

Detecting Zero-Day Attacks via Reconstruction of Feature Influence and Model Uncertainty

Hussein Fawaz¹, Student Member, IEEE, Jacopo Talpini², Marco Savi², Silvia Giordano²,
 and Omran Ayoub², Member, IEEE

Abstract—In practical Network Intrusion Detection System (NIDS) deployments, detecting anomalies is only the first step, while determining the exact nature of those anomalies is equally important. Commonly, anomalous traffic is forwarded to a supervised multiclass classifier trained to identify known attack categories. While effective for known threats, this step presents a significant limitation, as zero-day attacks can be misclassified as known attacks. Therefore, there is a need for approaches that go beyond standard classification and can reliably recognize when an input does not conform to any learned attack pattern, i.e., zero-day attacks. To tackle this problem, we propose a novel detection strategy that leverages per-instance feature importance scores from an explainable Artificial Intelligence (XAI) framework and prediction uncertainty estimates derived from an ensemble classifier. To evaluate our approach, we conduct extensive experiments using a leave-one-attack-out strategy across three benchmark datasets, CICIoT2023, NF-TON-IoT, and CIC-DDoS2019, and test performance under two underlying classifiers, namely XG-Boost and Random Forest, demonstrating the model-agnostic nature of our method. Experimental results show that our approach achieves best-case AUROC gains approaching 40% and F1-score improvements of up to 73%, while maintaining positive or near-neutral worst-case performance across datasets, highlighting the effectiveness and robustness of jointly modeling explanation-driven reconstruction error and predictive uncertainty for reliable zero-day threat identification.

Index Terms—Network Intrusion Detection, Explainable AI, Uncertainty Quantification, Zero-day Attacks

I. INTRODUCTION

Network intrusion detection systems (NIDS) are essential components of cybersecurity infrastructures, designed to monitor and analyze network traffic for signs of malicious behavior [1], [2]. As modern networks grow in complexity and scale, the detection of cyber threats becomes increasingly challenging. In this context, machine learning (ML) techniques have become a cornerstone of NIDS development, offering scalable, data-driven approaches for detecting and identifying threats [3], [4]. This effectiveness stems from their ability to learn complex patterns from large network traffic volumes,

enabling ML-based NIDS to significantly improve intrusion detection performance [5], [6].

Despite their strong performance in terms of classification scores, several challenges remain, particularly in practical, safety-critical environments. Among the most significant are limited generalization to unseen conditions, high false positive rates, and the intrinsic difficulty of detecting previously unseen (zero-day) attacks [7]. Specifically, zero-day attacks, i.e., attacks that exploit previously unknown vulnerabilities, are, by definition, absent from historical training data [8]. Consequently, ML models trained using supervised learning methods, which are commonly employed in signature-based NIDS [9], typically fail to identify such attacks, as those rely on labeled instances of known attack types. Accordingly, zero-day attack detection is commonly formulated as an anomaly detection problem, where deviations from established normal or known-attack patterns are identified [10].

Common strategies to address anomaly detection involve using unsupervised learning methods [11], [12] or semi-supervised learning methods [13]. These methods train ML models solely on benign traffic to learn compact representations of normal traffic behavior. Consequently, these models can flag deviations from learned patterns, potentially indicating anomalous or malicious activity.

Beyond merely detecting anomalies, accurately identifying the nature of the threat is a critical next step for deciding the appropriate countermeasure. In many real-world scenarios, once traffic has been flagged as suspicious, it is typically passed to a multiclass classifier trained using a supervised learning approach. This classifier is tasked with determining whether the anomaly matches any known attack (i.e., any known intrusion category) [14]. This attack-identification step, while effective for known threats, introduces a vulnerability: novel attacks (i.e., flows that have been correctly flagged as anomalies but belong to a previously unseen category of attacks) may be mistakenly classified as known ones [15]. In such cases, the NIDS must raise an alert indicating a potential zero-day threat. Accordingly, NIDS should not only employ classification models that discriminate among known attack classes, but also incorporate explicit mechanisms at the attack-identification stage to identify inputs that do not conform to any previously observed attack pattern, thereby signaling the potential presence of zero-day threats.

Figure 1 shows a scheme representing the scope of our focus in a NIDS pipeline. At a high level, the detection process is divided into two sequential stages. First, an anomaly detection model trained solely on benign traffic identifies

H. Fawaz is with the University of Applied Sciences and Arts of Southern Switzerland (SUPSI), Switzerland, and Università della Svizzera italiana (USI), Switzerland (e-mail: hussein.fawaz@usi.ch).

J. Talpini is with the University of Applied Sciences and Arts of Southern Switzerland (SUPSI), Switzerland, and the Department of Informatics, Systems and Communication (DISCO), University of Milano-Bicocca, Italy.

M. Savi is with the Department of Informatics, Systems and Communication (DISCO), University of Milano-Bicocca, Italy.

S. Giordano and O. Ayoub are with the University of Applied Sciences and Arts of Southern Switzerland (SUPSI), Switzerland.

Corresponding author e-mail: hussein.fawaz@usi.ch

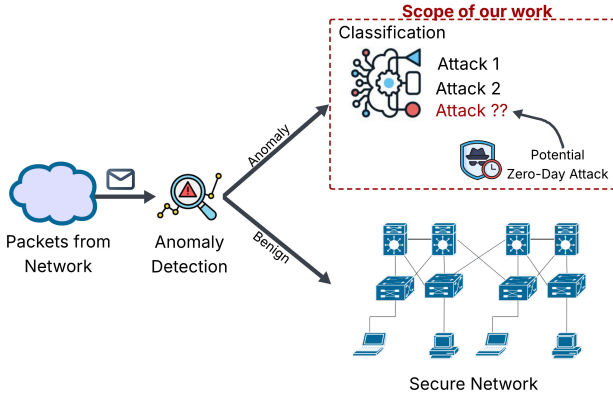


Fig. 1: Schematic representation of our main focus in the NIDS pipeline

instances that deviate from normal behavior using unsupervised learning techniques. Second, a supervised multiclass classifier attempts to assign each flagged anomaly to one of the known attack categories. We focus on the specific challenge of distinguishing zero-day attacks from previously known attacks at the attack-identification stage. Specifically, we aim to detect cases in which traffic that is first detected as anomalous and then classified by an ML-based multiclass classifier as a known attack type in fact corresponds to a zero-day attack. Even though such traffic has already been recognized as malicious, distinguishing whether it corresponds to a known or previously unseen attack remains crucial for informed operational decision-making after detection and the selection of appropriate countermeasures.

To this end, we propose a novel methodology that leverages the *per-instance feature influence scores* produced via explainable artificial intelligence (XAI), specifically Shapley Additive Explanations (SHAP) [16], which capture how each feature contributes to the classifier’s decision, and the *epistemic uncertainty* associated with each prediction, i.e., the uncertainty stemming from the model’s own lack of knowledge about unseen inputs, quantified via ensemble-based entropy decomposition [17]. We integrate these two complementary signals into a reconstruction–uncertainty–based detection mechanism: feature influence vectors are reconstructed using an autoencoder (AE) trained on known attacks, and their reconstruction error (RE) is jointly modeled with uncertainty estimates inside a density-based model, more specifically a Gaussian Mixture Model (GMM). Our rationale is that the feature influence patterns for instances belonging to previously seen (i.e., known) classes will differ systematically from those associated with zero-day attacks. Specifically, we expect that the explanation profiles of known-class samples will align with learned decision boundaries, whereas those of novel attacks will deviate noticeably. Similarly, we hypothesize that the classifier will exhibit higher predictive uncertainty when confronted with unseen classes, in contrast to the typically lower uncertainty observed for familiar classes in the training data. Using both signals ensures that zero-day attacks are detected even when either signal alone fails to flag them. More broadly, our proposal operates in an *open-set* regime, where the classifier is exposed at inference time to attack classes that were not observed during training and must therefore

identify inputs that do not belong to any known category. To evaluate our approach, we train Random Forest (RF) and XGBoost (XGB) classifiers on known attack categories and assess performance through a *leave-one-attack-out* strategy on three benchmark datasets (CICIoT2023, NF-TON-IoT, and CIC-DDoS2019), demonstrating consistent improvements over baseline methods with AUROC gains of up to 40%, average improvements of 4–15%, worst-case degradations below 3%, and F1-score gains ranging from 10% to over 70%.

The contributions of our work can be summarized as:

- We address the problem of zero-day attack detection at the attack identification stage, aiming to identify attacks that have been incorrectly classified by a multiclass classifier as belonging to a known threat category.
- We propose a novel, lightweight, and model-agnostic detection framework that leverages two per-instance quantities: feature influence scores obtained from an XAI framework and predictive uncertainty, both computed without requiring changes to the base model architecture or feature representation.
- We introduce an explanation-based reconstruction mechanism in which an autoencoder is trained on feature-attribution vectors of known attacks to capture typical explanation patterns and jointly model the resulting RE with epistemic uncertainty using a GMM to identify deviations from known-class behavior.
- We present the complementary roles of explanation-based signals and prediction uncertainty for zero-day detection, showing that neither signal alone is sufficient across models and scenarios, while their joint modeling enables reliable detection even when one of the two fails.

The remainder of this paper is structured as follows. Section II presents necessary background information. Section III discusses related work. Section IV details our proposed approach. Sections V and VI present the evaluation settings and experimental results, respectively. Finally, Section VII concludes the paper.

II. BACKGROUND

This section provides a brief overview on XAI, with focus on SHAP, and uncertainty quantification.

A. Explainable Artificial Intelligence

Explainable Artificial Intelligence (XAI) refers to a set of techniques and methods designed to make the decisions of ML models interpretable to humans [18]. The main objective of XAI is to provide insights into how an ML model arrives at a particular prediction, which is essential for enhancing transparency. Explanations generated by XAI methods can take various forms, including feature importance scores that quantify the influence of each input feature to a prediction, and counterfactual explanations that describe how input values could be changed to alter the model’s output. Moreover, there exist two categories of explanations, namely, local explanations, which aim to interpret individual predictions, and global explanations, which seek to describe the overall behavior of the model across the entire dataset.

Several well-known XAI frameworks support the generation of these explanations, including SHAP [16]. SHAP offers a game-theoretic approach to interpreting ML predictions by fairly distributing the “payout” (i.e., the prediction) among features based on their individual influence to the prediction. As input, SHAP takes a trained ML model and a specific data instance to be explained. As output, SHAP provides each feature’s importance by comparing the model’s output with and without that feature, resulting in a vector of feature importance scores, where each value represents the contribution of a single feature to the final prediction.

Following this approach, SHAP provides individual (per-instance) explanations, allowing practitioners to observe how features influence the predictions made by the underlying ML model. In multiclass classification tasks such as the one addressed in this work, this enables the comparison of feature importance patterns across different predicted classes. For example, the same feature may be highly influential for predicting one class but have minimal or even opposite influence for another. These differences arise because models often rely on a unique combination of features when distinguishing among the diverse classes. As such, SHAP values reveal not only the magnitude but also the directionality (positive or negative contribution) of each feature’s influence on a specific class prediction. By analyzing these class-dependent feature importance vectors, practitioners can uncover discriminative patterns that the model has learned. In our work, we leverage this information to flag potential zero-day attacks by focusing on the most relevant features used to distinguish known attack classes and by analyzing the resulting attribution patterns at decision time. Our central hypothesis is that, when confronted with previously unseen attacks, the model exhibits feature influence patterns that deviate systematically from those observed for known attack classes. Such deviations, captured through reconstructed feature attributions, can therefore serve as reliable signals for identifying zero-day attack behavior.

