized framework for zero-day DNS tunnel detection using N-grams. In: *Proc. 2023 IEEE Int. Conf. Advanced Networks and Telecommunications Systems (ANTS)*, pp. 177–182 (2023). <https://doi.org/10.1109/ANTS59832.2023.10469430>
- [27] Talabani, H.S., Abdul, Z.K., Mohammed Saleh, H.M.: DNS over HTTPS tunneling detection system based on selected features via ant colony optimization. *Future Internet* 17, 211 (2025). <https://doi.org/10.3390/fi17050211>
- [28] Hussein, S., et al.: Enhancing intrusion detection in encrypted DoH traffic through a robust ensemble learning framework. *PLoS ONE* (2025). <https://doi.org/10.1371/journal.pone.0345880>
- [29] Jung, W.K., Kwak, B.I.: MTL-DoHTA: multi-task learning-based DNS over HTTPS traffic analysis for enhanced network security. *Sensors* 25, 993 (2025). <https://doi.org/10.3390/s25040993>
- [30] Sammour, M., Othman, M.F.I., Hassan, A., Bhais, O., Talib, M.S.: Advanced DNS tunneling detection: a hybrid reinforcement learning and metaheuristic approach. *Front. Comput. Sci.* 7, 1728980 (2026). <https://doi.org/10.3389/fcomp.2025.1728980>
- [31] Pedregosa, F., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830 (2011)

## Statements and Declarations

### *Funding*

Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

### *Competing Interests*

The authors have no relevant financial or non-financial interests to disclose.

### *Author Contributions*

Enas Selem contributed to methodology, including model design and implementation, results validation and analysis, figures preparation, results discussion, and conduction of all experiments; she also reviewed the manuscript and revised it to ensure the accuracy and quality of the final version. Rania Salama contributed to the proposal of the model and the selection and definition of the techniques used to describe the environment and construct the system model; she also contributed to reviewing the manuscript to assure that the paper was prepared and presented with the highest possible quality. All authors read and approved the final manuscript.

### *Data Availability*

The dataset supporting the conclusions of this article, CIRA-CIC-DoHBrw-2020, is publicly available from the Canadian Institute for Cybersecurity (CIC), University of New Brunswick.

### *Ethics Approval and Consent to Participate*

Not applicable. This study did not involve human participants, human data, or animals.

### *Consent for Publication*

Not applicable.