CoLD: Collaborative Label Denoising Framework for Network Intrusion Detection

Shuo Yang†, Xinran Zheng‡, Jinze Li†, Jinfeng Xu† and Edith C. H. Ngai†*
† The University of Hong Kong, Hong Kong SAR, China
‡ University College London, London, United Kingdom

***Abstract*—Label noise presents a significant challenge in network intrusion detection, leading to erroneous classifications and decreased detection accuracy. Existing methods for handling noisy labels often lack deep insight into network traffic and blindly reconstruct the label distribution to filter samples with noisy labels, resulting in sub-optimal performance. In this paper, we reveal the impact of noisy labels on intrusion detection models from the perspective of causal associations, attributing performance degradation to local consistency of features across categories in network traffic. Motivated by this, we propose CoLD, a Collaborative Label Denoising framework for network intrusion detection. CoLD partitions the original feature set into multiple subsets and employs Local Joint Learning to disrupt local consistency, compelling the encoder to learn fine-grained and robust representations. It further applies Causal Collaborative Denoising to detect and filter noisy labels by analyzing causal divergences between multiple representations and their potentially true label, yielding a purified dataset for training a noise-resilient classifier. Experiments on several benchmark datasets demonstrate that CoLD effectively improves classification performance and robustness to label noise, highlighting its potential for enhancing network intrusion detection systems in noisy environments.**

I. Introduction

In the field of network security, Intrusion Detection Systems (IDS) are critical for identifying and mitigating malicious activities [1]. As network traffic continues to grow in both complexity and volume [2], [3], the need for accurate and reliable IDS becomes increasingly urgent [4], [5], [6]. Modern IDSs predominantly rely on data-driven models trained on labeled data [7], [8], [9], where high-quality labels are essential for learning effective representations of network behavior. However, in real-world environments, obtaining clean labels is always challenging [10], [11], and the performance of IDS models is severely constrained by the quality of training data.

As illustrated in Fig. 1, an IDS safeguards trusted internal networks and operates in conjunction with firewalls to defend against threats originating from untrusted external sources. The user interface allows administrators to monitor system performance and manage alerts. The general IDS pipeline includes a feature extractor, a labeling module, and a data-driven model used during both training and detection phases. In this pipeline, the labeling process is fundamental to dataset quality. However, it is prone to noise due to human bias, labeling error, and the dynamic nature of evolving network environments [12], [13]. For example, a full attack sequence may initially resemble benign activity [14], as adversaries often use low-rate stealth attacks or encryption techniques [15], [16] to mimic normal patterns. Advanced malware, including polymorphic and metamorphic variants [17], [18], further blurs the distinction between benign and malicious behaviors. As a result, network traffic flows are frequently mislabeled. IDS models trained on such noisy datasets tend to perform poorly, generating false positives for benign actions and failing to detect critical threats [11]. This undermines system reliability and imposes a heavy burden on security teams [19], [20].

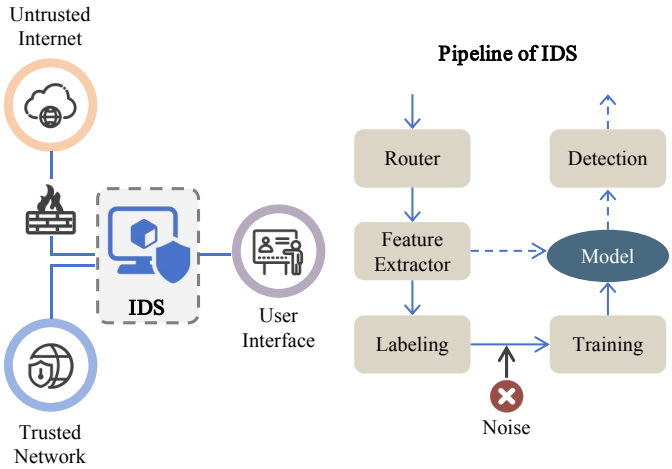


Fig. 1: The overview of IDS with data-driven models in noisy environments.

Existing approaches for handling noisy labels generally fall into two categories: robust training and dataset purification. Robust training methods attempt to make models resilient to noise by modifying loss functions [21], [22], [23], [24] or adjusting training strategies [25], [26], [27], [28]. However, they often rely on assumptions such as prior knowledge of label reliability [27], [28] or access to clean validation data [24], which are often unrealistic in network intrusion detection tasks. In contrast, dataset purification seeks to di-

* Corresponding Author. chngai@eee.hku.hk, shuoyang.ee@gmail.com.