B. Uncertainty Quantification

Since intrusion detection can be considered a safety-critical task, there is a strong need for models that are trustworthy or reliable [19]. Model trustworthiness goes beyond guaranteeing high accuracy in terms of model prediction with test samples statistically similar to the data used for model training, and requires that the considered model performs consistently across different settings. More specifically, in the considered scenario, we expect a reliable ML model to show a high level of uncertainty for inputs pertaining to new classes (i.e., new kinds of attacks). In the following, we discuss some concepts on uncertainty quantification (UQ) that will be used in our proposed pipeline.

Uncertainty in model predictions can be decomposed into two contributions: aleatoric uncertainty and epistemic uncertainty. Aleatoric uncertainty takes into account the inherent randomness in the input data and is typically high for ambiguous inputs. In contrast, epistemic uncertainty reflects the model’s lack of knowledge, and it is typically high for samples that are out of distribution with respect to the training set.

To formalize these concepts, let us consider a common supervised classification problem in which a set $\mathcal{D}$ of N input–output pairs, $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, is given. The aim is to define a function h (e.g., a neural network or decision tree) that provides the conditional probability distribution $p(\mathbf{y}|\mathbf{x}, h)$ over K classes for a given input $\mathbf{x}$ [20]. In this setting, a common and effective way to capture different sources of uncertainty is to consider an ensemble of models [21]. For tree-based models, common ensembling approaches include bagging and boosting, which produce a set of classifiers $\{h_1, \dots, h_M\}$, each providing a categorical distribution over the output classes [21]. The final ensemble prediction is obtained by averaging the prediction probabilities of its members and is denoted by $p(\mathbf{y}|\mathbf{x}, \mathcal{D})$. Its total uncertainty can be quantified using Shannon entropy $\mathbb{H}$, which is maximized for a uniform distribution over the classes (i.e., the most uncertain scenario). Moreover, the total uncertainty can be further decomposed [22], as follows:

$$\underbrace{\mathbb{H} \left[\frac{1}{M} \sum_{i=1}^M p(\mathbf{y}|\mathbf{x}, h_i) \right]}_{\text{Total Uncertainty}} = \underbrace{\mathbb{I}[\mathbf{y}, h|\mathbf{x}, \mathcal{D}]}_{\text{Model Uncertainty}} + \underbrace{\frac{1}{M} \sum_{i=1}^M \mathbb{H}[p(\mathbf{y}|\mathbf{x}, h_i)]}_{\text{Aleatoric Uncertainty}} \quad (1)$$

where $\mathbb{I}$ is the information gain between the model and output and captures model uncertainty, while the second term is the aleatoric uncertainty, computed as the mean entropy of the predictions obtained from the ensemble members [23]. The overall intuition is that model uncertainty is high when ensemble members disagree by assigning different, highly confident predictions to a given input.

III. RELATED WORK

This section reviews related work on anomaly-based intrusion detection, uncertainty quantification and XAI for NIDS.

A. Anomaly-Based Intrusion Detection

Numerous works have explored anomaly-based intrusion detection for identifying zero-day attacks, using techniques ranging from traditional supervised ML to unsupervised deep learning [11]–[14], [24], [25].

Unsupervised deep learning (DL) methods have been adopted to identify deviations as anomalies. For instance, [24] introduced a DNN-based anomaly detection method that employs a feature-selection mechanism and uses generative adversarial networks (GANs) to synthesize packet-level traffic for underrepresented attack classes. In [25], the authors proposed a real-time NIDS via an ensemble of AEs in software-defined networking (SDN). The authors in [26] introduced a continual-learning intrusion detection framework that employs a Bayesian neural network to preserve previously acquired knowledge while adapting to new data, combined with a generative multiclass classifier that incrementally incorporates and classifies newly emerging attack types. In [30], the authors

investigate optimized conventional ML and DL models for intrusion anomaly detection and propose a vision-transformer architecture designed to capture complex patterns.

More recent works have focused on adding noise to the input layers of AEs, known as denoising autoencoders (DAEs), to improve anomaly detection accuracy in network traffic data. For instance, in [27], the authors proposed a NIDS combining a DAE with Wasserstein GANs to process high-dimensional and imbalanced traffic data, while [28] employed DAEs for automatic feature extraction and unsupervised representation learning. The DAEs in these works aim to reconstruct clean patterns from corrupted inputs (e.g., packet flows) when detecting anomalies. The authors in [29] combine an AE-based anomaly detector with RF and XGB classifiers and evaluate the resulting hybrid models on unseen samples, showing that the anomaly-detection stage substantially improves performance over the standalone classifiers. In addition, [31] introduce a similar paradigm where they train a normalizing flows model on the internal features of a classification model to identify out-of-distribution (i.e., zero-day attack) inputs (albeit on the packet-level). [32] use diffusion models to model the latent space from VAE on flow-level data. More recently, [10] proposed EFC, an energy-based flow classifier for open-set classification, which estimates per-class probability distributions and computes flow energy to measure how likely a flow belongs to a given class.

GMMs have also been explored for anomaly-based intrusion detection, often in combination with deep learning models for feature extraction or latent-space density estimation [33]–[36]. In [33], the authors integrate GMMs with bi-directional LSTMs to incrementally separate unidentified traffic, feeding newly detected clusters back into the training pipeline. Similarly, [34] employ Pearson correlation for linear dimensionality reduction and stacked sparse AEs for nonlinear compression, using GMMs to estimate sample energy and flag abnormal traffic through learned reconstruction-error thresholds. Other works focus on improving the robustness of latent representations before GMM density estimation. For instance, [35] propose an unsupervised ensemble of autoencoders whose attention-based latent representations are modeled by GMMs using expectation–maximization, enabling the system to adapt across domains with skewed distributions. More recently, [36] integrates autoencoders with GMMs and one-class Support Vector Machines (SVMs), where the AE extracts representative features from normal traffic and the semi-supervised detectors jointly identify deviations in unseen samples.

These approaches are mostly employed to detect attacks by identifying anomalous instances that deviate from benign samples. In our work, we identify anomalies at the attack-identification stage, i.e., we tackle an open-set classification problem without changing the existing base classifiers or feature pipelines. Unlike prior approaches that model the data or latent feature distribution of network traffic, our framework instead validates classifier decision behavior using SHAP-based feature-attribution reconstruction and predictive uncertainty, enabling the detection of predictions inconsistent with known attack patterns.

B. Uncertainty Quantification for Intrusion Detection

Few works have leveraged uncertainty quantification for intrusion detection, with the aim of enhancing the reliability of the employed approaches by addressing overconfidence in predictions and adopting a more trustworthy decision making. For instance, [15] proposed an uncertainty-aware Bayesian Neural Networks (BNN) model to avoid overconfident predictions in closed-set classification settings and identify unknown threats through Out-of-Distribution (OoD) detection. In [37], the authors proposed a packet-level uncertainty-quantified deep learning model that used Hamiltonian Monte Carlo to detect OoD cyber attacks and introduced new UQ scoring measures to evaluate predictive certainty. Similarly, [38] investigated methods for extending pretrained classifiers by incorporating UQ techniques and proposed a distance-based approach for computing OoD scores. Moreover, [39] explored the use of Bayesian Autoencoder (BAE) models for UQ, proposing a heteroscedastic aleatoric uncertainty modeling approach that jointly captures both aleatoric and epistemic uncertainty. Additionally, [40] proposed a Bayesian explanatory model for IDS, allowing feature-based explainability in BAEs with UQ to generate interpretable local and global explanations with uncertainty-aware scoring. More recently, the authors in [41] introduced a method that improves predictive accuracy and UQ using Bayesian ensembles, leveraging CNN–LSTM models with Monte Carlo (MC) Dropout to estimate uncertainty through averaged predictions and standard deviations across stochastic forward passes.

The above-mentioned works have shown that UQ can be leveraged to identify potentially incorrect predictions, either in the context of anomaly detection or classification of attacks, by imposing a threshold on the level of acceptable uncertainty of model’s decisions. Similarly, our work leverages UQ, but differs from previous approaches by not relying on uncertainty alone, but incorporating it as an input feature, combined with per-instance feature importance scores to form a richer representation for potentially identifying zero-day attacks.

C. XAI for Intrusion Detection

The application of XAI for ML-based NIDS has been prevalent recently, primarily to enhance the transparency of the underlying ML models and to extract insights from model behavior such as by identifying which are the most influential features [42]–[50].

For instance, [44] apply various XAI methods such as SHAP, and Local Interpretable Model-Agnostic Explanations (LIME) to DL-based IDS to quantify feature importance and identify key features impacting model’s decisions. In [45], the authors combine multiclass anomaly detection with defect-type classification incorporating seven interpretability methods (including SHAP and LIME) for feature importance analysis, with the aim of extracting insights to the model’s decision process. Similarly, [46] employ CNN models using both LIME and SHAP to interpret model decisions by identifying critical features and their influence on predictions. In [47], authors apply LIME to provide local explanations for an SVM model to understand the model’s prediction and provide insights

behind its decision-making, while [48] leveraged SHAP to extract explanations and reveal the most crucial features for attack detection. Moreover, in [49], the authors apply SHAP for global feature importance analysis alongside a DAE for feature selection, while addressing data imbalance. Recently, [50] propose an approach that integrates a calibration-oriented loss function into XGB together with SHAP-based recursive feature elimination to improve both predictive performance and the quality of confidence estimates.

In our work, we leverage SHAP-based per-instance feature importance scores to identify deviations in feature influence patterns that may indicate zero-day attacks. While [51] similarly trains an AE on SHAP attributions from a binary XGB classifier and relies solely on attribution reconstruction, we additionally integrate epistemic uncertainty into the detection process, enabling more robust zero-day identification.

Table I summarizes existing zero-day detection methods in terms of classification type (binary or multiclass), signal type (e.g., reconstruction error, density estimation, feature attribution, or predictive uncertainty), and the stage at which detection occurs (pre-classification or attack identification).

TABLE I: Comparison between different zero-day intrusion detection approaches.

Method	Cls Type	Signal Type	Stage
AE-based [24], [27], [28]	Binary	Reconstruction	Pre-classification
GMM-based [33]–[35]	Binary	Density estimation	Pre-classification
SHAP-AE [51]	Binary	Feature attribution	Attack identification
Energy-based [10]	Multi	Energy Flows	Pre-classification
Flow [31]	Multi	Likelihood (flow)	Pre-classification
Diffusion [32]	Multi	Generative diffusion	Pre-classification
UQ-based [15], [37], [41]	Multi	Predictive uncertainty	Attack identification
Ours	Multi	Attribution + UQ	Attack identification

IV. PROBLEM STATEMENT AND METHODOLOGY

We consider a realistic deployment scenario for NIDS, in which network traffic is continuously monitored and evaluated in multiple stages to detect and classify potential security threats. In particular, we assume that a set of known attacks is available during training while previously unseen attacks may emerge only after deployment, without prior labeled examples. In this scenario, the objective extends beyond accurately classifying known attacks to also detecting novel attacks that were initially flagged as anomalies and then routed to a classifier for identification, even though they are zero-day attacks that lie outside the classifier’s learned knowledge.

