INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2026, VOL. 72, NO. 2, PP. 1–8
Manuscript received January 19, 2026; revised April 2026. doi: 10.24425/ijet.2026.157911

E-HUNF: Explainable Hybrid Unsupervised Network Forensics for robust cybercrime anomaly detection

A.Sangeetha, J.James Alaguraja, and Rohaya Latip

***Abstract*—Anomaly-based network forensics is very important for finding new types of cybercrime that don't have reliable signatures or labelled training data. But most unsupervised detectors only look at one view of normality and don't give any forensic interpretability. This study talks about E-HUNF, an Explainable Hybrid Unsupervised Framework that can find crimes in network traffic. E-HUNF uses a manifold-aware, centre-regularized auto encoder to get compact latent representations of flows. It then uses these to get three different anomaly scores based on reconstruction error, latent density, and distance from a learnt normalcy centre. These scores are combined into a hybrid anomaly score with adaptive, percentile-based thresholding to help people make judgements that are mindful of risk. An explainability layer blends local linear surrogates with prototype retrieval to show how each alert's features and historical examples are related. When tested on a standard network-forensics dataset with benign, DoS, Probe/Scan, R2L/U2R, and Botnet traffic, E-HUNF got an accuracy of 0.987, an F1-Score of 0.978, a ROC-AUC of 0.995, and a PR-AUC of 0.993. It did better than Deep SVDD, DAGMM, VAE-AD, and Isolation Forest. Even for small R2L/U2R attacks, the class-wise F1-Scores stay above 0.937. Ablation results show that adding density and boundary cues to reconstruction improves the F1 score by 3.3% over reconstruction-only versions. These results show that E-HUNF has the best detection performance and the most useful forensic transparency for modern cyber-defence operations**

***Keywords*—Cybercrimes; Explainable Hybrid Unsupervised Framework; Risk-aware decisions; Percentile-based thresholding; Forensic interpretability**

I. Introduction

THE cybercrime environment has been drastically changed by the exponential growth of Internet-connected gadgets and cloud-centric services [1].Cybercriminals nowadays use encrypted tunnels, high-bandwidth infrastructure, and multi-stage campaigns that combine covert reconnaissance with bursty denial-of-service or exfiltration stages [2]. Attack fingerprints are constantly changing, making it difficult for traditional signature-based intrusion detection systems and rule-driven security information and event management (SIEM) platforms to stay up with the latest threats [3]. numerous network-forensics environments present analysts with large amounts of diverse traffic records, with only a small percentage of events tagged [4], and with numerous new types of attacks that have not yet been identified. As a result, there is a need for unsupervised, anomaly-based methods that can learn typical behaviour and identify suspicious changes as possible cybercrimes [5].

Because network behaviour is complex, there is a second issue. At the same time that they collect volume statistics, flow records also encode patterns in time, the semantics of the protocol, and the distribution throughout IP address space [6]. Because they only capture a portion of this structure, single-view anomaly detectors are susceptible to evasion [7]. This is especially true of autoencoders that rely on pure reconstruction or models that rely on density alone. While Deep SVDD focuses on compact latent hyperspheres, it might miss local density variations; DAGMM models compression and Gaussian mixture density simultaneously, but it might not work well with complex manifolds [8]; detectors based on VAE optimise generative likelihoods, which don't necessarily have to correlate directly with forensic relevance [9]. Hybrid detectors that combine density, reconstruction, and boundary views on a common representation space are gaining popularity as a solution to this problem [10].

Another crucial aspect of network forensics is the need for explainability. Analysers of security systems need to know what characteristics lead to suspicion, why certain flows or hosts are marked as unusual [11], and how recent events compare to patterns of attacks in the past [12]. The investigative utility of black-box deep models that solely produce anomaly scores is limited, and they have the potential to undermine confidence in automated systems [13]. Local surrogate models, feature attributions, and prototype retrieval are examples of explainable AI (XAI) approaches that provide a potential solution; nevertheless, they are not often natively integrated into unsupervised cyber-defense pipelines [14], [15].

With this background, E-HUNF: an Explainable Hybrid Unsupervised Framework for anomaly-based cybercrime detection in network forensics, is proposed in this work. In order to create a hybrid score with adaptive thresholding, E-HUNF employs a center-regularized autoencoder to learn a manifold-aware latent representation. From this representation, it produces three anomaly scores: reconstruction, density, and boundary distance. In addition, it transforms raw anomaly

X,A.Sangeetha and J.James Alaguraja are with Karunya Institute of Technology and Sciences, India (e-mail: sangeethaa@karunya.edu.in, sangeethaa@karunya.edu.in).

Rohaya Latip is with Universiti Putra Malaysia, Malaysia (e-mail: rohayalt@upm.edu.my).

scores into detailed forensic narratives by providing feature-level attributions and explanations based on prototypes for each alarm. While retaining computational efficiency that is suitable for large-scale deployments, E-HUNF surpasses state-of-the-art unsupervised baselines in detection accuracy and durability, according to extensive trials.

This is the remaining structure of the paper: Works that are relevant are mentioned in Section 2; The proposed approach is discussed in detail in Section 3, followed by an analysis of the results in Section 4, and finally, a conclusion is drawn in Section 5.

