E2D2: A Modular Framework for Explanation-Based Drift Detection in Unsupervised Network Intrusion Detection

Beny Nugraha and Thomas Bauschert
Chair of Communication Networks
Chemnitz University of Technology, Germany
{beny.nugraha, thomas.bauschert}@etit.tu-chemnitz.de

***Abstract*—Unsupervised Intrusion Detection Systems (IDS) become less accurate when network traffic patterns change over time, a problem known as concept drift. Existing drift detection methods can indicate that a shift has occurred, but typically provide limited information about which traffic features contributed to the change. We propose E2D2 (Explanation-to-Drift Detection), a modular framework that detects concept drift by monitoring how the feature-level explanations of an unsupervised IDS evolve over time. E2D2 computes a continuous Explanation Drift Intensity (XDI) score from per-window explanation fingerprints, raises threshold-based alarms, and outputs compact feature-level drift explanations. The framework is modular along four design axes: IDS model, attribution method, population view, and aggregation strategy. We evaluate E2D2 across five network traffic benchmarks, four drift scenarios, two detection models, and four attribution methods. The compositional analysis shows that the population view has the largest effect on detection quality, with an average F1 difference of 0.130 between its best and worst options. In its best configuration, E2D2 achieves an F1 Score of 0.848 with alarm filtering while providing compact 1–2 feature drift explanations that capture 88–93% of the overall drift magnitude.**

***Index Terms*—Concept Drift, Intrusion Detection Systems, Explainable AI, Variational Autoencoder, Network Security, Drift Detection.**

I. Introduction

Modern network traffic is increasingly diverse and evolving, making unsupervised anomaly detection an important approach for intrusion detection because it can operate without fully labeled attack data [1], [2]. Deep autoencoder-based approaches, including standard Autoencoders (AEs) [2] and Variational Autoencoders (VAEs) [3], can detect novel attacks without requiring labeled training data. However, benign network traffic naturally evolves over time due to infrastructure changes, shifting user behaviors, or protocol updates [4], [5]. When the learned benign reference no longer matches the current traffic distribution, static models generate a higher false alarm rate [6], [7].

To maintain detection accuracy, the IDS must detect when the underlying traffic distribution has changed. Existing drift detection methods monitor raw feature statistics [8], reconstruction error ratios [6], or anomaly score distributions [7]. While these methods can indicate drift, they usually provide limited information about which features drove the change or how the IDS behavior shifted, although such information is essential for deciding whether the change reflects a benign infrastructure update or a potential security threat [9]. Recent work has started using Explainable AI (XAI) outputs for drift detection. Lee et al. [10] proposed monitoring the SHAP-based Mahalanobis distance to detect model drift in unsupervised settings, and Haug et al. [11] studied changes in local explanations over time in evolving data streams. These studies show the promise of explanation-aware drift monitoring, but they do not provide a modular analysis of how the IDS model, attribution method, and monitored traffic population affect detection quality in unsupervised IDS.

To address these limitations, we propose E2D2 (Explanation-to-Drift Detection), a modular framework for detecting and explaining concept drift in unsupervised IDS. E2D2 generates three outputs: a continuous XDI time series that quantifies drift intensity over time, binary drift alarms, and a compact drift summary identifying the features most associated with the detected shift. The framework is modular along four design axes, enabling a systematic compositional analysis of how IDS model, attribution method, population view, and aggregation strategy jointly affect drift detection quality.

We structure our contributions around four Research Questions (RQs). **RQ1:** Can explanation-space monitoring detect diverse types of drift in unsupervised IDS with competitive performance? **RQ2:** Which design choices most influence E2D2's drift detection quality? **RQ3:** How does E2D2 compare to established drift detectors? **RQ4:** How compact and informative are E2D2's drift explanations?

The remainder of this paper is organized as follows. Section II reviews related work, Section III presents the E2D2 methodology, Section IV outlines the experimental setup, Section V discusses the empirical results, and Section VI concludes the paper.