Within this context, the problem can be formulated as follows: **Given** a stream of network traffic instances, some of which may be benign, known attacks, or previously unseen attacks, **detect and classify** suspicious traffic and **determine** whether an instance initially identified as anomalous and then classified as a known attack may actually correspond to a zero-day attack. The final **objective** is to maximize the accurate detection of zero-day attacks at the attack-identification stage while balancing false positives and false negatives to preserve reliable overall classification performance.

A. Overview of our approach

Our methodology introduces a mechanism at the attack-identification stage in which each prediction made by the multiclass classifier is further analyzed using (i) per-instance feature importance scores, computed using SHAP, and (ii) predictive uncertainty, quantified through ensemble entropy. The per-instance feature importance vectors are first reconstructed using an AE trained exclusively on known attack classes, where the resulting RE reflects how much a sample’s attribution pattern deviates from those typically exhibited by known attacks. These reconstruction errors are then combined with the corresponding epistemic uncertainty estimates to form a compact representation, which is subsequently modeled using a GMM to distinguish known attack behavior from potential zero-day attacks. Figure 2 shows a schematic representation of our proposed approach¹. The following subsections describe each stage in detail.

B. Stage 1: Unsupervised Anomaly Detection

The first stage performs unsupervised anomaly detection using any anomaly detection approach, such as an autoencoder. This is trained exclusively on benign traffic, where the model learns normal network behavior by minimizing the reconstruction Mean Squared Error (MSE). At deployment, every incoming flow passes through this AE. The flows whose reconstruction error exceeds a fixed threshold (e.g., 95th percentile of validation-set errors) are flagged as anomalies, i.e., potential attacks, whereas other flows are considered benign and discarded from further experimentation. Our experimental evaluation assumes that the first-stage anomaly detector has already separated malicious from benign traffic and therefore evaluates the attack-identification and zero-day detection stages independently. In an end-to-end deployment, errors from the upstream anomaly detector may propagate to these downstream stages and affect the overall system performance, and evaluating this interaction is left for future work.

C. Stage 2: Supervised Classification of Known Attacks

In the second stage, all attacks are passed to a supervised multiclass classifier, trained on a labeled dataset of flows $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$ with $\mathbf{y}_i \in \{1, \dots, K\}$. The classifier g (e.g., XGB or RF) assigns each anomaly to one of the known attack classes, obtaining the predicted class $\hat{\mathbf{y}} = g(\mathbf{x})$ under a standard multiclass log-loss optimization setup.

Per-Instance Feature Influence via SHAP: To compute the SHAP values of each feature, we use SHAP *TreeExplainer* on the base model and training set. Since this is a multiclass problem, SHAP computes *one importance vector per feature per class*, i.e., $\Phi = [\phi_{i,j,k}] \in \mathbb{R}^{N \times d \times K}$ where $\phi_{i,j,k}$ is the contribution of feature j to class k on sample i . For each anomaly $\mathbf{x}_i$, let $\hat{k} = \hat{y}_i$. Here, we simply index into the vector corresponding to the *predicted-class SHAP vector*: $\phi^{(i)} = [\phi_{i,1,\hat{k}}, \phi_{i,2,\hat{k}}, \dots, \phi_{i,d,\hat{k}}] \in \mathbb{R}^d$. Thus, for each data point, we obtain one real value per input feature of SHAP. To reduce dimensionality and focus on the most influential

¹Code is available at <https://github.com/hussein-fawaz/ZDA-SHAP-UQ>

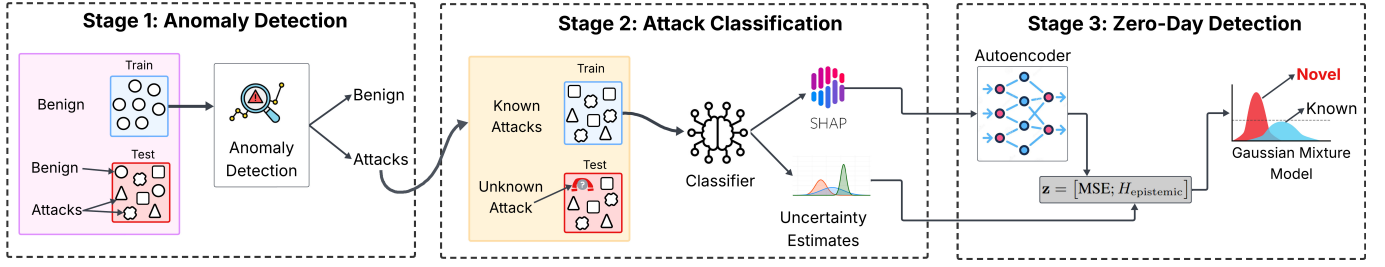


Fig. 2: Schematic representation of our proposed pipeline for zero-day attack detection. First, anomaly detection separates benign traffic from potential attacks. Then, attack identification is carried out using a classifier, where per-instance feature importance (SHAP values) and epistemic uncertainty are computed, with the resulting SHAP reconstruction error and epistemic uncertainty combined into a 2D representation and jointly modeled using a GMM. Instances whose anomaly score exceeds the threshold are labeled as zero-day attacks.

features, we select the subset of features that collectively account for 90% of the total absolute SHAP value globally across the training distribution [52].

Prediction Uncertainty Quantification: To quantify the uncertainty of the classifier’s predictions, we employ a sub-ensemble entropy method (Section II-B). Let the base classifier produce logits $\ell \in \mathbb{R}^K$ for K classes. We first split its T trees (in XGB or RF) into M disjoint sub-ensembles of roughly equal size. For each sub-ensemble $m = 1, \dots, M$, we compute the logit margins $\ell^{(m)}$ and convert them to softmax probabilities $s^{(m)} = \text{softmax}(\ell^{(m)}) \in \mathbb{R}^K$. We then form the mean prediction $\bar{s} = \frac{1}{M} \sum_{m=1}^M s^{(m)}$ and compute per sample:

$$H_{\text{total}} = - \sum_{k=1}^K \bar{s}_k \ln(\bar{s}_k) \quad (2)$$

$$H_{\text{aleatoric}} = \frac{1}{M} \sum_{m=1}^M \left(- \sum_{k=1}^K s_k^{(m)} \ln(s_k^{(m)}) \right) \quad (3)$$

$$H_{\text{epistemic}} = H_{\text{total}} - H_{\text{aleatoric}} \quad (4)$$

Since the epistemic uncertainty is expected to be high for unseen inputs, we use $H_{\text{epistemic}}$ as a single scalar uncertainty score for the zero-day attack detection.

D. Stage 3: Zero-Day Detection via Joint Modeling

The third stage leverages a GMM to jointly model the reconstruction fidelity of SHAP values and the prediction uncertainty. Our framework is model-agnostic with respect to any classifier admitting tractable per-instance feature attribution and proper uncertainty estimation.

Reconstruction of SHAP-Based Feature Attributions: The AE is trained solely on flows belonging to known attack classes. Following standard design conventions for reconstruction-based anomaly detection on tabular data, it uses a symmetric encoder–decoder architecture with dense layers of decreasing dimensionality ($256 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 16$) and a mirrored decoder ($16 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 256$). Each hidden layer uses batch normalization, ReLU activation, and 10% dropout, with the dropout rate selected by minimizing the validation loss. A tanh activation in the output layer constrains reconstructions to $[-1, 1]$. Training uses the Adam optimizer, MSE loss, a batch size of 512, and an 80/20 training/validation split. Early stopping based on the validation loss selects the

training duration, up to a maximum of 10 epochs, to limit overfitting.

For each instance belonging to the known attack classes, we first compute two quantities: (i) the AE RE computed using SHAP per-instance explanation vectors as input, and (ii) the epistemic uncertainty score $H_{\text{epistemic}}$. These values form a two-dimensional vector for each sample,

$$\mathbf{z} = [\text{MSE}; H_{\text{epistemic}}]. \quad (5)$$

GMM Joint Modeling: We then fit a GMM on the set of $\{\mathbf{z}_i\}_{i=1}^N$ vectors extracted from known attacks, modeling their joint distribution in this reduced space. Formally, the GMM assumes that each $\mathbf{z}$ is generated from a mixture of C Gaussian components, each with its own mean μ_c and covariance Σ_c .

To determine the appropriate number of mixture components and covariance structure, the model automatically evaluates multiple configurations on the validation set and selects the one with the lowest Bayesian Information Criterion (BIC) [53], ensuring a well-calibrated density estimate of known-attack behavior. Model parameters π_c, μ_c, Σ_c are estimated via the Expectation–Maximization (EM) algorithm [54], which alternates between computing the component responsibilities and maximizing the expected complete-data log-likelihood using these soft assignments. This results in a smooth, multi-modal density that captures the distribution of “normal” (known-attack) reconstruction–uncertainty pairs.

During inference, a test flow is processed through the SHAP-only AE to obtain its reconstruction error, paired with its epistemic uncertainty to form the quantity $\mathbf{z}$, defined in Equation (5). The fitted GMM then evaluates its log-probability density:

$$\log p(\mathbf{z}) = \log \left(\sum_{c=1}^C \pi_c \mathcal{N}(\mathbf{z} | \mu_c, \Sigma_c) \right), \quad (6)$$

We define the anomaly score as $s(\mathbf{z}) = -\log p(\mathbf{z})$, so that higher values indicate greater deviation from known-attack behavior. A sample is flagged as a zero-day attack if $s(\mathbf{z})$ exceeds the τ -th percentile of the validation-set anomaly score distribution, i.e., if it falls in a low density region of the joint RE–uncertainty space.

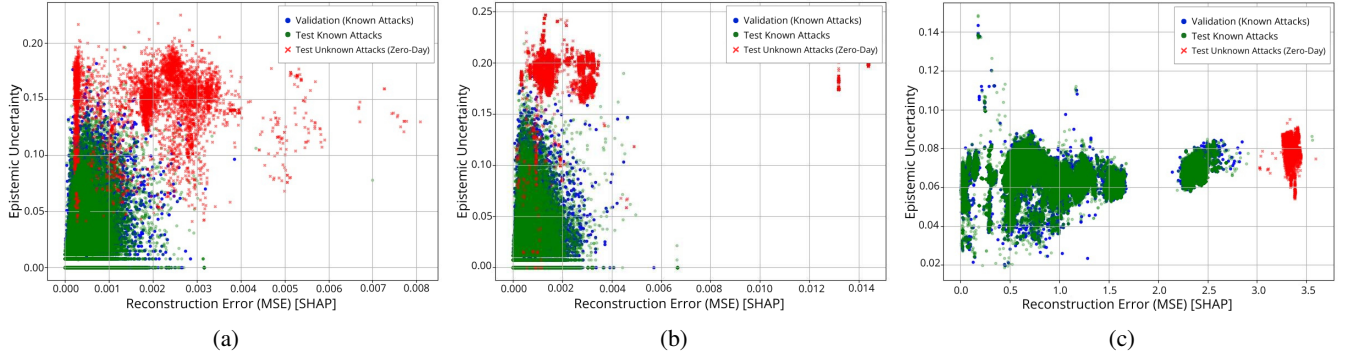


Fig. 3: 2D representations used as inputs to the GMM, where each sample is mapped to a point defined by its SHAP-based RE (x-axis) and epistemic uncertainty (y-axis). Plots (a), (b), and (c) correspond to the Backdoor (NF-TON-IoT), DoS (CICIoT-2023), and TFTP (CIC-DDoS2019) scenarios, respectively. Zero-day attacks are shown in red, whereas known-class samples from the validation and test sets are shown in blue and green, respectively.