Network and Distributed System Security (NDSS) Symposium 2026
23-27 February 2026, San Diego, CA, USA
ISBN 979-8-9919276-8-0
https://dx.doi.org/10.14722/ndss.2026.231950
www.ndss-symposium.org

rectly detect and correct mislabeled instances. Approaches based on metric learning [13], [29], [30], [31], [32] and active learning [33] have made progress in reducing label noise. Nevertheless, many of these approaches rely on distance-based measurements, which struggle to differentiate between clean and noisy samples in the presence of local consistency. Local consistency refers to the phenomenon where features from different categories share similar distributions in the feature space. In such cases, mislabeled and correctly labeled samples may appear deceptively similar. This may result in the removal of underrepresented or ambiguous samples, leading to the loss of valuable information. In addition, the mechanisms by which label noise influences model learning in network traffic remain poorly understood, hindering the development of more effective solutions.

To address these challenges, we introduce CoLD, a **Co**llaborative **L**abel **D**enoising framework for network intrusion detection. CoLD is built upon an in-depth investigation into noisy label learning on traffic datasets, revealing that performance degradation is largely driven by spurious associations induced by local consistency across traffic categories. Specifically, local features from different categories often share similar distributions, making it difficult for models to distinguish between them. While a high-performing IDS model should extract distinct representations of traffic categories in latent space, noisy labels disrupt this process by introducing spurious associations between erroneous labels and features [11]. This misguidance forces the model to learn naive patterns from locally consistent features, ultimately blurring true decision boundaries and impairing detection accuracy.

The proposed CoLD comprises three main components: Feature Reordering, Local Joint Learning, and Causal Collaborative Denoising. Feature Reordering enhances semantic relevance by rearranging the feature set based on Pearson correlation coefficients. Local Joint Learning partitions the reordered features into multiple subsets and applies self-supervised training to disrupt local consistency, encouraging the model to learn fine-grained and robust representations. Causal Collaborative Denoising purifies the dataset by using a Gaussian Mixture Model (GMM) to analyze the divergence of causal associations between multiple representations and their potential true labels. This process identifies and filters noisy samples, resulting in a purified dataset for training a noise-resilient classifier.

In summary, this paper makes the following contributions:

- We investigate the influence of noisy labels from the perspective of causal associations and attribute it to the local consistency across categories in network traffic.
- We propose CoLD, a collaborative label denoising framework that integrates self-supervised representation learning and causal inference to effectively identify and filter noisy labels, enabling a purified dataset for training a noise-resilient classifier.
- We conduct extensive experiments on benchmark datasets to evaluate the effectiveness of CoLD. The results show that CoLD significantly improves classification performance and robustness to label noise, outperforming existing baselines and state-of-the-art methods.
- We demonstrate the effectiveness of CoLD by evaluating it in realistic enterprise networks. The results indicate that CoLD can operate in challenging environments and enhance the performance of intrusion detection systems.

The remainder of this paper is organized as follows: Section II reviews related work on noisy label learning and robust intrusion detection. Section III analyzes the impact of noisy labels on detection performance from a causal perspective and outlines our motivations. Section IV describes CoLD in detail. Section V presents comparative experiments on benchmark datasets against state-of-the-art methods. Section VI reports the initial deployment and evaluation of CoLD in realistic enterprise networks. Section VII discusses the findings and future directions. Finally, Section VIII concludes the paper.

II. Related Work

A. Noisy Label Learning

Noisy label learning methods related to our work can be broadly categorized into two groups: robust training and dataset purification. Robust training methods aim to mitigate the impact of incorrectly labeled training data on model performance. Some works design robust loss functions [21], [22], [23], [34] and training strategies [25], [26], [27], [28], [35], [36], [37] to ensure performance despite mislabeled samples. Other methods apply the label transition matrix [38], [39], [40], which records the probability of a category being mislabeled into another category, to rectify loss values impacted by noisy labels. These methods generally rely on specific assumptions that are difficult to meet in network intrusion detection. Dataset purification methods aim to identify and correct mislabeled samples during training. The widely adopted option is the training loss, assessing the disparity between the model prediction and given labels [18], [41], [42], [43], with higher loss indicating incorrect labels. In addition, some methods design measurement functions [31], [44], [45], [46] to distinguish between clean and noisy labels. Although these methods have proven effective in image recognition or natural language processing, they are challenging to generalize to network intrusion detection tasks due to a lack of deep insight into network traffic.