II. Related Work

A. Concept Drift Detection in IDS

Existing drift detection methods for IDS operate at different levels of abstraction. Statistical methods such as ADWIN [8]

ISBN 978-3-903176-82-9

monitor scalar signals like anomaly scores or error rates. These methods can detect that a change has occurred, but they do not indicate which features are responsible for the shift.

Feature-space methods such as PCA-based change detection [12] operate directly on input distributions and can localize drift to specific features. However, they do not account for how the IDS model internally processes those features, so a feature shift that is irrelevant to the model's decisions may still trigger an alarm.

Model-internal approaches address this limitation by using outputs derived from the detection model itself. CADE [13] learns to distinguish old from new samples to detect drift in security applications, and Lee et al. [10] proposed monitoring SHAP-based explanation shifts in unsupervised anomaly detectors. However, SHAP-based methods can incur high per-sample cost, particularly on high-dimensional inputs [14]. Haug et al. [11] studied change detection for local explanations, but their approach targets supervised classification and does not provide compact drift explanations for unsupervised IDS.

B. Explainable AI for Drift Detection

Post-hoc methods such as KernelSHAP [15] provide informative feature attributions but are often too expensive for online use [9]. Faster gradient-based methods, including GEE [16], Integrated Gradients [17], and Input×Gradient [18], are more suitable for streaming settings, but are typically used to explain individual alerts rather than to monitor how explanation patterns evolve over time.

Recent work has begun to connect explainability and drift detection [11], [19], but prior work has not studied window-level explanation fingerprints for drift detection in unsupervised IDS together with a compositional analysis across detection model, attribution method, and population view. E2D2 addresses this gap by monitoring explanation fingerprints over time, comparing the main design axes, and generating compact feature-level drift explanations for analyst interpretation.

III. The E2D2 Framework

A. Overview

As illustrated in Fig. 1, E2D2 operates in two stages: offline preparation and online drift monitoring. During offline preparation, ANOVA F-test and TreeSHAP [14] are used to retain the most informative features, the IDS model (AE or VAE) is trained on benign reference traffic, and robust attribution scales are estimated from benign explanations.

During online monitoring, the trained IDS model computes an anomaly score for each incoming sample, and a feature attribution method, such as raw gradient, Input×Gradient, Integrated Gradients, or SmoothGrad, computes a per-feature explanation. Within each sliding window, a population view defines which samples are summarized: all samples, benign-like samples with low anomaly scores, or suspicious-like samples with high anomaly scores. Each selected group is then summarized into an explanation fingerprint that captures the main attribution pattern of the window.

E2D2 compares each fingerprint with a benign reference fingerprint to measure explanation shift. It computes a standardized shift score for each feature, aggregates these feature-level shifts into a scalar Explanation Drift Intensity (XDI) score, and raises a drift alarm when XDI exceeds a calibrated threshold. For each alarm, E2D2 also returns a minimal drift explanation identifying the features most responsible for the shift.

As highlighted in Fig. 1, E2D2 is modular along four design axes: the IDS model, the attribution method, the population view, and the XDI aggregation strategy. This design allows different combinations to be selected for different deployment settings. We focus on AE and VAE because both are representative reconstruction-based IDS models that naturally support gradient-based attribution.

B. Attribution Computation and Normalization

For each input sample $x \in \mathbb{R}^D$, the IDS model computes an anomaly score $s(x)$ based on reconstruction error. E2D2 then computes a feature attribution vector $a(x) \in \mathbb{R}^D$, where each entry indicates how much the corresponding feature contributes to the anomaly score. We evaluate four attribution methods: raw gradient [16], Input×Gradient [18], Integrated Gradients [17], and SmoothGrad [20].

Because attribution magnitudes differ across features, direct comparison would be misleading. E2D2 therefore normalizes attributions using robust statistics computed from benign reference data. For each feature j, the Median Absolute Deviation (MAD) is defined as

$$\mathrm{MAD}_j = \mathrm{median}(\{\, |a(x)_j - \mathrm{median}(a(\cdot)_j)| : x \in \mathcal{D}_{\mathrm{ref}} \,\}), \tag{1}$$

where $\mathcal{D}_{\mathrm{ref}}$ is the benign reference set. This MAD serves as a per-feature scale factor, so that a given shift is interpreted relative to the feature's normal variation.