E. Illustrative Example

Figure 3 shows the two-dimensional representations used as inputs to the GMM, where each point corresponds to a single instance embedded in a space defined by its SHAP-based RE and epistemic uncertainty. In the figures, the horizontal axis reflects the RE of the AE trained on SHAP feature-attribution vectors, while the vertical axis measures the epistemic uncertainty derived from the ensemble classifier. Figure 3(a) illustrates how combining SHAP-based RE with epistemic uncertainty provides a highly discriminative representation for separating known attacks from zero-day samples, as zero-day attacks (represented in red) occupy regions characterized by both elevated RE and high epistemic uncertainty. This observation is consistent with our hypothesis that unseen attacks diverge from the learned SHAP attribution patterns of known classes while also triggering greater model disagreement, captured through higher epistemic entropy. In contrast, known-class samples from both validation and test sets, represented in blue and green, respectively, cluster in regions of low RE and low uncertainty, reflecting both stable attribution profiles and classifier confidence.

However, this behavior does not hold uniformly across all scenarios. In some cases, SHAP RE alone may not fully separate unknown attacks from known ones, as certain unseen instances may exhibit attribution patterns that partially overlap with those of known classes. This behavior is illustrated in Figure 3(b), where zero-day samples are primarily distinguished by elevated epistemic uncertainty, while their SHAP RE overlaps with that of known-class instances. Conversely, epistemic uncertainty may occasionally be insufficient when the classifier becomes overconfident on anomalous inputs. Figure 3(c) shows a complementary scenario in which zero-day samples exhibit high SHAP RE and are clearly separated along the horizontal axis, whereas epistemic uncertainty alone provides limited discriminative power. When considered jointly, the two measures provide robustness: *whenever one signal is weak or ambiguous, the other compensates*. This complementary behavior results in consistent separation of zero-day points in the joint space, forming distinct red clusters clearly detached from the blue/green known-class manifold.

V. EVALUATION SETTINGS

In this section, we describe the different baselines considered in our work, along with the datasets and the experiments.

A. Considered Baselines

We compare our proposed approach against the baselines described below, which also constitute an ablation of the proposed pipeline by isolating the contributions of SHAP-based reconstruction alone, epistemic uncertainty alone, and a simple AE-based integration of both signals.

- *Training*: Represents the standard baseline in which an AE is trained on raw feature vectors of known attacks, where zero-day detection is performed by thresholding the RE of the raw input, reflecting standard AE-based anomaly detection practice.
- *EFC* [10]: Another baseline which is based on energy flow. Low energy indicates that the flow matches the learned distribution, whereas energy values above a threshold indicate unknown flows.
- *SHAP* [51]: The AE is trained exclusively on SHAP per-instance explanation vectors, allowing us to assess whether attribution patterns alone can reliably distinguish known from unseen attacks.
- *Epistemic* [15]: Zero-day detection is performed by thresholding epistemic uncertainty directly. This baseline evaluates the effectiveness of uncertainty as a standalone indicator of out-of-distribution behavior.
- *AE-based SHAP + UQ*: This is a simple alternative to the GMM, where we consider SHAP per-instance explanation vectors concatenated with the corresponding epistemic uncertainty score and used directly as inputs to an AE. For each anomaly, the SHAP vector $\phi^{(i)} \in \mathbb{R}^d$ is augmented with its epistemic value $H_{\text{epistemic},i}$, forming a $(d+1)$ -dimensional representation to serve as an input. This baseline assesses whether a single AE operating on a combined SHAP-uncertainty representation is sufficient, and provides a direct comparison point to our GMM, which models these signals more explicitly.

TABLE II: Average macro performance of known attacks across the three datasets: XGB vs. RF (mean $\pm$ std)

Dataset	Model	Accuracy (%)	Recall (%)	Precision (%)	F1 (%)
NF-ToN-IoT	XGB	99.99 (± 0.00002)	99.93 (± 0.0004)	99.85 (± 0.001)	99.89 (± 0.0004)
	RF	100.00 (± 0.00001)	99.94 (± 0.0002)	99.94 (± 0.0004)	99.94 (± 0.0002)
CIC-IoT2023	XGB	99.7 (± 0.002)	75.7 (± 0.050)	90.9 (± 0.017)	78.8 (± 0.043)
	RF	99.8 (± 0.002)	86.0 (± 0.031)	97.6 (± 0.007)	90.1 (± 0.021)
CIC-DDoS2019	XGB	97.2 (± 0.007)	70.9 (± 0.029)	73.2 (± 0.029)	71.7 (± 0.029)
	RF	96.5 (± 0.009)	66.6 (± 0.032)	67.5 (± 0.032)	67.0 (± 0.032)

B. Dataset Description

We evaluate our approach on multiple publicly available datasets.

NF-TON-IoT [55]: The *NF-TON-IoT* dataset is derived from the original TON-IoT telemetry collection and provides flow-based network records extracted from IoT, fog, and cloud layers. It contains approximately 16.9 million NetFlow entries spanning 9 attack types (Backdoor, Distributed Denial of Service (DDoS), Denial of Service (DoS), Injection, MITM, Password, Ransomware, Scanning, XSS) in addition to benign traffic. A total of 44 features are provided per flow.

CIC-IoT2023 [56]: The *CICIoT2023* dataset consists of around 46 million flows, including roughly 1 million benign samples and malicious traffic representing 33 attacks grouped into 7 categories (DDoS, DoS, Reconnaissance, Web, Brute Force, Spoofing, Mirai). Each flow is described by 47 features.

CIC-DDoS2019 [57]: The *CIC-DDoS2019* dataset contains over 11 million flows combining benign traffic with numerous DDoS variants (e.g., DNS, LDAP, MSSQL, Net-BIOS, NTP, Portmap, SNMP, SSDP, SYN, UDP, UDP-Lag, WebDDoS). Each flow is characterized by 88 features and labeled as benign or one of the attack types.

C. Experiments and Evaluation Metrics

For each dataset, we applied standard preprocessing, where rows with missing or infinite values were dropped, exact duplicates removed, and zero-variance features excluded. Highly skewed features were winsorized to limit the impact of outliers, and highly correlated features were removed. All remaining features were then standardized before training.

To simulate a zero-day attack scenario, we employ a leave-one-attack-out approach, where for each dataset, one attack class is excluded during training and used exclusively for testing. This setup mimics real-world conditions in which the system encounters previously unseen attacks. The procedure is repeated for every attack class available in the dataset, with each iteration training the model on all remaining attack types and testing it on the excluded one alongside unseen benign and known-class samples. Results are averaged over five runs with different random seeds to account for variability in the training process.

For the supervised multiclass classification component, we measure performance using accuracy, precision, recall, and F1-score. To evaluate the effectiveness of detecting zero-day attacks, we rely primarily on the Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve and the F1-Score. The AUROC is emphasized because it

is a threshold-independent metric that captures the trade-off between the True Positive Rate (Recall) and the False Positive Rate (FPR) across all possible decision thresholds. Nevertheless, we apply a fixed decision rule based on a percentile threshold: flows whose associated score (e.g. reconstruction error, epistemic uncertainty, negative log density for GMM) exceeds the 95th percentile of the validation error distribution are flagged as anomalies. This thresholding strategy reflects a practical assumption that only a small fraction of known-class validation samples should fall outside the learned distribution of normal behavior.

VI. RESULTS AND DISCUSSION

We now discuss the experimental results. The main take-aways of our study can be summarized as follows:

- 1) Independently of the underlying ML model, our proposed *GMM (SHAP RE + UQ)* approach delivers consistent and reliable zero-day detection performance across all datasets. While epistemic uncertainty alone performs well with RF, its performance degrades significantly when coupled with XGB, whereas our proposed approach maintains stable behavior across both models.
- 2) Under *best-case* conditions, i.e., considering the attack scenarios for which each approach achieves its highest relative improvement over the baseline, our approach consistently achieves the highest or second-highest performance across datasets, with substantial gains in both F1-score and AUROC, reaching up to 73% and 38% respectively, relative to the *Training* baseline.
- 3) In terms of *average* performance, our approach is either the best or the second-best across all configurations for both F1-score and AUROC, demonstrating strong overall effectiveness beyond isolated favorable cases.
- 4) Most importantly, the proposed approach exhibits superior robustness in *worst-case* scenarios. While alternative methods based solely on SHAP or uncertainty can suffer significant performance degradation, our approach shows substantially smaller worst-case degradation than the alternative methods and, in several cases, performance comparable to or better than the baseline.

A. Multiclass Classification of Known Attacks

We begin by evaluating the predictive performance of the multiclass classifier used to identify known attacks. Although the primary aim of our study is not the detection of known attacks but rather the identification of previously unseen (zero-day) attacks, it remains essential that the classifier reliably

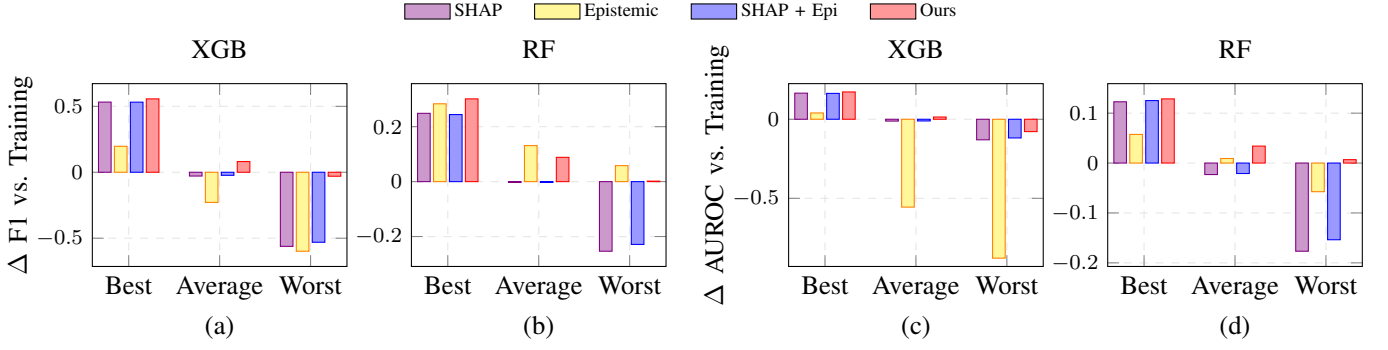


Fig. 4: Relative improvement in F1-score and AUROC over the *Training* baseline for the NF-TON-IoT dataset.