B. Robust Intrusion Detection

Advances in deep learning have significantly improved the capabilities of intrusion detection systems [47], [48], [49]. These models are capable of learning complex representations from raw traffic data due to their strong capacity for pattern fitting. However, most existing research has been conducted in closed-world settings, which limits the generalization ability of these models in diverse or evolving real-world environments [50], [51]. This constraint results in decreased performance when such models are deployed in practical scenarios [33]. To overcome this limitation, recent studies have explored the development of more robust intrusion detection methods. For example, Diallo *et al.* [5] proposed supervised

adaptive clustering techniques to learn cluster centers that improve robustness against outliers and enhance generalization. Yue *et al.* [6] applied data augmentation and contrastive learning to extract semantic relationships between samples, further strengthening model robustness. Other works [52], [53], [54] developed adaptive intrusion detection techniques to address the challenge of concept drift, which is caused by evolving attack patterns.

Nevertheless, the success of intrusion detection models is closely tied to the quality of the training data, and label noise still poses a major challenge. Qing *et al.* [55] utilized distribution differences between benign and malicious traffic to estimate potential labels, but their method is limited to binary classification and assumes a balanced dataset. Zhao *et al.* [33] designed an effective online anomaly detection framework that relies on security experts to relabel samples based on uncertainties predicted by quality and classification models, at the cost of increased manual labor. Wu *et al.* [18] introduced a semi-supervised learning framework that integrates robust training by leveraging early-epoch loss magnitude to distinguish clean from noisy labels. They identified noisy samples and treated them as unlabeled data used for representation learning. While they demonstrated the effectiveness on malware classification tasks, their method's reliance on a predefined label splitting ratio constrains its flexibility and limits its applicability across varying noise levels. Yuan *et al.* [13] combined data cleaning and robust training by approximating an ideal representation function. However, their approach relies on clear boundaries between noisy and clean samples. In high-noise scenarios, the boundaries are obscured due to local consistency, where features from different classes exhibit similar distributions. As a result, underrepresented clean samples may be discarded, leading to information loss and suboptimal performance.

These limitations underscore that label noise remains a critical barrier to building reliable IDSs. We identify two core challenges: **C1**. While it is well-established that noisy labels negatively impact data-driven models, the underlying mechanism of how noisy labels affect learning in network traffic remains poorly understood. **C2**. Existing methods are either not suitable for network intrusion detection or their ability to purify datasets is insufficient to cope with more challenging label noise environments. In this work, we address both challenges. For **C1**, we investigate the root causes of performance degradation from a causal association perspective, revealing that local consistency promotes spurious associations between features and noise labels. This insight is supported by theoretical analysis and empirical validation. For **C2**, we propose a Causal Collaborative Denoising method. It employs Local Joint Learning to generate fine-grained representations that disrupt local consistency, and leverages causal collaborative inference to effectively identify and remove noisy labels while preserving underrepresented clean samples.

In summary, CoLD differs from existing methods by leveraging a causal perspective to address label noise, targeting local consistency in network traffic features that often mislead traditional models. Unlike distance- or confidence-based approaches, CoLD employs self-supervised multi-view representation learning and causal divergence analysis to identify and filter noisy labels, achieving superior performance in high-noise environments.

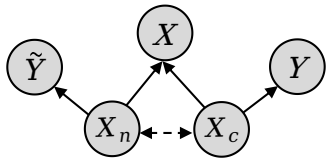


Fig. 2: Causal Graph via SGM. Each traffic flow X comprises causal features X_c and non-causal features X_n. Note that only the causal feature X_c determines the ground truth label Y.

III. Motivating CoLD: Causal Analysis and Local Consistency

In this section, we introduce the motivation behind the CoLD, emphasizing that it arises from a causal analysis of noisy label learning and key observations on network traffic data. Our analysis reveals how noisy labels introduce spurious causal associations and mislead the model's learning through local consistency, thereby providing both theoretical insights and practical foundations for our approach.