C. Drift Monitoring and Alarm Generation

The incoming traffic stream is partitioned into overlapping sliding windows of $W = 500$ samples with stride $S = 250$, so consecutive windows share 50% of their samples. Within each window t, samples are grouped according to the selected population view p: all samples, benign-like samples with anomaly scores below the 95th percentile of the reference distribution, or suspicious-like samples with scores above that threshold.

For each window and population, E2D2 summarizes the normalized attributions into an explanation fingerprint defined by the per-feature median and MAD. The benign reference fingerprint is constructed from the first $B = 5$ windows by computing, for each feature, the median fingerprint value across those windows, providing a stable reference for normal explanation behavior.

For each subsequent window, E2D2 measures the shift of each feature j from the benign reference using

$$z_t^{(p)}[j] = \frac{\mu_t^{(p)}[j] - \mu_0^{(p)}[j]}{\max(m_0^{(p)}[j],\, m_{\min}) + \epsilon}, \tag{2}$$

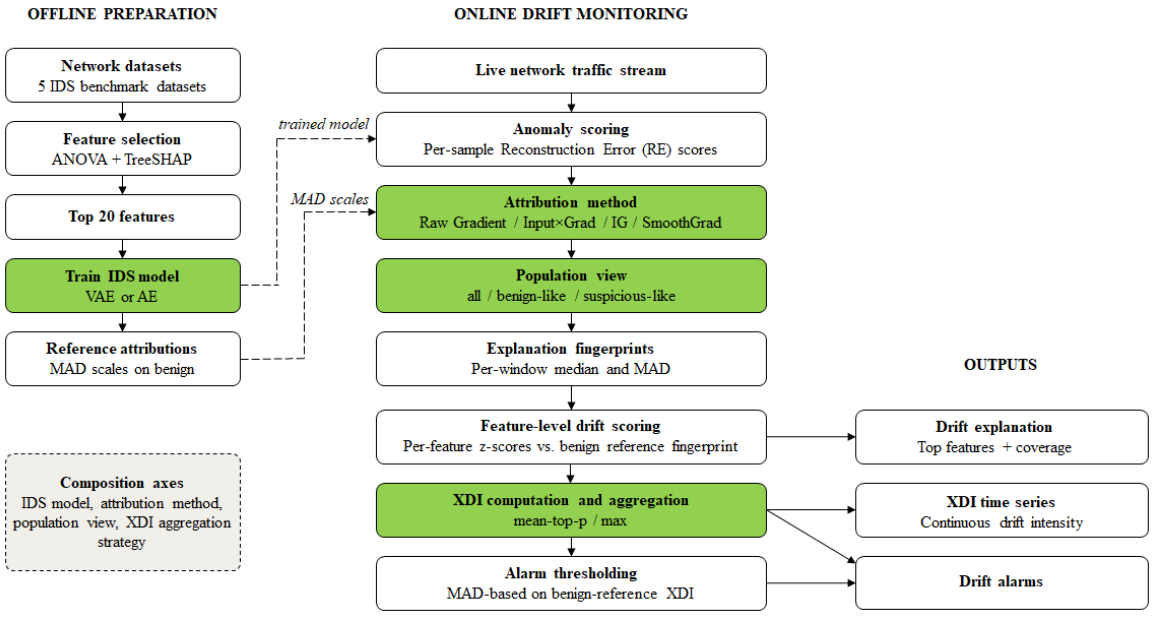

Fig. 1. The E2D2 end-to-end workflow.

where $\mu_t^{(p)}[j]$ and $\mu_0^{(p)}[j]$ are the current and benign-reference median attributions of feature j, and $m_0^{(p)}[j]$ is the reference MAD. The constant $m_{\min} = 0.05$ prevents unstable scores when the reference MAD is very small, and $\epsilon = 10^{-8}$ is added for numerical stability. Larger $|z_t^{(p)}[j]|$ values indicate stronger shifts relative to normal variation. To prevent very large values from dominating the final drift score, z-scores are limited to $[-10, 10]$.