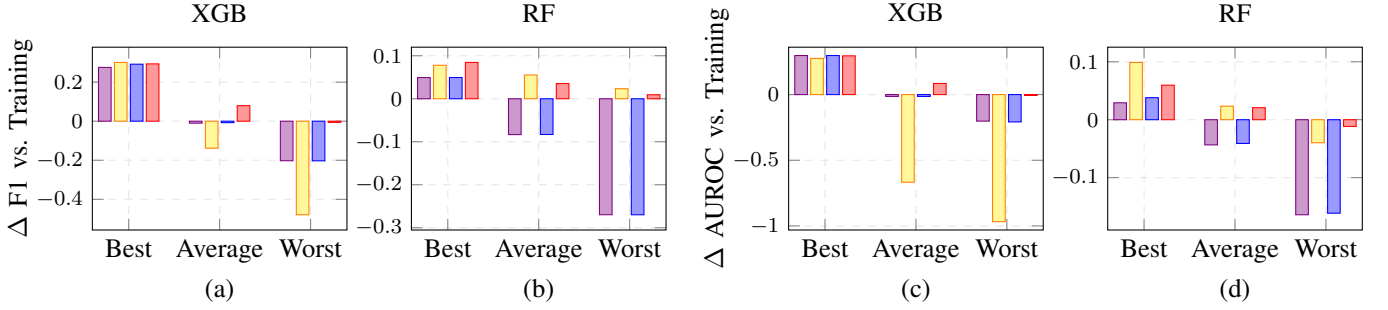


Fig. 5: Relative improvement in F1-score and AUROC over the *Training* baseline for the CICIoT-23 dataset.

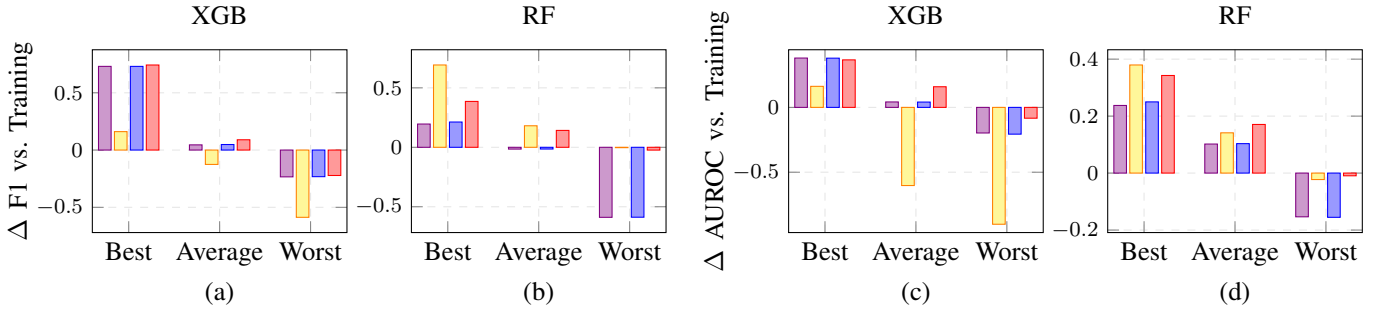


Fig. 6: Relative improvement in F1-score and AUROC over the *Training* baseline for the CIC-DDoS2019 dataset.

recognizes familiar attack types once encountered. This ensures that zero-day detection performance is not confounded by misclassifications at the supervised stage. Recall that, for each dataset, one attack class is held out as an unseen zero-day attack, while the remaining classes are used for training and evaluation. Table II summarizes the mean macro-average accuracy, recall, precision, and F1-scores of XGB and RF, together with their standard deviations computed across all leave-one-attack-out experiments.

We first consider the NF-TON-IoT dataset, where both models achieve near-perfect performance across all metrics. XGB attains an average accuracy of 99.99% (± 0.00002) and an F1-score of 99.89% (± 0.0004), while RF performs similarly, reaching 100.00% (± 0.00001) accuracy and an F1-score of 99.94% (± 0.0002). The consistently low variance across folds confirms the stability of both models in this setting. Because the classification performance on known attacks is already near optimal, this dataset provides a clear environment for assessing the incremental value of the proposed zero-

day detection stage without confounding classifier limitations. In the CIC-IoT2023 dataset, accuracy remains high for both models (99.7% for XGB and 99.8% for RF), yet recall and F1 reveal pronounced differences. XGB achieves a recall of 75.7% (± 0.050) and an F1-score of 78.8% (± 0.043), whereas RF performs better with a recall of 86.0% (± 0.031) and an F1-score of 90.1% (± 0.021). These discrepancies reflect the greater diversity and imbalance among attack types, with misclassifications driven primarily by recall rather than false alarms, as evidenced by the strong precision of both models.

A similar trend is observed in the CIC-DDoS2019 dataset, where XGB attains an accuracy of 97.2% and an F1-score of 71.7%, while RF achieves a comparable accuracy of 96.5% but a slightly lower F1-score of 67.0%. The reduced recall values for both models reflect overlapping traffic patterns among heterogeneous DDoS subclasses, making this dataset a challenging testbed for zero-day detection.

B. Zero-Day Attack Detection: Overview

We now focus on zero-day attack detection by presenting a panoramic comparison of all approaches against the state-of-the-art baseline (i.e., *Training* and *EFC*). For each configuration, defined by an underlying model and a dataset, and evaluated through the corresponding leave-one-attack-out experiments in which each attack is separately treated as a zero-day, we report the *best*, *average*, and *worst* values of F1-score and AUROC across all the attacks, achieved by each approach and normalized with respect to the *Training* baseline. For instance, Figure 4(a) illustrates this procedure: after iterating over all attacks in the NF-TON-IoT dataset (nine in this case), the best, average, and worst relative performance of each approach are computed and expressed with respect to the baseline. This panoramic view quantifies not only how much an approach can improve over the baseline, but also the extent to which it may degrade performance in unfavorable zero-day scenarios.

Figures 4, 5, and 6 report these results for NF-TON-IoT, CIC-IoT2023, and CIC-DDoS2019, respectively, shown separately for the two underlying classifiers (XGB and RF). We clarify that the y-axis denotes the relative improvement (absolute difference in metric value) with respect to the *Training* baseline. Note that all results are computed using a 0.95 percentile threshold on the detection score (i.e., the threshold is set such that the false positive rate is approximately 0.05), following a protocol similar to [51].

Results show that, in terms of F1-score, across all datasets and independently of the underlying ML model, our proposed approach achieves the highest or second-highest performance under *best-case* conditions, with improvements reaching up to 73% in some cases (e.g., for CIC-DDoS2019 under both XGB and RF) and up to 56% in NF-TON-IoT under XGB. In terms of *average* F1-score performance, our approach achieves either the highest or second-highest improvement across configurations, typically ranging between 0.03 and 0.17 relative to the *Training* baseline. Under *worst-case* conditions, it also achieves either the highest or second-highest improvement. Specifically, in NF-TON-IoT, our approach exhibits a worst-case degradation of only about -0.03, substantially outperforming the other approaches under XGB, which degrade by approximately 0.15 to 0.6 with respect to training. In CIC-IoT2023, the proposed approach shows worst-case performance essentially equal to the *Training* baseline, indicating that, even in the least favorable scenario, it does not negatively affect zero-day detection. Finally, in CIC-DDoS2019, the proposed approach shows a degradation of around -0.22 F1 when using XGB, while under RF it achieves performance comparable to that of the baseline.

We now consider AUROC to compare performance under a threshold-independent setting. Across all datasets and independently of the underlying model, our approach achieves either the best or second-best performance under *best-case* conditions, with improvements as high as 40% (e.g., CIC-DDoS2019). In terms of *average* AUROC, it achieves the best performance across all configurations except one, where it ranks second. More importantly, in terms of *worst-case* AU-

ROC, our approach consistently achieves the best performance among the compared methods. In addition to outperforming the other approaches, its worst-case performance remains close to that of the *Training* baseline, either marginally improving or marginally underperforming, with a maximum degradation of about 8% under XGB and near-neutral behavior under RF. Overall, these results indicate that the proposed method delivers more reliable and consistent performance than the alternatives, providing substantial gains in favorable scenarios, strong average improvements in most settings, and worst-case behavior that remains in line with the baseline. This reliability is particularly valuable for intrusion detection, where methods must behave consistently across different zero-day scenarios. See Appendix A (Tables VII, VIII, and IX), reporting per-attack AUROC values for NF-TON-IoT, CIC-IoT-23, and CIC-DDoS2019, respectively.

C. Per-Attack Analysis

We now inspect the performance of the various approaches at the granularity of individual attacks within each dataset. Tables III, IV, and V report the per-attack average results over five runs, together with the corresponding standard deviations to highlight the variability across attacks and training procedures, for the three datasets, respectively. Bold values refer to the best performance, while underlined ones to the second best. For NF-TON-IoT under XGB, our approach achieves the best performance in seven cases out of nine, lagging only marginally behind the top-performing approach in the remaining two cases. As seen earlier, our approach yields the highest average F1-score, improving over the training baseline by approximately +0.08 (from 0.725 to 0.806), and substantially outperforming the EFC baseline (0.596). For instance, in the DDoS case, our approach attains an F1-score of 0.977, compared to 0.421 for the training baseline, 0.409 for EFC, and 0.218 for the epistemic uncertainty variant. Similarly pronounced gains are observed for Password (from 0.933 to 0.971) and XSS (from 0.875 to 0.937). Similarly, in terms of AUROC, our approach attains the highest performance, with an average AUROC of 0.970 compared to 0.955 for the training baseline and 0.861 for EFC, while significantly outperforming the other approaches. It is also important to note that it achieves relatively low standard deviations for both F1-score (0.168) and AUROC (0.038), which further supports the consistency and reliability of the method.

We now examine the results achieved with RF as the underlying ML model. Epistemic uncertainty achieves the best overall F1-score in 8 out of 9 cases, while our proposed approach achieves the best performance once and the second-best performance in the remaining cases, typically lagging only marginally behind epistemic uncertainty (often by less than 0.05-0.10 F1). For example, in the Password and XSS attacks, the F1-score difference between epistemic uncertainty and the proposed approach remains below 0.03, indicating near-equivalent performance. As highlighted previously, this yields the second-best average F1-score for our approach under RF (0.813 versus 0.856 for epistemic uncertainty), while still substantially outperforming both the training baseline (0.722)

TABLE III: NF-TON-IoT: F1-score (Macro)

Attack	XGB						RF			
	Training	EFC	SHAP	Epistemic	SHAP+Epi	Ours	SHAP	Epistemic	SHAP+Epi	Ours
Backdoor	0.829	0.783	0.791	0.491	0.792	<u>0.799</u>	0.618	0.932	0.600	<u>0.839</u>
DDoS	0.421	0.409	<u>0.953</u>	0.218	0.952	0.977	0.669	<u>0.704</u>	0.665	0.822
DoS	0.777	0.859	0.743	0.572	0.746	<u>0.792</u>	0.748	0.906	0.747	0.845
Injection	0.874	0.594	<u>0.906</u>	0.475	0.906	0.924	0.886	0.982	0.886	<u>0.934</u>
MITM	0.539	0.528	0.529	0.542	0.529	<u>0.541</u>	0.499	0.628	0.498	<u>0.543</u>
Password	<u>0.933</u>	0.394	0.882	0.334	0.895	0.971	0.680	0.991	0.709	<u>0.976</u>
Ransomware	<u>0.522</u>	0.497	0.524	0.491	0.524	0.524	0.511	0.587	0.512	<u>0.524</u>
Scanning	0.755	0.916	0.625	0.952	0.629	0.850	0.919	0.985	0.917	<u>0.957</u>
XSS	<u>0.875</u>	0.382	0.313	0.393	0.344	0.937	0.970	0.989	0.969	<u>0.982</u>
Avg. F1 ↑	0.725	0.596	0.696	0.496	0.702	0.806	0.722	0.856	0.722	<u>0.823</u>
Std. F1 (across attacks) ↓	<u>0.184</u>	0.207	0.214	0.203	0.208	0.168	0.172	0.167	0.173	0.179
Std. F1 (across runs) ↓	0.043	0.005	0.054	<u>0.023</u>	0.057	0.026	0.018	<u>0.003</u>	0.020	0.001
Avg. AUROC ↑	0.955	0.861	0.944	0.400	0.945	0.970	0.932	<u>0.964</u>	0.934	0.989
Std. AUROC (across attacks) ↓	<u>0.052</u>	0.100	0.054	0.355	<u>0.051</u>	0.038	0.052	<u>0.071</u>	<u>0.047</u>	0.018
Std. AUROC (across runs) ↓	0.018	0.008	0.025	0.063	0.024	0.023	0.011	0.002	0.015	<u>0.004</u>