A. Noisy Labels in a Causal View

Understanding the root cause of performance degradation caused by noisy labels is fundamental to designing targeted solutions and improving model robustness. To this end, we employ the Structural Causal Model (SCM) to delineate the interactions between features and labels, thus constructing a detailed causal graph. Causal modeling provides an effective framework for representing path dependencies, as the training of classification models fundamentally relies on capturing these underlying causal relationships [56], [57]. Fig. 2 illustrates this interaction with five variables: the input traffic sample X, causal features X_c, non-causal features X_n, the ground-truth label Y, and the observed noisy label $\widetilde{Y}$. Here, causal and non-causal features X_c and X_n are distinct yet partial subsets of X. Whether a feature is causal or non-causal depends on the specific label (Y or $\widetilde{Y}$), meaning X_n could be causal for $\widetilde{Y}$ but non-causal for Y. X_n can be regarded as a feature subset of X that exhibits similar distributions across different categories, potentially establishing opposite causal associations with Y and $\widetilde{Y}$. Below, we explain the causal graph in detail:

- $X_c \rightarrow X \leftarrow X_n$: The input X comprises two subsets of features: causal features X_c and non-causal features X_n, which may overlap but are not completely coincident.
- $X_c \rightarrow Y$: From a causal view, the ground-truth label Y is determined solely by causal features X_c.
- $X_n \rightarrow \widetilde{Y}$: While X_n is non-causal features of Y, it can directly influence the noisy label $\widetilde{Y}$.

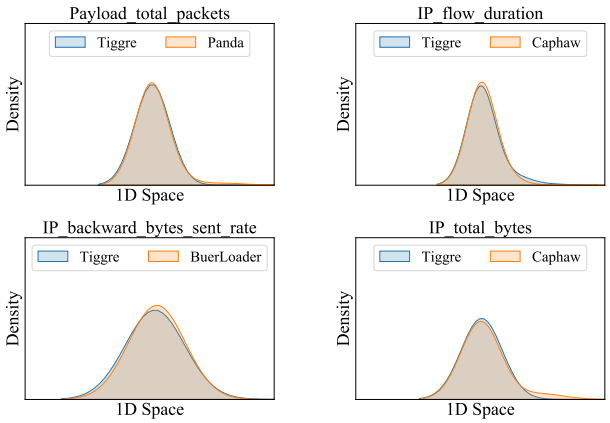


Fig. 3: Examples for Local Consistency. Tiggre, Panda, Caphaw, and BuerLoader are different types of traffic.

- $X_c \dashleftarrow\dashrightarrow X_n$: Dashed arrows represent additional dependencies between causal features X_c and non-causal features X_n, offering insights into how non-causal features might indirectly affect Y through complex pathways.

The causal architecture defines that X_c directly influences Y, establishing a clear causal pathway. However, in general learning tasks with the ground-truth label Y, $X_n \dashleftarrow\dashrightarrow X_c \rightarrow Y$ can create spurious associations between non-causal features X_n and the ground-truth label Y. This makes X_c a confounder between X_n and Y, opening a backdoor path $X_n \dashrightarrow X_c \rightarrow Y$[58]. Bias introduced through such backdoor path is a primary cause of degraded model performance. Additionally, with the involvement of noisy label $\widetilde{Y}$, the bias is further amplified. Although X_n is non-causal for Y, it becomes increasingly relevant to $\widetilde{Y}$. This shift in causal relevance underscores the flexible nature of feature relevance depending on whether the given label is ground truth (Y) or noisy ($\widetilde{Y}$). The noisy label $\widetilde{Y}$ compels the model to focus on associations involving X_n, distorting the causal pathway. Specifically, the causal association between X_c and Y is ignored, while a shortcut $X_c \dashrightarrow X_n \rightarrow \widetilde{Y}$ is activated. As a result, X_c, which was originally causal for Y, establishes a spurious causal relationship with $\widetilde{Y}$.

In scenarios where Y is correctly identified, the causal influence of X_c is predominant, and the association between X_n and Y is typically overshadowed. However, when Y is transformed into $\widetilde{Y}$, the situation reverses. The noisy label $\widetilde{Y}$ directs the model to focus on features that are causally associated with noise labels, causing the model to overemphasize X_n. This shift introduces spurious associations that distort the model's learning process. Features within X_n, which present local consistency across samples of different categories (Y and $\widetilde{Y}$), misleading the model to capture its causal significance with $\widetilde{Y}$. Consequently, models trained on noisy data form erroneous shortcuts, bypassing the real causal pathways established by X_c.