E2D2 then aggregates the feature-level shifts into a scalar Explanation Drift Intensity (XDI) score. In the *mean-top-p* strategy, features are ranked by $|z_t[j]|$ and only the top $k = \lceil D \cdot p \rceil$ most shifted features are averaged; in the *max* strategy, XDI is the single largest shift:

$$\mathrm{XDI}(t) = \frac{1}{k} \sum_{j \in \text{top-k}} |z_t[j]| \quad \text{(mean-top-}p\text{)}, \tag{3}$$

$$\mathrm{XDI}(t) = \max_j |z_t[j]| \quad \text{(max)}. \tag{4}$$

With $D = 20$ and $p = 0.25$, mean-top-p averages the 5 most shifted features, capturing drift spread across several features, whereas max focuses on the strongest individual shift.

A window is flagged as drifted when its XDI exceeds a threshold calibrated from the reference period:

$$\theta = \mathrm{median}(\mathrm{XDI}_{\mathcal{B}}) + \kappa \cdot \mathrm{MAD}(\mathrm{XDI}_{\mathcal{B}}), \tag{5}$$

where $\kappa = 2.0$ controls alarm sensitivity. Optionally, E2D2 can require N_c consecutive windows above $\lambda \cdot \theta$ before raising an alarm, reducing false alarms from isolated spikes. The default uses $\lambda = 1.0$ and $N_c = 1$ (no filtering); the effect of tuning these parameters is evaluated in Section V-C.

For each flagged window, E2D2 generates a minimal drift explanation by ranking features by $|z_t[j]|$ and selecting the smallest set whose cumulative contribution reaches at least 80% of the total drift magnitude (i.e., the total amount of feature-level shift in that window). This usually yields one or two features, giving a compact summary of the traffic characteristics responsible for the detected shift.

IV. Experimental Setup

A. Datasets, Preprocessing, and Drift Scenarios

We evaluated E2D2 on five network intrusion detection datasets: the CIC-DDoS2019 NTP, Portmap, and Syn subsets [21], CICIoT2023 [22], and 5GNIDD [23]. The implementation is available online.[1] For each dataset, we removed non-informative columns, invalid rows, and zero-variance features. The remaining features were normalized using Min-Max scaling fitted exclusively on benign reference data. We then used ANOVA F-test as an initial filter and TreeSHAP global importance [14] to rank the remaining features, retaining the top 20 features per dataset.

Each dataset was split into 60% benign reference data, used for IDS model training, per-feature attribution normalization, and benign reference fingerprint construction, and 40% stream data, used for drift scenario construction and evaluation. Both the VAE and AE models use an encoder with hidden layers of 64 and 32 neurons and a 16-dimensional latent space, followed by a symmetric decoder. The models were trained on benign traffic for 50 epochs with a batch size of 256, a learning rate of 10^{-3}, and early stopping with a patience of 10 epochs. The E2D2 hyperparameters were set as follows: $W = 500$, $S = 250$, $p = 0.25$, $\kappa = 2.0$, $B = 5$. These values were chosen empirically during an initial development phase to provide stable behavior and were then kept fixed for all datasets and

[1] github.com/DLTeamTUC/E2D2

TABLE I
DRIFT SCENARIOS

Scenario	Traffic	Drift Mechanism
Gradual Infra	Benign only	5 features gradually scaled to $\times 2.0$ over 30% of the stream (simulates infrastructure upgrade)
Sudden Attack	Benign $\rightarrow$ mixed	30% of samples replaced with real attacks (simulates new threat)
Attack Shift	Mixed (20% attacks)	Noise (σ=0.5) added to attack features in second half (simulates attacker behavior change)
Benign Drift	Benign only	Abrupt Gaussian shift (σ=1.5) on 5 features (simulates sudden user behavior change)

scenarios to evaluate E2D2 under a consistent configuration without per-dataset tuning.