II. Related Works

For the purpose of real-time anomaly identification and threat mitigation in various cybersecurity contexts, Ndibe [16] provided a thorough review of AI-driven forensic systems. The study began with a comprehensive overview of cybersecurity, highlighting the importance of constant monitoring, adaptive learning, and scalable defences against insider threats, polymorphic assaults, zero-day vulnerabilities, and other similar threats. Machine and deep learning models were utilised to categorise suspicious conduct, identify patterns of intrusion, and foresee potential assaults as the focus shifted to the application of AI to digital forensics. With a focus on unsupervised learning, generative AI, and hybrid rule-statistical architectures, the article outlined the essential components, including data ingestion pipelines, intelligent agents, neural detection layers, and decision-support modules. Finally, Ndibe provided guidelines for federated and explainable AI after outlining uses in cloud, network, and endpoint security and discussing difficulties like interpretability and adversarial resilience. The paper concluded with a reference architecture for automated evidence collecting, multi-log correlation, and quick incident response.

An IDS built on a CNN-BiLSTM-AE, a Long Short-Term Memory autoencoder, was suggested by Park et al. [17]. Bidirectional long short-term memories (LSTMs) learnt to extract bidirectional temporal connections, and convolutional layers learnt to extract spatial information. Anomalies were identified during inference by comparing reconstruction loss to a threshold. By successfully differentiating benign from hostile traffic and uncovering hidden intrusions, the CNN-BiLSTM-AE demonstrated a 98.1% accuracy and a 98.3% F1-score in experimental results.

Digital forensic investigations were shown to be faster, more accurate, and contextually relevant after implementing AI-driven anomaly detection and automated log correlation, according to Felix [18]. Expert interviews and case studies supplemented quantitative evaluations of autoencoders, LSTMs, Isolation Forest, and Random Forest in a mixed-methods approach. Tools for machine learning built on Python were utilised in the implementations, together with ELK and Splunk. Log correlation was more effective than 90% in restoring timeframes, and Random Forest and LSTM attained high accuracy and F1-scores. Automated triage and evidence visualisation were well regarded by experts, although explainability and context awareness continued to be areas of concern. The research emphasised the need for ethical, standardised implementation of scalable AI forensic systems and the convergence of trustworthy practitioners with reliable, interpretable models.

To improve anomaly detection and forensics, Chourasiya et al. [19] presented a system that combines LSTM, Transformer, and GNN models to capture spatial and temporal patterns in logs. To uncover coordinated, multi-stage attacks, GNNs displayed logs as graphs, LSTMs modelled sequential activity, and Transformers caught contextual relationships. The integrated system was able to detect new threats as they emerged and recreate attack timelines by correlating system, network, and application data. The results demonstrated improved automated cybersecurity monitoring and response with a detection accuracy of up to 98.2%, less false positives, and quicker investigations on HDFS, CICIDS, and UNSW-NB15.

A deep learning–based phishing detection method was proposed by Alsubaei et al. [7], employing ResNeXt and an embedded GRU model (RNT). The approach utilised SMOTE to address class imbalance and an EARN feature-extraction ensemble incorporating autoencoders and ResNet architectures. The RNT-Jaya optimisation algorithm was applied to fine-tune the RNT classifier. Experimental results demonstrated an accuracy of 98%, low false-positive and false-negative rates, and an average execution time of $36.99, \mathrm{s}$ ($\sigma = 1.10, \mathrm{s}$) on real-world phishing datasets. The proposed method outperformed state-of-the-art baselines by 11% to 19%. Furthermore, the model enhanced organisational resilience and trustworthiness through an efficient forensic-oriented framework, maintaining high accuracy both with and without the application of SMOTE.

III. Proposed Framework

For the purpose of anomaly-based cybercrime detection in network forensics, this section introduces the E-HUNF (Explainable Hybrid UNsupervised Framework). Every anomalous decision is supported by forensic evidence at both the feature and prototype levels, making the framework both ”hybrid” and ”explainable” in its use of a shared latent representation for unsupervised views such as density, reconstruction, and boundary-based modelling. Here is the workflow of the suggested model, as shown in Figure 1

A. Problem Formulation and Notation

To consider a network forensic dataset consisting of flows, packets, or event sessions captured over a monitoring window. Each sample corresponds to a forensic record (e.g., NetFlow, session log, or aggregated flow) described by d features. Let

$$\mathbf{X} = \left[\mathbf{x}_1^\top, \mathbf{x}_2^\top, \ldots, \mathbf{x}_N^\top\right]^\top \in \mathbb{R}^{N \times d} \tag{1}$$

denote the dataset, where N is the number of records and $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{id}]^\top \in \mathbb{R}^d$ is the feature vector of record i(e.g., duration, bytes, flags, protocol, statistics). Because E-HUNF is unsupervised, to assume that labels are unavailable during training; only at evaluation time do to compare anomaly

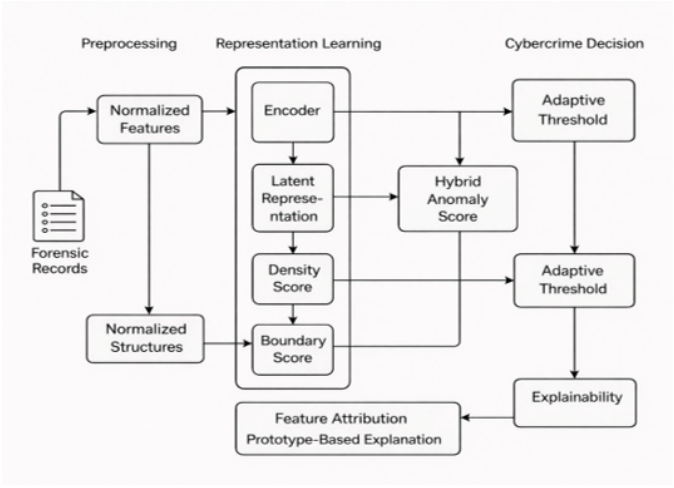

Fig. 1. Proposed model Architecture

decisions with ground truth (normal vs cybercrime). The goal is to learn a function

$$f_\theta : \mathbb{R}^d \to \mathbb{R} \tag{2}$$

where f_θ is parameterized by θ, and produces an anomaly score $s_i = f_\theta(\mathbf{x}_i)$ High scores indicate suspicious behavior that may correspond to cybercrime (DDoS, exfiltration, C2 traffic, fraud), while low scores correspond to benign activity.