TABLE IV: CICIoT-23: F1-score (Macro)

Attack	XGB						RF			
	Training	EFC	SHAP	Epistemic	SHAP+Epi	Ours	SHAP	Epistemic	SHAP+Epi	Ours
BruteForce	<u>0.505</u>	0.501	0.502	0.487	0.502	0.514	0.498	0.528	0.498	<u>0.514</u>
Web	<u>0.524</u>	0.509	0.506	0.487	0.506	0.536	0.508	0.559	0.508	<u>0.539</u>
Spoofing	<u>0.820</u>	0.717	0.617	0.477	0.617	0.822	0.550	0.887	0.550	<u>0.830</u>
Recon	0.775	0.686	0.650	0.479	0.651	<u>0.769</u>	0.582	0.853	0.583	<u>0.792</u>
Mirai	<u>0.900</u>	0.576	0.879	0.421	0.878	0.906	0.950	0.960	0.950	<u>0.951</u>
DoS	0.409	0.391	0.433	0.610	0.437	0.649	0.359	<u>0.462</u>	0.357	0.493
DDoS	0.429	0.284	<u>0.704</u>	0.333	0.720	0.722	0.331	0.499	0.334	<u>0.491</u>
Avg. F1 ↑	0.623	0.524	0.613	0.485	0.616	0.703	0.540	0.678	0.540	<u>0.659</u>
Std. F1 (across attacks) ↓	0.203	<u>0.154</u>	0.151	0.114	0.152	0.145	0.204	0.212	0.204	<u>0.193</u>
Std. F1 (across runs) ↓	<u>0.012</u>	0.000	0.048	0.043	0.051	0.029	0.016	0.003	0.017	0.003
Avg. AUROC ↑	<u>0.879</u>	0.847	0.866	0.212	0.865	0.964	0.836	0.902	0.838	<u>0.900</u>
Std. AUROC (across attacks) ↓	0.163	0.147	<u>0.082</u>	0.344	0.084	0.043	0.131	0.169	<u>0.125</u>	0.165
Std. AUROC (across runs) ↓	<u>0.011</u>	0.000	0.065	0.031	0.065	0.018	0.028	<u>0.002</u>	0.027	0.007

TABLE V: CIC-DDoS2019: F1-score (Macro)

Attack	XGB						RF			
	Training	EFC	SHAP	Epistemic	SHAP+Epi	Ours	SHAP	Epistemic	SHAP+Epi	Ours
DNS	0.539	<u>0.641</u>	0.579	0.476	0.581	0.671	0.599	0.715	0.599	0.703
LDAP	<u>0.566</u>	0.527	0.503	0.477	0.503	0.567	0.575	0.688	0.575	<u>0.668</u>
MSSQL	0.692	<u>0.707</u>	0.653	0.443	0.651	0.806	0.575	0.904	0.576	<u>0.889</u>
NTP	0.599	0.695	0.605	0.221	0.602	<u>0.643</u>	0.795	0.956	0.812	<u>0.955</u>
NetBIOS	0.508	0.569	0.577	0.487	<u>0.587</u>	0.611	0.535	0.616	0.536	<u>0.594</u>
Portmap	0.521	0.561	0.559	0.488	<u>0.563</u>	0.589	0.528	0.603	0.528	<u>0.591</u>
SNMP	<u>0.566</u>	0.521	0.499	0.477	0.499	0.645	0.555	0.735	0.555	<u>0.677</u>
Syn	0.957	0.499	0.723	0.369	0.724	<u>0.735</u>	0.368	0.964	0.369	0.933
TFTP	0.246	0.293	<u>0.977</u>	0.292	<u>0.977</u>	0.979	0.397	0.939	0.375	<u>0.632</u>
UDP	0.404	0.447	0.483	0.565	0.483	<u>0.508</u>	0.424	0.499	0.423	<u>0.472</u>
UDPLag	0.648	<u>0.673</u>	0.632	0.457	0.655	0.792	0.712	<u>0.805</u>	0.723	0.835
WebDDoS	0.498	0.501	0.499	0.487	<u>0.500</u>	0.501	<u>0.500</u>	0.495	<u>0.500</u>	0.501
Avg. F1 ↑	0.562	0.553	<u>0.607</u>	0.437	0.610	0.671	0.547	0.743	0.548	0.704
Std. F1 (across attacks) ↓	0.169	0.118	0.096	<u>0.104</u>	0.137	0.137	<u>0.123</u>	0.171	0.130	0.164
Std. F1 (across runs) ↓	0.030	0.004	0.047	<u>0.026</u>	0.048	0.026	0.031	<u>0.009</u>	0.031	<u>0.009</u>
Avg. AUROC ↑	0.770	<u>0.822</u>	0.812	0.167	0.811	0.929	0.871	<u>0.911</u>	0.873	0.940
Std. AUROC (across attacks) ↓	<u>0.128</u>	0.208	0.135	0.208	0.139	0.046	<u>0.057</u>	<u>0.116</u>	<u>0.057</u>	0.052
Std. AUROC (across runs) ↓	0.076	0.003	0.049	0.049	0.048	<u>0.020</u>	0.028	<u>0.005</u>	0.028	0.011

and EFC (0.596). In contrast, and importantly, in terms of AUROC, our approach achieves the best performance overall under RF, with an average AUROC of 0.989, compared to 0.932 for the training baseline, 0.861 for EFC, and 0.964 for epistemic uncertainty. This difference highlights that epistemic uncertainty can be more sensitive to the specific underlying classifier, performing strongly with RF in terms of F1-score in this dataset, while the proposed joint modeling approach remains consistently competitive and provides stronger threshold-independent separation as reflected by AUROC.

Similar takeaways can be extracted from the CIC-IoT2023 and CIC-DDoS2019 datasets. Across both datasets, epistemic uncertainty exhibits a strong dependence on the underlying classifier. In particular, when RF is used, epistemic uncertainty often achieves competitive or even superior F1-score performance in 6 out of 7 cases in CIC-IoT2023 and 9 out of 12 cases in CIC-DDoS2019, as also reflected in the aggregate results. However, when XGB is employed, epistemic uncertainty becomes markedly less reliable, frequently yielding negative average shifts and substantial worst-case degradations, especially in CIC-IoT2023 and CIC-DDoS2019, where average F1-score drops not only below the training baseline but also below EFC in several cases, and worst-case degradations approach those of SHAP-only methods. This behavior highlights the inherent model sensitivity of epistemic uncertainty, whose effectiveness is tightly coupled to the ensemble structure and diversity of the underlying classifier.

In contrast, our approach exhibits a substantially more consistent behavior across both classifiers and all datasets. For example, in CIC-IoT2023, our approach maintains positive average improvements over both the training and EFC baselines (approximately +0.08 under XGB and +0.12 relative to EFC under XGB, as well as +0.12 under RF relative to training and +0.14 relative to EFC) and near-neutral worst-case performance under both XGB and RF, whereas epistemic uncertainty degrades sharply under XGB. A similar pattern is observed in CIC-DDoS2019, where epistemic uncertainty performs strongly with RF but fails to generalize under XGB, while our approach continues to deliver stable average gains and controlled worst-case behavior. These results indicate that jointly modeling SHAP-based RE and epistemic uncertainty mitigates the classifier-dependent limitations of uncertainty alone, leading to a more robust and model-agnostic zero-day detection mechanism.

We further note that while epistemic uncertainty combined with RF occasionally achieves marginally higher F1-scores for specific attack categories, our approach consistently achieves equal or superior AUROC across datasets. For instance, in CIC-DDoS2019 under RF, our proposal achieves an average AUROC of approximately 0.94 compared to about 0.91 for epistemic uncertainty, 0.77 for the training baseline, and 0.82 for EFC, and similarly dominates worst-case AUROC across all datasets. This suggests that the joint modeling strategy provides better threshold-independent separability between known and zero-day attacks, reinforcing its suitability for operational intrusion detection systems where robustness across models and scenarios is critical.

We additionally examine cases where our approach shows

the smallest or negative relative improvement. These fall into two categories: attacks where the *Training* baseline is already at ceiling ($\text{AUROC} \geq 0.95$) or at the random-guess floor ($\text{F1} \approx 0.50$) for both classifiers, leaving no added value (e.g., *WebDDoS*, *ransomware*, *mitm*), most notably *Syn* on CIC-DDoS2019, where XGB's already-strong *Training* performance is degraded by fusing in a comparatively unreliable epistemic-uncertainty signal, a pattern not observed for RF. This suggests residual limitations stem from per-classifier uncertainty calibration rather than the joint reconstruction-uncertainty mechanism itself.

We performed paired *t*-tests, pairing our approach against each baseline across attack-run replicates (five runs $\times$ attacks, pooled across the three datasets), with Holm-Bonferroni correction across the five pairwise comparisons and significance assessed at $\alpha = 0.05$. The results show that our approach performs significantly better than *Training* ($p < 0.0001$), *EFC* ($p < 0.0001$), *SHAP* ($p < 0.0001$), and *SHAP+Epi* ($p < 0.0001$) across both classifiers, and performs significantly better than *Epistemic* under XGB ($p < 0.0001$). The only exception is F1-Macro under RF, for which *Epistemic* significantly outperforms our approach ($p < 0.0001$). In contrast, under XGB, our approach significantly outperforms *Epistemic* in terms of both F1-Macro and AUROC ($p < 0.0001$).

D. Sensitivity Analysis and ROC Evaluation

We analyze the sensitivity of the proposed approach to the choice of the GMM threshold used to flag zero-day attacks at inference time. A threshold of the 95th percentile means that the 5% of validation samples with the highest anomaly scores define the rejection boundary, and any test sample falling beyond this boundary is flagged as a zero-day attack.

The three datasets exhibit subtly different sensitivity profiles, reflecting their underlying complexity. The standard deviation, computed over five independent runs, remains below 0.01 for RF on every dataset, confirming that the method is stable and robust with respect to the training procedure and data-split variation. On NF-ToN-IoT (Figure 7a), both classifiers maintain consistently high F1-scores across the evaluated range, with RF achieving its highest F1-Macro of 0.868 at the 99th percentile and retaining 0.813 at the 95th percentile, indicating that even a moderately strict threshold suffices for this relatively well-separated dataset. CIC-DDoS 2019 (Figure 7c) exhibits a more pronounced ramp: F1-Macro climbs from 0.587 at the 99th percentile to a peak of 0.716 ± 0.004 at the 90th percentile for RF and 0.687 ± 0.008 at the 91st percentile for XGB, reflecting the greater diversity of attack signatures in this corpus. CICIoT-2023 (Figure 7b) peaks earlier, where XGB reaches its maximum F1-Macro of 0.707 ± 0.024 at the 93rd percentile before declining to 0.679 at the 85th percentile, suggesting that novel attacks in this dataset occupy a relatively compact high-anomaly-score region that is well separated from normal traffic. Taken together, these results confirm that the 95th percentile default used throughout this work sits comfortably inside the flat optimum region for all three benchmarks and both classifiers, and that overall detection quality is not brittle to moderate misspecification of this single hyperparameter.