Table I summarizes the four drift scenarios. In all cases, drift begins at the stream midpoint. For Gradual Infrastructure Drift and Benign Concept Drift, 5 of the 20 retained features are randomly selected as drift targets. Gradual Infrastructure Drift scales these features gradually to $\times 2.0$ over 30% of the stream, while Benign Concept Drift applies an abrupt Gaussian shift with $\sigma = 1.5$. Sudden Attack replaces 30% of post-midpoint samples with real attacks, and Attack Shift adds Gaussian noise with $\sigma = 0.5$ only to attack samples in the second half.

B. Baselines and Evaluation Metrics

We compared E2D2 against four drift detection baselines: ADWIN [8], PCA-based drift detection [12], KL-divergence monitoring [24], and an adapted SHAP-Mahalanobis baseline inspired by Lee et al. [10]. For E2D2, we evaluated two unsupervised models (VAE and AE), four attribution methods (raw gradient, Input$\times$Gradient, Integrated Gradients, and SmoothGrad), three population views (all, benign-like, suspicious-like), and two XDI aggregation strategies (mean-top-p and max), yielding 48 configurations per scenario.

For drift detection quality, we report precision, recall, and F1 Score against ground-truth drift regions; detection delay, measured in windows from the drift midpoint to the first correct alarm; false alarm rate (FAR) per 100 non-drift windows; and AUC (Area Under the ROC Curve), which measures how well the continuous XDI score separates drift windows from non-drift windows across all possible threshold values; an AUC of 1.0 indicates perfect separation, while 0.5 indicates no separation ability. For explanation quality of E2D2, we report coverage, defined as the fraction of total drift magnitude captured by the minimal explanation set, the number of selected features, and Jaccard similarity with the ground-truth drift features. We also discuss computational cost to assess practical deployment trade-offs.

V. RESULTS AND DISCUSSION

A. RQ1: Detection Performance Across Drift Types

We first evaluated whether E2D2 can detect all four drift types using a default configuration with a VAE model, raw gradient attribution, all-sample population, mean-top-p aggregation, and no alarm filtering (i.e., $\lambda = 1.0$ and $N_c = 1$; see Section III-C). This serves as a reference configuration before the compositional analysis in Section V-B. Table II reports the detection metrics averaged across all five datasets for each scenario. Benign Drift achieved the strongest results (F1 Score of 0.809, AUC of 0.961), followed by Sudden Attack (F1 Score of 0.757) and Gradual Infra (F1 Score of 0.739). Attack Shift was the most challenging scenario (F1 Score of 0.678, delay of 2.2 windows), although the continuous XDI signal still provided useful separation between drift and non-drift windows (AUC of 0.773). Across datasets, Portmap achieved the highest average F1 Score (0.837), while CICIoT2023 achieved the lowest (0.587).

TABLE II
E2D2 DETECTION PERFORMANCE ACROSS DRIFT TYPES

Scenario	Prec.	Rec.	F1 Score	Delay	FAR	AUC
Gradual Infra	0.675	0.889	0.739	0.6	51.6	0.909
Sudden Attack	0.677	0.933	0.757	0.2	52.9	0.851
Attack Shift	0.728	0.707	0.678	2.2	23.6	0.773
Benign Drift	0.696	1.000	0.809	0.0	50.4	0.961
Average	0.694	0.882	0.746	0.8	44.6	0.874

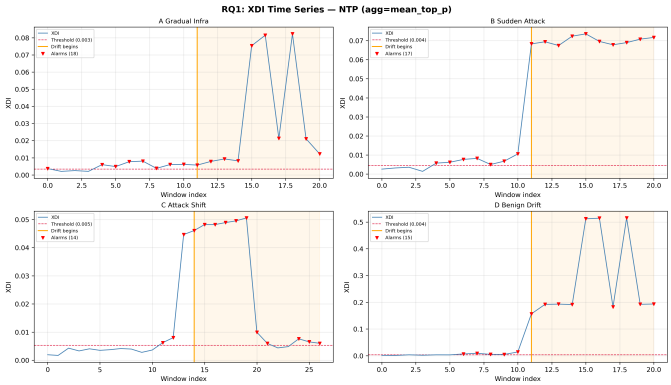

Fig. 2. XDI time series across four drift scenarios on the CIC-DDoS2019 NTP dataset. The orange vertical line marks where drift begins, the dashed red line shows the alarm threshold, and red triangles indicate alarm points.