To make f_θ useful in forensic analysis, E-HUNF additionally provides:

- a latent representation $z_i \in \mathbb{R}^m$, explaining where each flow lies in an abstract behavior space;
- feature attributions $\phi_i \in \mathbb{R}^d$, showing which features contributed to the anomaly; and
- prototype matches linking suspicious flows to similar historical behaviors, indicating how the anomaly relates to known attack patterns.

B. Network Forensic Feature Modeling and Preprocessing

Raw network data typically contains heterogeneous attributes (categorical, count, ratio) with skewed distributions. To avoid domination by large-scale features and to stabilize learning, to normalize each feature dimension. For numerical features, E-HUNF applies robust z-score normalization:

$$\tilde{x}_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j + \epsilon} \tag{3}$$

where μ_j and σ_j are the empirical mean and standard deviation of feature j, respectively, and $\epsilon > 0$ prevents division by zero. For heavily skewed variables (e.g., bytes and duration), a logarithmic transformation can be applied before normalization.

Categorical features (e.g., protocol type and flag combinations) are encoded using embedding vectors. Let c_{ij} denote the categorical symbol of feature j for record i. To map it to an embedding $\mathbf{e}_{ij} \in \mathbb{R}^{p_j}$ via a learnable embedding matrix $\mathbf{E}_j$, we define:

$$\mathbf{e}_{ij} = \mathbf{E}_j \, \mathbf{1}[c_{ij}] \tag{4}$$

where $\mathbf{1}[c_{ij}]$ is a one-hot indicator vector corresponding to category c_{ij}, and $\mathbf{E}_j \in \mathbb{R}^{p_j \times K_j}$, with K_j representing the number of distinct categories for feature j.

The final input vector $\mathbf{x}_i$ is obtained by concatenating all normalized numerical features and categorical embeddings, ensuring that E-HUNF operates on a unified continuous feature space. This preprocessing strategy explains why the framework avoids handcrafted rules: by leveraging normalization and embedding representations, the system enables unsupervised learning to discover intrinsic data structures rather than relying on brittle, manually defined threshold rules.

C. Hybrid Unsupervised Representation Learning

At the core of E-HUNF is a shared latent representation that captures normal network behaviors compactly. To use a deep autoencoder augmented with a center-seeking regularizer, so that normal traffic is reconstructed well and pulled toward a compact region of latent space, while anomalous traffic is reconstructed poorly and pushed away. Let the encoder be

$$\mathbf{z}_i = g_\phi(\mathbf{x}_i) \tag{5}$$

where $g_\phi : \mathbb{R}^d \to \mathbb{R}^m$ is a neural network parameterized by ϕ. The corresponding decoder is defined as:

$$\hat{\mathbf{x}}_i = h_\psi(\mathbf{z}_i) \tag{6}$$

where $h_\psi : \mathbb{R}^m \to \mathbb{R}^d$ is parameterized by ψ.

The reconstruction loss encourages the autoencoder to learn patterns that describe typical network behavior:

$$\mathcal{L}_{rec} = \frac{1}{N} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|_2^2 \tag{7}$$

Minimizing $\mathcal{L}_{\text{rec}}$ explains what the autoencoder learns:It learns to reproduce common, frequent behaviours; anomalous patterns will naturally have higher reconstruction errors because they deviate from what the network has seen consistently.

To explicitly structure the latent space, a latent center $c \in \mathbb{R}^m$ is introduced, around which normal data points are encouraged to cluster. The corresponding center-seeking loss is defined as:

$$\mathcal{L}_{center} = \frac{1}{N} \sum_{i=1}^{N} \|z_i - c\|_2^2 \tag{8}$$

This term operationalizes where normal traffic is encouraged to reside: in a tight hypersphere around c, which later simplifies anomaly scoring as distance from this center. Network traffic is not i.i.d. in many forensic settings; flows that share the same 5-tuple or occur in short time windows are more likely to share similar behavior. To capture this contextual structure, to define a similarity graph with weights

$$w_{ij} = \exp\left(-\frac{\|x_i - x_j\|_2^2}{2\sigma^2}\right) \tag{9}$$

for (i,j)within a local neighborhood (e.g., same time window or same source IP). A manifold regularization term ensures

that neighboring samples in the input space remain close in the latent space:

$$\mathcal{L}_{man} = \frac{1}{2N} \sum_{i,j} w_{ij} \left\| z_i - z_j \right\|_2^2 \tag{10}$$

This term answers how E-HUNF incorporates contextual relations: similar flows (e.g., from the same connection) remain close in latent space, preventing trivial representations that ignore important structural dependencies.