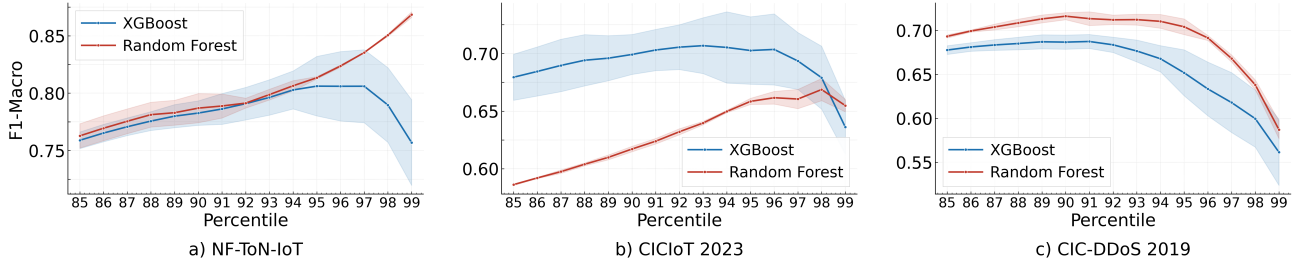


Fig. 7: Sensitivity analysis of the GMM threshold percentile across the evaluated datasets and models, reporting the mean and standard deviation over five runs.

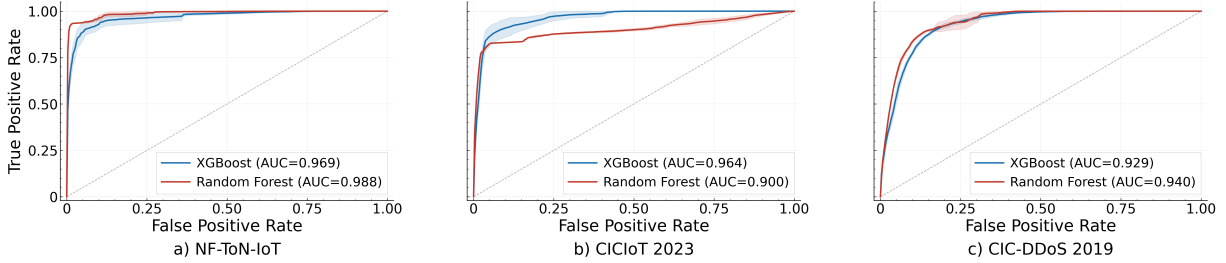


Fig. 8: ROC curves across the evaluated models and datasets, reporting the mean and standard deviation over five runs.

TABLE VI: Per-component inference latency breakdown (Per attack sample of a CIC-IoT-23).

Model	Model (ms)	Epistemic (ms)	SHAP (ms)	AE (ms)	GMM (ms)	Throughput (flows/s)
RF	11.57	11.15	301.18	17.34	0.32	2.93
XGB	2.55	2.42	6.21	16.02	0.33	36.33

The ROC curves in Figure 8 corroborate these findings by showing per-class discrimination quality at all operating points simultaneously. Both classifiers achieve mean AUROCs well above 0.90 on NF-ToN-IoT (RF: 0.989, XGB: 0.970) and CIC-DDoS 2019 (RF: 0.940, XGB: 0.929), and XGB reaches 0.964 on CICIoT-2023 while RF attains 0.900; all figures substantially exceed the naïve 0.5 random baseline and the simpler SHAP-only or epistemic-only variants reported in Tables III–V. The standard deviations around the mean ROC curves remain narrow across the five independent runs, underscoring that the pipeline’s discriminative power is reproducible rather than an artifact of a particularly favorable random seed.

E. Latency Analysis

Importantly, our method operates at the attack-identification stage. It does not introduce packet queuing or interfere with network traffic flow because it is not part of the online packet-processing pipeline. Instead, it is applied only after the base classifier has already flagged a flow as potentially anomalous, meaning that it performs a secondary refinement step on a reduced set of suspicious instances rather than all incoming traffic. With this in mind, Table VI reports the per-component inference latency for a single flagged flow, averaged over the attack samples of CIC-IoT-23.

For XGB, the pipeline is highly efficient: model inference completes in 2.55 ms, epistemic uncertainty estimation adds only 2.42 ms, and SHAP attribution requires 6.21 ms, yielding a throughput of 36.33 flows/s. For RF, SHAP is the dominant cost at 301.18 ms per sample, owing to the larger ensemble size; epistemic uncertainty estimation (11.15 ms) and model

inference itself (11.57 ms) are comparatively minor contributors. Across both classifiers, the autoencoder and GMM components introduce negligible overhead (≈ 17 ms and < 1 ms, respectively), confirming that the novelty-detection stage does not meaningfully affect end-to-end latency.

Our approach is lightweight and model-agnostic. Although computing SHAP values can be expensive in general, we rely on TreeSHAP, which offers an efficient and tractable way to obtain per-instance feature importance scores for tree-based models. The method operates on compact representations, specifically, a reduced set of SHAP feature importance values together with a single scalar uncertainty estimate, and uses only a simple autoencoder and a GMM defined over a two-dimensional space. As a result, no retraining or modification of the underlying classifier or feature extraction pipeline is required, making the approach suitable for seamless integration into existing NIDS deployments.

VII. CONCLUSION

In this paper, we addressed the problem of zero-day attack detection at the attack identification stage, focusing on traffic instances that are misclassified by a multiclass classifier as belonging to known attack categories. To tackle this challenge, we proposed a reconstruction–uncertainty–driven detection strategy in which per-instance SHAP feature importance vectors are reconstructed using an AE trained solely on known attacks, and the resulting RE is combined with predictive epistemic uncertainty to form a compact representation of known-class behavior modeled using a GMM. Extensive experiments conducted under a leave-one-attack-out protocol

across three benchmark datasets demonstrate that the proposed approach delivers consistent and reliable zero-day detection performance. Under XGB, our approach achieves best-case AUROC improvements approaching 40%, maintains strong average gains of 10–15%, and exhibits negligible worst-case degradation, while in terms of F1-score it yields substantial improvements across datasets, exceeding 70% for several attack categories. Even when RF is used, where epistemic uncertainty alone can perform competitively, our proposed method remains highly robust, consistently ranking among the top approaches and providing superior threshold-independent separability, demonstrating that jointly modeling reconstruction-based signals and predictive uncertainty provides a principled, effective, and generalizable solution for enhancing zero-day attack detection in modern NIDS pipelines.

While the proposed approach achieves strong and consistent performance, several opportunities for further improvement remain. Although SHAP-based feature attributions provide interpretability, they may introduce noise or instability in high-dimensional settings, so exploring more reliable feature behavior could yield more robust representations. Moreover, since the effectiveness of our detector depends on the reliability of the uncertainty estimates produced by the underlying classifier, developing models that simultaneously deliver strong predictive accuracy and meaningful uncertainty estimation could improve downstream zero-day detection performance. Our experiments instantiate the framework using tree-based classifiers (XGB and RF) via TreeSHAP, whereas extending it to non-tree-based classifiers is also left for future work. Also, while leave-one-attack-out experiments provides a controlled proxy for evaluating generalization to unseen attack types, it does not capture broader deployment-time distribution shifts, such as evolving benign and malicious traffic patterns or differences across networks. Evaluating the proposed framework under temporal evolution and cross-dataset shifts is therefore an important direction for future work.

ACKNOWLEDGMENT

This work has been supported by the EUREKA CELTIC-NEXT project SUSTAINET-Advance (No. 119.588 INT-ICT), funded by the Swiss Innovation Agency Innosuisse, and by the Italian Ministry of University and Research (MUR) under the PRIN 2022 PNRR framework (EU Contribution – NextGenerationEU – M. 4.C.2, I. 1.1), SHIELDED project, ID P2022ZWS82. Hussein Fawaz is supported by the Swiss Government Excellence Scholarship (ESKAS) No. 2024.0474.

REFERENCES

- [1] Kaur, Ramanpreet, et al. "Artificial intelligence for cybersecurity: Literature review and future research directions." *Information Fusion* 97 (2023): 101804.
- [2] Sowmya, Ta, and EA Mary Anita. "A comprehensive review of AI based intrusion detection system." *Measurement: Sensors* 28 (2023): 100827.
- [3] Dina, Ayesha S., and Dakshnamoorthy Manivannan. "Intrusion detection based on machine learning techniques in computer networks." *Internet of Things* 16 (2021): 100462.
- [4] Ahmad, Zeeshan, et al. "Network intrusion detection system: A systematic study of machine learning and deep learning approaches." *Transactions on Emerging Telecommunications Technologies* 32.1 (2021): e4150.
- [5] Zhang, Chunying, et al. "Comparative research on network intrusion detection methods based on machine learning." *Computers & Security* 121 (2022): 102861.
- [6] Ali, Shamshair, et al. "Comparative evaluation of ai-based techniques for zero-day attacks detection." *Electronics* 11.23 (2022): 3934.
- [7] Guo, Yang. "A review of Machine Learning-based zero-day attack detection: Challenges and future directions." *Computer communications* 198 (2023): 175-185.
- [8] Ahmad, Rasheed, et al. "Zero-day attack detection: a systematic literature review." *Artificial Intelligence Review* 56.10 (2023): 10733-10811.
- [9] Sommer, Robin, and Vern Paxson. "Outside the closed world: On using machine learning for network intrusion detection." 2010 IEEE symposium on security and privacy. IEEE, 2010.
- [10] Souza, Manuela MC, et al. "A novel open set energy-based flow classifier for network intrusion detection." *Computers & Security* 157 (2025): 104569.
- [11] Zoppi, Tommaso, et al. "Unsupervised algorithms to detect zero-day attacks: Strategy and application." *Ieee Access* 9 (2021): 90603-90615.
- [12] Aldhaheer, Alyazia, et al. "Deep learning for cyber threat detection in IoT networks: A review." *Internet of Things and cyber-physical systems* 4 (2024): 110-128.
- [13] Mbona, Innocent, and Jan HP Eloff. "Detecting zero-day intrusion attacks using semi-supervised machine learning approaches." *Ieee Access* 10 (2022): 69822-69838.
- [14] Yang, Zhen, et al. "A systematic literature review of methods and datasets for anomaly-based network intrusion detection." *Computers & Security* 116 (2022): 102675.
- [15] Talpini, Jacopo, et al. "Enhancing trustworthiness in ML-based network intrusion detection with uncertainty quantification." *Journal of Reliable Intelligent Environments* 10.4 (2024): 501-520.
- [16] Lundberg, Scott M. and Su-In Lee. "A Unified Approach to Interpreting Model Predictions." *Neural Information Processing Systems* (2017).
- [17] Liu, Jeremiah, et al. "Accurate uncertainty estimation and decomposition in ensemble learning." *Advances in neural information processing systems* 32 (2019).
- [18] Vilone, Giulia, and Luca Longo. "Explainable artificial intelligence: a systematic review." *arXiv preprint arXiv:2006.00093* (2020).
- [19] Fawaz, Hussein, et al. "Energy Cost of Enhancing Reliability of Machine Learning Models for Edge IoT Security." 21st Wireless On-Demand Network Systems and Services Conference (WONS). IEEE, 2026.
- [20] Kevin P. Murphy. "Probabilistic Machine Learning: Advanced Topics," MIT Press, 2023.
- [21] Hüllermeier, Eyke, and Willem Waegeman. "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods." *Machine learning* 110.3 (2021): 457-506.
- [22] Depeweg, Stefan, et al. "Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning." *International conference on machine learning*. PMLR, 2018.
- [23] Shaker, Mohammad Hossein, and Eyke Hüllermeier. "Aleatoric and epistemic uncertainty with random forests." *International Symposium on Intelligent Data Analysis*. Cham: Springer International Publishing, 2020.
- [24] Sharma, Bhawana, et al. "Anomaly based network intrusion detection for IoT attacks using deep learning technique." *Computers and Electrical Engineering* 107 (2023): 108626.
- [25] Yang, Liyan, et al. "Griffin: Real-time network intrusion detection system via ensemble of autoencoder in SDN." *IEEE Transactions on Network and Service Management* 19.3 (2022): 2269-2281.
- [26] Talpini, Jacopo, et al. "Hierarchical Multiclass Continual Learning for Network Intrusion Detection." 2024 IEEE 10th International Conference on Network Softwarization (NetSoft). IEEE, 2024.
- [27] Arafah, Mohammad, et al. "Anomaly-based network intrusion detection using denoising autoencoder and Wasserstein GAN synthetic attacks." *Applied Soft Computing* 168 (2025): 112455.
- [28] Alrayes, Fatma S., et al. "Intrusion detection in IoT systems using denoising autoencoder." *IEEE Access* (2024).
- [29] Dai, Zhen, et al. "An intrusion detection model to detect zero-day attacks in unseen data using machine learning." *PloS one* 19.9 (2024): e0308469.
- [30] Sana, Larab, et al. "Securing the IoT cyber environment: Enhancing intrusion anomaly detection with vision transformers." *IEEE Access* 12 (2024): 82443-82468.
- [31] Matejek, Brian, et al. "SAFE-NID: Self-attention with normalizing-flow encodings for network intrusion detection." *Transactions on Machine Learning Research* (2025).
- [32] Zhang, Milin, et al. "NI-Diff: Zero-Day and Adversarial Network Intrusion Detection with Diffusion Models." *MILCOM 2025-2025 IEEE Military Communications Conference (MILCOM)*. IEEE, 2025.
- [33] Shieh, Chin-Shiuh, et al. "Detection of unknown ddos attacks with deep learning and gaussian mixture model." *Applied Sciences* 11.11 (2021): 5213.