The default unfiltered configuration also generated a high FAR, with an average 44.6. Simply raising the threshold would reduce FAR but would increase detection delay and could potentially miss weaker drift events; alarm filtering addresses this more effectively by requiring consecutive windows above the threshold rather than a single spike. The effect of alarm filtering on this trade-off is examined in Section V-C.

Fig. 2 illustrates the XDI time series on the CIC-DDoS2019 NTP dataset under the default E2D2 configuration. It shows that XDI provides continuous drift monitoring rather than only binary alarms, that different drift types generate different XDI patterns over time, and where alarms are triggered. Scenario C has more windows because it is constructed from a mixed stream with attack samples, while the other scenarios use benign-only pre-drift traffic. The pre-drift XDI values and thresholds also differ because each scenario is generated independently and calibrated from its own reference windows.

TABLE III
IMPORTANCE OF DESIGN AXES AND BEST OPTIONS IN E2D2

Design Axis	ΔF1 Score	ΔAUC	Best Option
Population	0.130	0.122	all
IDS Model	0.039	0.014	VAE
Attribution Method	0.020	0.039	input×gradient
Aggregation	0.018	0.027	mean-top-p

TABLE IV
E2D2-BEST VS. DRIFT DETECTION BASELINES

Method	Prec.	Rec.	F1 Score	Delay	FAR	AUC
E2D2-best	0.725	0.950	0.802	**0.20**	45.3	0.857
E2D2-best (filtered)	0.812	0.932	0.848	0.40	29.6	0.857
Adapted SHAP-Mahalan. inspired by Lee et al. [10]	0.764	0.893	0.810	**0.20**	29.7	0.895
PCA-CD	0.828	**0.951**	**0.868**	0.40	25.1	**0.951**
KL-Divergence	**0.852**	0.855	0.801	**0.20**	**23.3**	0.929
ADWIN	0.668	0.882	0.706	0.35	53.7	0.869

B. RQ2: Compositional Analysis

We next examined which design axes most influence E2D2's drift detection quality. Table III summarizes this analysis. The design axes are ordered by ΔF1 Score, our primary detection metric, while ΔAUC is reported as supporting evidence for the same overall pattern. For each design axis, ΔF1 Score and ΔAUC quantify how strongly F1 Score and AUC change when switching between the different options of that axis, while the Best Option denotes the option with the highest average F1 Score. Because the analysis is averaged across datasets and scenarios, it should be interpreted as a global trend rather than a per-dataset guarantee.

The population view had the largest effect under both metrics (ΔF1 Score = 0.130, ΔAUC = 0.122), making it the most influential design axis. The remaining axes had smaller effects. Under ΔF1 Score, the order was IDS model (ΔF1 Score = 0.039, ΔAUC = 0.014), attribution method (ΔF1 Score = 0.020, ΔAUC = 0.039), and aggregation strategy (ΔF1 Score = 0.018, ΔAUC = 0.027), while ΔAUC shows the same overall pattern with a small difference in the middle ranks.

For the population view, the best option was *all*. Monitoring the full window provides the most stable drift signal, whereas splitting the window into benign-like or suspicious-like subsets reduces the effective sample size and may discard informative samples. The suspicious-like option performed worst, likely because drift changes the anomaly score distribution itself, making this split less stable and excluding samples that still contain useful drift information.

For the detection model, VAE outperformed AE. Overall, the results indicate that E2D2 is influenced more by which part of the traffic is monitored than by the specific attribution or aggregation choice, while the detection model also has a noticeable, but smaller, effect. Still, as shown in Section V-C, these smaller differences remain important when selecting a complete composition, because the best overall configuration does not simply follow from combining the individually best options.