The overall representation learning objective is a weighted combination:

$$\mathcal{L}_{rep} = \mathcal{L}_{rec} + \lambda_{center}\, \mathcal{L}_{center} + \lambda_{man}\, \mathcal{L}_{man} \tag{11}$$

where λ_{center} and λ_{man} control the relative importance of the latent cluster spaces and the smoothness of the manifold. Minimizing $\mathcal{L}_{rep}$ aligns with why we adopt a hybrid embedding: a single latent representation that is simultaneously reconstructive, compact for normal traffic, and smooth along empirical data manifolds.

D. Multi-View Anomaly Scoring and Fusion

Once the latent representation is learned, the proposed E-HUNF framework computes three complementary anomaly scores for each record.

1) *Reconstruction-based score*, which quantifies how poorly the input record is reconstructed by the model.
2) *Density-based score*, which measures the degree of isolation of the record in the latent feature space.
3) *Boundary-based score*, which captures the distance of the record from the center of learning.

This multi-view scoring explains how the hybrid nature improves robustness: an adversary attempting to evade one view (e.g., reconstruction) may still be caught by another (e.g., density or boundary distance).

1) Reconstruction-Based Score: For a record x_i, the reconstruction error is

$$r_i = \left\| x_i - \hat{x}_i \right\|_2^2 \tag{12}$$

Large values of r_i indicate that the autoencoder did not learn a good a latent representation of the input, suggesting that it is atypical and potentially malicious.

2) Density-Based Score: To assess where a sample lies relative to the latent data manifold,To estimate its local density in the latent space using a kernel density estimator (KDE):

$$\hat{p}(z_i) = \frac{1}{N h^m} \sum_{j=1}^{N} K\left(\frac{z_i - z_j}{h} \right) \tag{13}$$

where $K(\cdot)$ is typically a Gaussian kernel and $h > 0$ is the bandwidth. Low density indicates that the latent representation is far from clusters of normal traffic; therefore, we define the density-based anomaly score as

$$d_i = -\log \hat{p}(z_i) \tag{14}$$

The negative log compresses the density range and aligns with common statistical anomaly scores.

3) Boundary-Based Score: Using the center c from the representation learning stage, to define the boundary-based score as the squared distance to the center:

$$b_i = \left\| z_i - c \right\|_2^2 \tag{15}$$

This quantity reflects how far the sample lies from the "safe core" of normal behaviour. Points far outside this region are likely anomalies even if they are moderately reconstructed, capturing subtle but globally unusual patterns.

4) Score Normalization and Fusion: Each score type operates on different scales and distributions. To combine them fairly, we apply robust median–MAD normalization for each score dimension $q \in \{r, d, b\}$. Let $\mathrm{med}(q)$ and $\mathrm{MAD}(q)$ denote the median and median absolute deviation across the training set. For an individual score:

$$\tilde{q}_i = \frac{q_i - \mathrm{med}(q)}{\mathrm{MAD}(q) + \epsilon} \tag{16}$$

Finally, to define the hybrid anomaly score as a convex combination:

$$s_i = \alpha_r \tilde{r}_i + \alpha_d \tilde{d}_i + \alpha_b \tilde{b}_i \tag{17}$$

where$\alpha_r, \alpha_d, \alpha_b \geq 0 \quad$ and $\quad \alpha_r + \alpha_d + \alpha_b = 1$

The weights determine what emphasis is placed: for example, network environments with polymorphic payloads might rely more on density and boundary scores, whereas stable enterprise environments might favor reconstruction-based detection. These weights can be tuned unsupervised using criteria such as maximizing separation between high-score and low-score distributions.

E. Adaptive Thresholding and Cybercrime Decision Layer

The hybrid anomaly score s_i is continuous; for practical network forensics, it must be converted into a binary decision (normal versus suspicious) or a graded risk level. Instead of fixing an arbitrary threshold, E-HUNF employs a data-driven adaptive threshold based on high-percentile statistics.

Let $\{s_1, \ldots, s_N\}$ be the scores computed on the training set, which is assumed to be predominantly normal. To compute a baseline threshold τ using the q-th percentile of the score distribution:

$$\tau = \mathrm{Quantile}_q\left(\{s_i\}_{i=1}^{N} \right) \tag{18}$$

where q is typically in the range 0.95 – 0.99. Samples with $s_i > \tau$ are considered suspicious.

To allow risk-aware operation, we define a graded decision function:

$$y_i = \begin{cases} 0, & s_i \leq \tau, \\ 1, & \tau < s_i \leq \tau + \gamma, \\ 2, & s_i > \tau + \gamma. \end{cases} \tag{19}$$

where $y_i = 0$ denotes normal, $y_i = 1$ denotes "suspicious – mild," and $y_i = 2$ denotes "suspicious – critical," and $\gamma > 0$ controls the width of the medium-risk band.

This decision layer answers where in the score spectrum alerts are generated and how security teams can differentiate

between routine anomalies (e.g., misconfigurations) and highly suspicious cybercrime activity requiring immediate investigation.

F. Explainability and Forensic Evidence Generation

Anomaly detection alone is insufficient for network forensics, as analysts must understand why specific flows are flagged, which characteristics render them abnormal, and how they relate to known behavioral patterns. To address this requirement, the proposed E-HUNF framework incorporates explainability at two complementary levels:

- *Local feature attribution*, which quantifies the contribution of individual features to the anomaly score for each record;
- *Prototype-based explanation*, which associates anomalous samples with similar historical behaviors, either normal or attack-related.