- [34] Zhang, Tianyue, et al. "An intrusion detection method based on stacked sparse autoencoder and improved gaussian mixture model." *Computers & Security* 128 (2023): 103144.
- [35] An, Peng, Zhiyuan Wang, and Chunjong Zhang. "Ensemble unsupervised autoencoders and Gaussian mixture model for cyberattack detection." *Information Processing & Management* 59.2 (2022): 102844.
- [36] Wang, Chao, et al. "Intrusion detection system based on one-class support vector machine and gaussian mixture model." *Electronics* 12.4 (2023): 930.
- [37] Wong, Joshua A., et al. "Uncertainty-quantified, robust deep learning for network intrusion detection." *2023 Winter Simulation Conference (WSC)*. IEEE, 2023.
- [38] Jorgensen, Steven, et al. "Extensible machine learning for encrypted network traffic application labeling via uncertainty quantification." *IEEE Transactions on Artificial Intelligence* 5.1 (2023): 420-433.
- [39] Yang, T., Qiao, Y., Lee, B. (2024). Towards trustworthy cybersecurity operations using Bayesian Deep Learning to improve uncertainty quantification of anomaly detection. *Computers & Security*, 144, 103909.
- [40] Yang, T., Qiao, Y., & Lee, B. (2024). Interpretable probabilistic bayesian neural networks for cybersecurity intrusion detection. *IEEE Access*.
- [41] Medjadba, Yasmine, et al. "Intrusion detection in Software-Defined Networking using hybrid Bayesian model averaging for reliable uncertainty quantification." *Computer Networks* (2025): 111436.
- [42] Gaspar, Diogo, et al. "Explainable ai for intrusion detection systems: Lime and shap applicability on multi-layer perceptron." *IEEE Access* (2024).
- [43] Arreche, Osvaldo, et al. "E-XAI: Evaluating black-box explainable AI frameworks for network intrusion detection." *IEEE Access* 12 (2024): 23954-23988.
- [44] Abou El Houda, Z., et al. (2022). "why should i trust your ids?": An explainable deep learning framework for intrusion detection systems in internet of things networks. *IEEE Open Journal of the Communications Society*, 3, 1164-1176.
- [45] Gummedi, Anna Namrita, et al. "XAI-IoT: an explainable AI framework for enhancing anomaly detection in IoT systems." *IEEE Access* 12 (2024): 71024-71054.
- [46] Sharma, Bhawana, et al. "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach." *Expert Systems with Applications* 238 (2024): 121751.
- [47] Patil, Shruti, et al. "Explainable artificial intelligence for intrusion detection system." *Electronics* 11.19 (2022): 3079.
- [48] Mahmoud, Maiada M., et al. "XI2S-IDS: An Explainable Intelligent 2-Stage Intrusion Detection System." *Future Internet* 17.1 (2025): 25.
- [49] Hooshmand, M.K., et al. "Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (XAI)." *Alexandria Engineering Journal* 94 (2024): 120-130.
- [50] H. Fawaz, et al. "Towards Better-Calibrated ML Models for Reliable Network Intrusion Detection via Calibration-Aware SHAP-Based Feature Selection," *21st International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2025.
- [51] Barnard, Pieter, et al. "Robust network intrusion detection through explainable artificial intelligence (xai)." *IEEE Networking Letters* 4.3 (2022): 167-171.
- [52] Mariotti, Ettore, et al. "Measuring model understandability by means of Shapley additive explanations." *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2022.
- [53] G. Schwarz, "Estimating the Dimension of a Model," *Ann. Statist.*, vol. 6, no. 2, pp. 461 – 464, 1978.
- [54] Moon, Todd K. "The expectation-maximization algorithm." *IEEE Signal processing magazine* 13.6 (1996): 47-60.
- [55] Luay, Majed, et al. "Temporal Analysis of NetFlow Datasets for Network Intrusion Detection Systems." *arXiv preprint arXiv:2503.04404* (2025).
- [56] Neto, Euclides Carlos Pinto, et al. "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment." *Sensors* 23.13 (2023): 5941.
- [57] Sharafaldin, Iman, et al. "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy." *2019 international carnahan conference on security technology (ICCST)*. IEEE, 2019.

APPENDIX A
PER-ATTACK AUROC RESULTS
TABLE VII: NF-TON-IoT: AUROC

Attack	Training	EFC	XGB				RF			
			SHAP	Epistemic	SHAP+Epi	Ours	SHAP	Epistemic	SHAP+Epi	Ours
Backdoor	0.992	0.942	0.942	0.353	0.938	0.947	0.894	0.999	0.888	0.999
DDoS	0.824	0.788	0.989	0.015	0.988	0.997	0.947	0.798	0.949	0.953
DoS	0.956	0.986	0.826	0.380	0.838	0.878	0.934	0.899	0.933	0.963
Injection	0.955	0.773	0.969	0.076	0.969	0.986	0.967	0.991	0.968	0.994
MITM	0.989	0.931	0.960	0.986	0.958	0.989	0.812	0.999	0.835	0.999
Password	0.978	0.711	0.980	0.239	0.979	0.974	0.928	0.998	0.927	0.997
Ransomware	0.986	0.844	0.999	0.370	0.999	0.999	0.953	0.999	0.954	0.999
Scanning	0.942	0.974	0.931	0.982	0.935	0.967	0.975	0.999	0.975	0.999
XSS	0.974	0.796	0.896	0.195	0.895	0.987	0.979	0.995	0.979	0.998
Avg. AUROC	0.955	0.861	0.944	0.400	0.945	0.970	0.932	0.964	0.934	0.989
Std. AUROC	0.052	0.100	0.054	0.355	0.051	0.038	0.052	0.071	0.047	0.018

TABLE VIII: CICIOT-23: AUROC

Attack	Training	EFC	XGB				RF			
			SHAP	Epistemic	SHAP+Epi	Ours	SHAP	Epistemic	SHAP+Epi	Ours
BruteForce	0.953	0.943	0.895	0.005	0.895	0.982	0.925	0.994	0.924	0.988
DDoS	0.685	0.706	0.983	0.501	0.983	0.980	0.695	0.784	0.710	0.745
DoS	0.603	0.578	0.817	0.878	0.810	0.867	0.632	0.563	0.641	0.591
Mirai	0.978	0.869	0.956	0.023	0.958	0.972	0.988	0.999	0.988	0.996
Recon	0.983	0.944	0.844	0.038	0.848	0.979	0.854	0.994	0.844	0.995
Spoofing	0.990	0.951	0.803	0.023	0.808	0.990	0.826	0.998	0.829	0.997
Web	0.963	0.935	0.761	0.016	0.755	0.979	0.933	0.988	0.933	0.989
Avg. AUROC	0.879	0.847	0.866	0.212	0.865	0.964	0.836	0.903	0.838	0.900
Std. AUROC	0.163	0.147	0.082	0.344	0.084	0.043	0.131	0.169	0.125	0.165

TABLE IX: CIC-DDoS2019: AUROC

Attack	Training	EFC	XGB				RF			
			SHAP	Epistemic	SHAP+Epi	Ours	SHAP	Epistemic	SHAP+Epi	Ours
DNS	0.685	0.915	0.739	0.061	0.739	0.927	0.836	0.876	0.836	0.926
LDAP	0.767	0.893	0.569	0.042	0.560	0.910	0.840	0.906	0.839	0.934
MSSQL	0.821	0.917	0.764	0.025	0.757	0.957	0.831	0.977	0.831	0.972
NTP	0.860	0.963	0.923	0.170	0.922	0.918	0.951	0.992	0.954	0.991
NetBIOS	0.776	0.914	0.864	0.056	0.877	0.961	0.878	0.939	0.878	0.942
Portmap	0.756	0.937	0.840	0.075	0.847	0.945	0.897	0.952	0.897	0.958
SNMP	0.662	0.898	0.577	0.017	0.570	0.937	0.837	0.950	0.838	0.944
Syn	0.992	0.241	0.813	0.224	0.809	0.909	0.839	0.989	0.837	0.983
TFTP	0.605	0.630	0.985	0.362	0.984	0.971	0.842	0.984	0.854	0.947
UDP	0.579	0.773	0.794	0.741	0.783	0.802	0.792	0.569	0.794	0.792
UDPLag	0.773	0.795	0.904	0.170	0.906	0.935	0.938	0.858	0.939	0.920
WebDDoS	0.959	0.988	0.966	0.058	0.972	0.974	0.978	0.936	0.979	0.977
Avg. AUROC	0.770	0.822	0.812	0.167	0.811	0.929	0.871	0.911	0.873	0.940
Std. AUROC	0.128	0.208	0.135	0.208	0.139	0.046	0.057	0.116	0.057	0.052