C. RQ3: Comparison with Baselines

Section V-B reported the best individual option for each design axis based on marginal average F1 Score. However, the best composition was identified by directly comparing all 48 evaluated compositions across all datasets and scenarios. Under this full-composition comparison, the highest average F1 Score was obtained by VAE, Integrated Gradients, all population, and max aggregation. This differs from the combination of individually best options in Table III, indicating interaction effects between design axes, particularly between attribution method and aggregation.

Table IV compares this best composition against four drift detection baselines, averaged across five datasets and four scenarios. Without alarm filtering ($\lambda = 1.0$, $N_c = 1$; see Section III-C), E2D2-best achieved a recall of 0.950, a delay of 0.20 windows, and a FAR of 45.3. However, as shown in Section V-A, this unfiltered setting is sensitive to isolated noise spikes. To study alarm filtering, we evaluated multiple combinations of λ and N_c and selected ($\lambda = 1.25$, $N_c = 3$), which gave the best overall balance between improved F1 Score, reduced FAR, and limited recall loss. With this setting, F1 Score improved from 0.802 to 0.848, FAR decreased from 45.3 to 29.6, and recall decreased only slightly from 0.950 to 0.932. E2D2's filtered F1 Score exceeded the adapted SHAP-Mahalanobis baseline inspired by Lee et al. [10] (0.810), KL-Divergence (0.801), and ADWIN (0.706), while remaining competitive with PCA-CD (0.868), differing by only 0.020.

Beyond detection metrics, E2D2 offers two practical advantages. First, it generates compact feature-level drift explanations rather than only signaling that drift occurred, as examined in Section V-D. Second, it is attribution-method-agnostic, unlike the adapted SHAP-Mahalanobis baseline. In our implementation, raw gradient and Input×Gradient process a benchmark stream in about 0.3 s, while Integrated Gradients requires about 30 s because it uses multiple gradient evaluations. This allows practitioners to trade some accuracy for lower computational cost when latency is important.

D. RQ4: Drift Explanation Quality

Table V reports explanation quality for the three scenarios where the ground-truth perturbed features are known, averaged across all five datasets. The Sudden Attack scenario is excluded because it uses real attack samples without the ground-truth perturbed features. Across the remaining scenarios, E2D2 captured 88–93% of the overall drift magnitude captured by the explanation scores using only 1–2 features, indicating that most of the detected shift is concentrated in a small subset of features.

Jaccard similarity between the selected and ground-truth drift features was low (0.120–0.240). This is expected because E2D2 selects only 1–2 features per alarm, whereas 5 features were perturbed, which limits the possible overlap with the full ground-truth feature set. Moreover, the ground truth itself

TABLE V
EXPLANATION QUALITY: COVERAGE AND COMPACTNESS

Scenario	Coverage	Features	Jaccard
Gradual Infra	0.93	1.4	0.240
Attack Shift	0.88	1.6	0.120
Benign Drift	0.91	2.0	0.220

is limited to the artificially perturbed features, while E2D2 explains the features that dominate the attribution shift. These results should therefore be interpreted as evidence of compact dominant-drift explanations, rather than as complete identification of all perturbed features.

VI. CONCLUSION

In this paper we present E2D2, a modular framework for detecting and explaining concept drift in unsupervised IDS by monitoring how feature-level explanations change over time. E2D2 supports different IDS models, attribution methods, population views, and aggregation strategies, allowing practitioners to adapt the framework to different deployment needs. Across five datasets and four drift scenarios, E2D2 achieved competitive detection quality against established baselines (F1 Score up to 0.848) while generating compact drift explanations based on 1–2 features that captured 88–93% of the overall drift magnitude. The compositional analysis showed that the population view has the largest effect on drift detection quality, and that the best full composition does not simply follow from the individually best options of each design axis. In our evaluation, the strongest overall composition used all-sample monitoring, a VAE model, Integrated Gradients, and max aggregation, while the framework also supports faster attribution methods when lower latency is required.

Future work will investigate adaptive thresholding, additional unsupervised IDS backbones beyond AE/VAE, repeated-split robustness analysis, and naturally occurring drift conditions.

ACKNOWLEDGMENT

This work has been performed in the framework of the SUSTAINET-Advance project, funded by the German BMFTR (ID:16KIS2280).