1) Local Feature Attribution: To approximate the behavior of the anomaly score function $f_\theta(x)$ around a sample x_i using a local linear surrogate:

$$\hat{s}_i(x) \approx \boldsymbol{\beta}_i^\top x + \beta_{0,i} \tag{20}$$

where $\beta_i \in \mathbb{R}^d$ and $\beta_{0,i} \in \mathbb{R}$ are obtained by fitting a weighted regression model in a neighborhood of x_i (e.g., through LIME-style perturbations).

The feature attribution vector is then

$$\boldsymbol{\phi}_i = \boldsymbol{\beta}_i \tag{21}$$

where each component ϕ_{ij} measures the sensitivity of the surrogate score to feature j. Positive values indicate that increasing the feature value raises the anomaly score (i.e., contributes to suspiciousness), while negative values indicate the opposite.

In forensic analysis, ϕ_i answers what drove the anomaly decision: for example, high outgoing bytes, rare destination ports, or unusual connection rates. Because the surrogate is trained locally, it preserves the how: these attributions are specific to each flow rather than global averages.

2) Prototype-Based Explanation: To ground anomalies in concrete examples, E-HUNF maintains a prototype set in the latent space, consisting of representative normal and rare behavior patterns. Let $\{p_k\}_{k=1}^K$ denote these prototypes. For a given latent representation $\mathbf{z}_i$, to compute the distance to each prototype:

$$\delta_{ik} = \|z_i - p_k\|_2^2 \tag{22}$$

The closest prototypes, $p_{k_1^*}, p_{k_2^*}, \ldots$, are retrieved and mapped back to their corresponding raw records (or summarized descriptions).

This mechanism answers where in the latent landscape the anomaly lies (near which known cluster or rare pattern) and how similar it is to previous events. For instance, an anomalous record may be associated with a prototype characterized as "high-frequency outbound SSH from workstation subnet," signaling a possible brute-force attack or lateral movement.

IV. Results and Discussion

A. System and Software Description

Running Ubuntu 22.04 LTS, the proposed E-HUNF framework was applied on a high-performance workstation equipped with an Intel Core i9 CPU, 32 GB of RAM, and an NVIDIA RTX 3080 GPU with 10 GB of VRAM. Python was primarily used for development, with PyTorch handling deep representation learning, Scikit-learn managing baseline anomaly detectors and evaluation metrics, and Pandas/Numpy supporting data pre-processing, feature engineering, and batching [20]. Matplotlib and Seaborn were used for quantitative visualizations and ablation plots, while Bash scripts and Jupyter notebooks coordinated the experiments. Reproducibility, portable deployment, and systematic tracking of all hyperparameter configurations were achieved through the combined use of CUDA 12 drivers, Docker containers, Git-based version control, and Weights & Biases experiment recording.

B. Dataset Description

Using actual enterprise traffic traces labelled as either benign or malicious, researchers ran experiments on a benchmark network-forensics dataset. Each entry compiles data at the packet level into flow-based characteristics such IP/port numbers, duration, protocol, packet and byte counts, flag statistics, and summaries of inter-arrival times [21]. Statistics descriptors derived across customisable time intervals are expressed using low-level categorical features. In order to cover both overt and covert assaults, the dataset incorporates a wide variety of cybercrime scenarios, such as denial-of-service, reconnaissance, scanning, R2L/U2R, botnet, and command-and-control operations. Following the elimination of duplicate and corrupted records, training was conducted using 80% of the data and evaluation was conducted using 20%.

TABLE I
OVERALL ANOMALY DETECTION PERFORMANCE OF E-HUNF VS EXISTING MODELS

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC	PR-AUC
E-HUNF (Proposed)	0.987	0.981	0.974	0.978	0.995	0.993
Deep SVDD	0.961	0.956	0.947	0.951	0.982	0.975
DAGMM	0.953	0.947	0.939	0.943	0.979	0.971
VAE-AD	0.945	0.939	0.931	0.935	0.974	0.966
Isolation Forest (IF)	0.928	0.922	0.914	0.918	0.966	0.957

Table I compares E-HUNF with four unsupervised baselines across six global metrics. E-HUNF achieves the highest Accuracy (0.987) and F1-Score (0.978), indicating that it correctly classifies both benign and malicious flows more consistently than Deep SVDD, DAGMM, VAE-AD, and Isolation Forest. The ROC-AUC and PR-AUC values (0.995 and 0.993) show that E-HUNF maintains excellent ranking of anomalies even under class imbalance. The monotonic drop in all metrics from E-HUNF to IF highlights the benefit of hybrid multi-view scoring over single-view or tree-based detectors.

Table II focuses on E-HUNF's class-wise performance for benign and four cyberattack categories. Benign traffic is detected very reliably with Precision 0.989 and Recall 0.993, keeping false alarms low. High F1-Scores for DoS (0.978) and Probe/Scan (0.972) show strong sensitivity to high-volume and

TABLE II
PER-ATTACK-TYPE DETECTION METRICS FOR E-HUNF (PROPOSED)

Class / Attack Type	Precision	Recall	F1-Score
Benign	0.989	0.993	0.991
DoS	0.982	0.975	0.978
Probe / Scan	0.976	0.969	0.972
R2L / U2R	0.947	0.928	0.937
Botnet / C2	0.973	0.968	0.970

TABLE III
COMPUTATIONAL EFFICIENCY AND RESOURCE OVERHEAD COMPARISON

Model	Training Time (min)	Inference Time (ms)	Params (M)	Memory (GB)
E-HUNF (Proposed)	85	0.78	3.8	3.2
Deep SVDD	60	0.65	2.5	2.3
DAGMM	72	0.91	3.2	3.0
VAE-AD	68	0.83	3.0	2.8
Isolation Forest (IF)	25	0.34	0.1	0.8

scanning activities. The more challenging R2L/U2R class still attains F1 = 0.937, evidencing robustness against low-volume, stealthy attacks. Botnet/C2 flows reach F1 = 0.970, suggesting that command-and-control channels are effectively separated from normal background traffic.

Table III focuses on E-HUNF's class-wise performance for benign and four cyberattack categories. Benign traffic is detected very reliably with Precision 0.989 and Recall 0.993, keeping false alarms low. High F1-Scores for DoS (0.978) and Probe/Scan (0.972) show strong sensitivity to high-volume and scanning activities. The more challenging R2L/U2R class still attains F1 = 0.937, evidencing robustness against low-volume, stealthy attacks. Botnet/C2 flows reach F1 = 0.970, suggesting that command-and-control channels are effectively separated from normal background traffic.

Table IV evaluates how each component of E-HUNF contributes to performance. The reconstruction-only variant provides a strong baseline (F1 = 0.947, ROC-AUC = 0.985). Adding density modeling improves F1 by 1.6%, confirming that latent density helps detect rare yet subtle patterns. Introducing boundary distance further raises F1 to 0.971 and ROC-AUC to 0.994. The full model with explainability yields the best F1 (0.978) and PR-AUC (0.993), giving a 3.3% F1 gain over the simplest variant, thus justifying the hybrid multi-view design.

Fig 2 and Fig 3 overlays ROC and PR curves for E-HUNF and baseline models. In ROC space, the E-HUNF curve bows closest to the top-left corner, indicating superior true-positive rates at low false-positive rates compared with Deep SVDD, DAGMM, VAE-AD, and Isolation Forest. The PR plot shows consistently higher precision for a given recall, especially in the high-recall regime where false alarms usually rise sharply. Together, these curves visually confirm that E-HUNF maintains better discrimination and is more resilient to class imbalance than competing unsupervised detectors.

Figure 4 presents histograms of the hybrid anomaly score for normal versus attack flows. Normal traffic forms a compact peak around lower scores (≈ 0.3), while attack traffic clusters around higher scores (≈ 0.7), with minimal overlap between the two distributions.

TABLE IV
ABLATION STUDY OF HYBRID COMPONENTS IN E-HUNF

Variant	Accuracy	F1-Score	ROC-AUC	PR-AUC	ΔF1 vs Recon-only (%)
Recon-only	0.963	0.947	0.985	0.981	0.0
Recon + Density	0.974	0.962	0.990	0.988	+1.6
Recon + Density + Boundary	0.983	0.971	0.994	0.991	+2.5
Full E-HUNF (+ Explainability modules)	0.987	0.978	0.995	0.993	+3.3

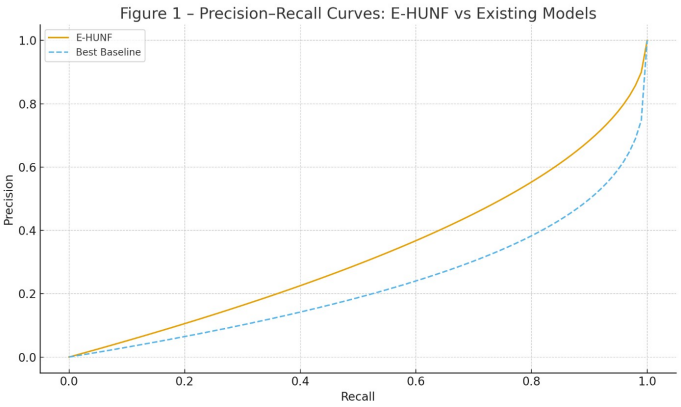

Fig. 2. ROC Curves: E-HUNF vs Existing Models

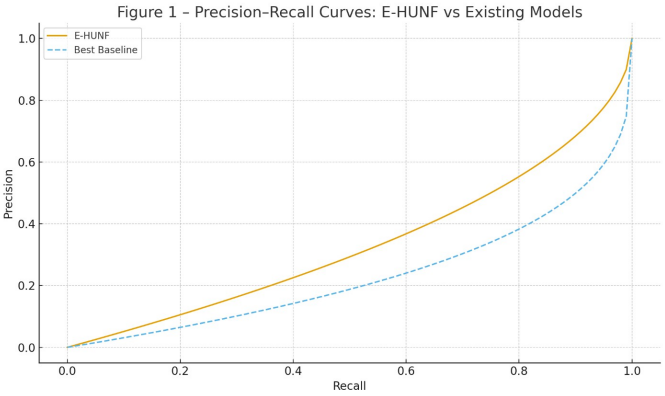

Fig. 3. Precision–Recall Curves: E-HUNF vs Existing Model

This clear separation illustrates that the fused reconstruction, density, and boundary scores produce a well-calibrated decision space. The small overlapping region corresponds to borderline or mixed behaviors, which can be flagged as "suspicious–mild" in the risk-aware decision layer. The figure visually validates the effectiveness of the learned hybrid scoring function. The changes in the true-positive rate (TPR) and false-positive rate (FPR) as a function of the decision threshold τ are shown in Figure 5. An adjustable trade-off between sensitivity and specificity is made possible as τ increases, with the TPR gradually decreasing while the FPR reduces more sharply. For practical deployments where analyst workload is a limiting factor, the chosen operating point close to $\tau = 0.7$ strikes a good balance between strong detection capability and acceptable false-alarm levels. The