REFERENCES

[1] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[2] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2018.

[3] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

[4] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys*, vol. 46, no. 4, pp. 1–37, 2014.

[5] A. Kuppa and N.-A. Le-Khac, "Learn to adapt: Robust drift detection in security domain," *Computers and Electrical Engineering*, vol. 102, p. 108239, 2022.

[6] B. Nugraha, K. Yadav, and T. Bauschert, "A novel adaptive concept drift detection approach for evolving network traffic patterns," in *2025 9th Cyber Security in Networking Conference (CSNet)*, Abu Dhabi, United Arab Emirates, 2025, pp. 1–8.

[7] M. A. Shyaa, N. F. Ibrahim, Z. Zainol, R. Abdullah, M. Anbar, and L. Alzubaidi, "Evolving cybersecurity frontiers: A comprehensive survey on concept drift and feature dynamics aware machine and deep learning in intrusion detection systems," *Engineering Applications of Artificial Intelligence*, vol. 137, p. 109143, 2024.

[8] A. Bifet and R. Gavaldà, "Learning from time-changing data with adaptive windowing," in *Proceedings of the SIAM International Conference on Data Mining*, 2007, pp. 443–448.

[9] A. L. Samed and S. Sagiroglu, "Explainable artificial intelligence models in intrusion detection systems," *Engineering Applications of Artificial Intelligence*, vol. 144, p. 110145, 2025.

[10] Y. Lee, Y. Lee, E. Lee, and T. Lee, "Explainable artificial intelligence-based model drift detection applicable to unsupervised environments," *Computers, Materials & Continua*, vol. 76, no. 2, pp. 1701–1718, 2023.

[11] J. Haug, A. Braun, S. Zürn, and G. Kasneci, "Change detection for local explainability in evolving data streams," in *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM)*, 2022, pp. 706–716.

[12] L. I. Kuncheva and W. J. Faithfull, "PCA feature extraction for change detection in multidimensional unlabelled data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 69–80, 2014.

[13] L. Yang, W. Guo, Q. Hao, A. Ciptadi, A. Ahmadzadeh, X. Xing, and G. Wang, "Cade: Detecting and explaining concept drift samples for security applications," in *Proceedings of the 30th USENIX Security Symposium*, 2021, pp. 2327–2344.

[14] B. Nugraha, A. V. Jnanashree, and T. Bauschert, "A versatile xai-based framework for efficient and explainable intrusion detection systems," *Annals of Telecommunications*, vol. 80, pp. 1095–1120, 2025.

[15] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[16] Q. P. Nguyen, K.-W. Lim, D. M. Divakaran, K.-H. Low, and M. C. Chan, "Gee: A gradient-based explainable variational autoencoder for network anomaly detection," in *2019 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2019, pp. 91–99.

[17] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, vol. 70, 2017, pp. 3319–3328.

[18] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017, pp. 3145–3153.

[19] D. Pelosi, D. Cacciagrano, and M. Piangerelli, "Explainability and interpretability in concept and data drift: A systematic literature review," *Algorithms*, vol. 18, no. 7, p. 443, 2025.

[20] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smoothgrad: Removing noise by adding noise," *arXiv preprint arXiv:1706.03825*, 2017.

[21] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (ddos) attack dataset and taxonomy," in *Proceedings of the International Carnahan Conference on Security Technology (ICCST)*, 2019, pp. 1–8.

[22] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "Ciciot2023: A real-time dataset and benchmark for large-scale attacks in iot environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023.

[23] S. Samarakoon, Y. Siriwardhana, P. Porambage, M. Liyanage, S. Y. Chang, J. Kim, and M. Ylianttila, "5g-nidd: A comprehensive network intrusion detection dataset generated over 5g wireless network," *arXiv preprint arXiv:2212.01298*, 2022.

[24] T. Dasu, S. Krishnan, S. Venkatasubramanian, and K. Yi, "An information-theoretic approach to detecting changes in multidimensional data streams," in *Proceedings of the 38th Symposium on the Interface of Statistics, Computing Science, and Applications*, 2006.