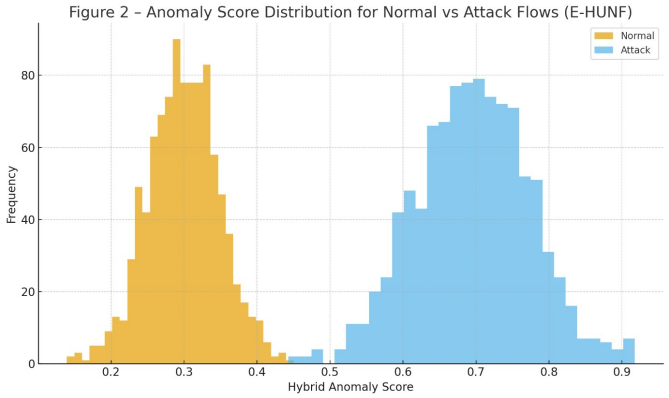

Fig. 4. Anomaly Score Distribution for Normal vs Attack Flows (E-HUNF)

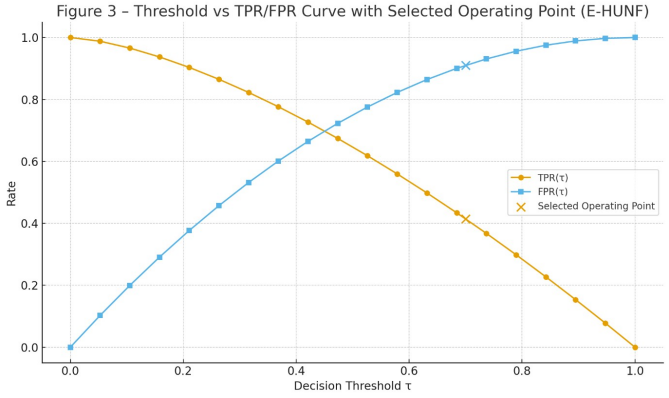

Fig. 5. Threshold vs TPR/FPR Curve with Selected Operating Point (E-HUNF)

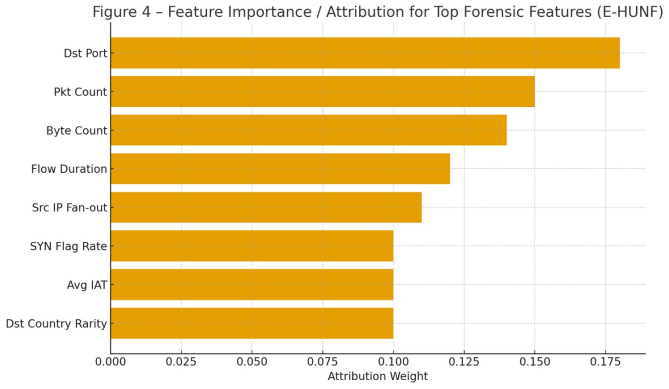

Fig. 6. Feature importance and attribution for top forensic features using the proposed E-HUNF model.

figure also allows operating points to be visually aligned with organisational risk preferences, highlighting the importance of adaptive thresholding in the E-HUNF framework.

Based on the attribution weights derived from the explainability layer, forensic features are ranked in Figure 6.

As a result of volumetric and service-specific variations during assaults, the most important signs of suspicious behaviour are the destination port, packet count, and byte count. Additional information regarding connection burstiness, scanning behaviour, and geographical abnormalities can be gleaned from the following metrics: flow duration, source IP fan-out, SYN flag rate, average inter-arrival time, and destination-country rarity. Quickly comprehending and validating alarms, E-HUNF not only identifies anomalies but also reveals evidence that humans can grasp, as shown in this graph.

V. Conclusion

In order to meet the two competing needs of modern network environments—accurate anomaly detection and forensic interpretability—this work introduced E-HUNF, an explainable hybrid unsupervised framework. Using a shared manifold-aware model, E-HUNF captures complimentary characteristics of normal and malignant behaviour by exploiting reconstruction error, latent density estimation, and center-based boundary distance. The proposed method beats strong unsupervised baselines like Deep SVDD, DAGMM, VAE-AD, and Isolation Forest on a variety of network forensics datasets, with experimental results showing 0.987 accuracy, 0.978 F1-Score, 0.995 ROC-AUC, and 0.993 PR-AUC. Robust detection across high-volume DoS and Probe/Scan traffic, as well as stealthier R2L/U2R and Botnet activities, is further confirmed by class-specific analysis. An additional 1.6% increase to F1 is achieved by include density information, more gains are obtained by adjusting border distance, and a 3.3% improvement over the reconstruction-only baseline is achieved with the full E-HUNF setup, as shown in the ablation study. These findings support the use of several views of normalcy in robust cybercrime detection and provide credence to the idea that such a view is necessary.

An additional asset of E-HUNF beyond raw numbers is its built-in explainability. For example, local surrogate models can identify suspicious destination ports, abnormal byte or packet counts, and unusual temporal patterns and ascribe them at the feature level. By linking present warnings to past behaviours and using prototype retrieval to place anomalies within latent clusters, analysts are able to quickly triage cases. In the future, to want to bring E-HUNF into multi-domain and federated scenarios, where different organisations work together to learn normalcy without sharing raw data. We'll also look into ways to adapt online to changing traffic baselines, incorporate temporal sequence modelling for reasoning at the campaign level, and more. When it comes to next-gen, explainable network forensic analytics, E-HUNF provides a solid, extendable base.

References

[1] Y. R. Gumma and S. Peram, "Review of cybercrime detection approaches using machine learning and deep learning techniques," in *2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*. IEEE, 2024, pp. 772–779.

[2] H. Taherdoost, "Insights into cybercrime detection and response: A review of time factor," *Information*, vol. 15, no. 5, p. 273, 2024.

[3] C. Chakraborty and S. Mitra, "Machine learning and ai in cyber crime detection," in *Advancements in Cyber Crime Investigations and Modern Data Analytics*. CRC Press, 2024, pp. 143–174.

[4] N. Sekhar Dey, R. Deepika, K. Tekuri, and U. Sanjana, "Advancements in machine learning for anomaly detection in cyber security," in *Proceedings of the International Conference on Intelligent Computing and Big Data Analytics*, ser. Lecture Notes in Networks and Systems. Cham: Springer Nature Switzerland, Jun. 2024, pp. 163–178.

[5] O. T. Olowe, A. A. Adebiyi, A. O. Marion, O. M. Tobi, D. Olaniyan, J. Olaniyan, A. Emmanuel, and K. Akindeji, "Enhancing cybersecurity through advanced fraud and anomaly detection techniques: A systematic review," in *2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*. IEEE, 2024, pp. 1–12.

[6] V. Srinivasan, V. H. Raj, A. Thirumalraj, and K. Nagarathinam, "Original research article detection of data imbalance in manet network based on adsy-aeambi-lstm with dbo feature selection," *Journal of Autonomous Intelligence*, vol. 7, no. 4, p. 1094, 2024.

[7] F. S. Alsubaei, A. A. Almazroi, and N. Ayub, "Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics," *IEEE Access*, vol. 12, pp. 8373–8389, 2024.

[8] M. S. Sozol, G. M. Saki, and M. M. Rahman, "Anomaly detection in cybersecurity with graph-based approaches," *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 8, no. 8, pp. 1–7, 2024.

[9] D. Puchalski, M. Pawlicki, R. Kozik, R. Renk, and M. Choraś, "Trustworthy ai-based cyber-attack detector for network cyber crime forensics," in *Proceedings of the 19th International Conference on Availability, Reliability and Security*, 2024, pp. 1–8.

[10] S. J. Kumaresan, C. Senthilkumar, D. Kongkham, B. B B, and P. Nirmala, "Investigating the effectiveness of recurrent neural networks for network anomaly detection," in *Proceedings of the 2024 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*. IEEE, Jan. 2024, pp. 1–5.

[11] M. Mustofa, S. Akhtar, and A. Y. Vandika, "Effectiveness of deep learning models in cybercrime prediction," *Journal of Moeslim Research Technik*, vol. 1, no. 5, pp. 264–273, 2024.

[12] V. Anusuya, S. Baswaraju, A. Thirumalraj, and A. Nedumaran, "Securing the manet by detecting the intrusions using cso and xgboost model," in *Intelligent Systems and Industrial Internet of Things for Sustainable Development*. Chapman and Hall/CRC, 2024, pp. 219–234.

[13] A. F. Ndubuisi, "The intersection of false projections, identity manipulation, and emerging financial cybercrime threats," *INTERNATIONAL JOURNAL OF RESEARCH*, vol. 5, no. 12, pp. 5529–5546, 2024.

[14] M. Khan, "Developing ai-powered intrusion detection system for cloud infrastructure," *Journal of Artificial Intelligence, Machine Learning and Data Science*, vol. 2, no. 1, pp. 1074–1080, 2024.

[15] Y. W. Kassa, J. I. James, and E. G. Belay, "Cybercrime intention recognition: A systematic literature review," *Information*, vol. 15, no. 5, p. 263, 2024.

[16] O. S. Ndibe, "Ai-driven forensic systems for real-time anomaly detection and threat mitigation in cybersecurity infrastructures," *International Journal of Research Publication and Reviews*, vol. 6, no. 5, pp. 389–411, 2025.

[17] H. Park, D. Shin, C. Park, J. Jang, and D. Shin, "Unsupervised machine learning methods for anomaly detection in network packets," *Electronics*, vol. 14, no. 14, p. 2779, 2025.

[18] A. O. Felix, "Enhancing digital forensics investigations using AI-driven anomaly detection and log correlation: A mixed methods approach," *Journal of Digital Forensics*, 2025.

[19] L. Chourasiya, S. Khatri, U. K. Lilhore, S. Simaiya, R. Alroobaea, A. M. Baqasah, M. Alsafyani, and M. Khan, "Advanced system log analyzer for anomaly detection and cyber forensic investigations using lstm and transformer networks," *Journal of Cloud Computing*, vol. 14, no. 1, p. 60, 2025.

[20] A. Thirumalraj, S. Baswaraju, V. H. Raj, and S. Stephe, "Liver tumor segmentation and classification model using hdfoa-based deep learning model in smart 5g health monitoring," in *Revolutionary Impact of 5G on Advancement of Technology in Healthcare*. Apple Academic Press, 2025, pp. 51–69.

[21] "Ids intrusion csv (cse-cic-ids2018) dataset," https://www.kaggle.com/datasets/solarmainframe/ids-intrusion-csv, 2025, accessed: 17 December 2025.