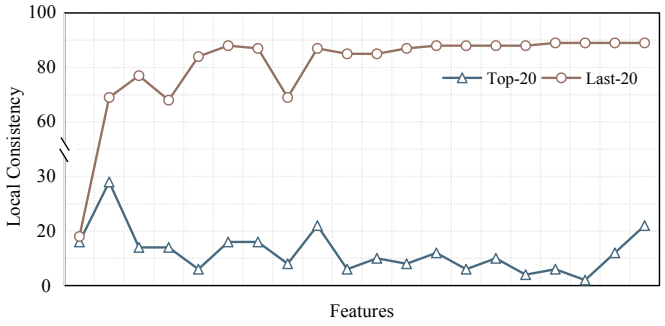


Fig. 4: The Local Consistency from MALTLS-22 Dataset. The horizontal axis represents the top-20 or last-20 important features. The vertical axis represents the number of category combinations with similar distribution.

B. Instantiated Local Consistency

Previously, we discussed how local features with similar distributions can promote undesired associations in the model, thereby propagating bias. In this section, we formally define this phenomenon as *local consistency* and further instantiate it across traffic categories in network data. Intuitively, local consistency arises from the fact that different types of network traffic are often generated under similar conditions [13], [14]. For example, IoT devices sharing the same firmware or protocols may produce comparable traffic patterns, regardless of whether the activity is benign or malicious. Furthermore, the evolution of attack strategies has also enabled adversaries to mimic benign behaviors, such as embedding malicious payloads in encrypted traffic [50], [53], making it ever more challenging to distinguish between classes in feature space. By incorporating label noise learning in multi-class scenarios, our approach effectively addresses local consistency both within malicious categories and between benign and malicious traffic. This strategy meets the challenges posed by increasingly complex network attacks and facilitates more fine-grained detection.

To illustrate this phenomenon, we analyze the MALTLS-22 dataset. As shown in Fig. 3, different attack types exhibit almost identical distributions across certain features. indicating that even when two traffic samples demonstrate high local consistency, an ideal classifier should still be capable of categorizing them into distinct classes. To further examine the pervasiveness of local consistency, we conduct a detailed analysis using the same dataset. Specifically, we measure the similarity of feature distributions across categories using the Kolmogorov-Smirnov (KS) test at a 0.05 significance level. Feature importance is determined via a random forest classifier, and both the top-20 and bottom-20 features are evaluated. The results presented in Fig. 4 show substantial local consistency, particularly among the last-20 features, where over 80 category combinations share similar distributions. Notably, even the top-20 most important features, typically expected to provide strong discriminative power, exhibit significant overlap across categories. This high degree of local consistency, even

among important features, increases the likelihood that noisy samples can mislead the model into learning naive patterns that fail to capture the true decision boundaries. Consequently, the model may struggle to associate them with their correct categories, resulting in degraded performance.

Takeaway. Through the above causal analysis and instantiation of local feature consistency, we have developed a comprehensive understanding of how noisy labels impact model performance. From a causal perspective, the interaction between causal and non-causal features explains how noisy labels distort true causal pathways, leading to spurious associations that misguide learning. Local consistency, arising from shared feature distributions across categories, further amplifies this problem by encouraging the model to learn naive, non-discriminative patterns. Empirical analysis of real-world network traffic confirms that even high-importance features can exhibit substantial overlap across classes. These findings underscore the importance of disrupting local consistency and suppressing spurious associations to improve the robustness of intrusion detection models in noisy environments.

C. Motivation for CoLD

Noisy labels cause models to overemphasize partial features with local consistency across categories, leading to spurious associations and degraded performance. To mitigate the effects of local consistency, an intuitive approach is to create multiple views of the feature space for the model, rather than relying solely on a single input [13], [31]. Therefore, we propose partitioning the raw feature set into multiple subsets, akin to augmentation strategies in image processing such as cropping [59]. These multiple views help isolate the pure causal effects from the misleading associations introduced by locally consistent features. Considering that the cropped image patches remain continuous while the correlations between different traffic features may be insufficient, we reorder the input features before partitioning to enhance the inter-relationships within the subsets. By employing joint self-supervised learning across these partitions, the model diversifies the latent representations and enhances its ability to distinguish true causal associations from spurious ones. Guided by the observed labels, each subset is analyzed for its distinct causal impact, enabling robust differentiation between causal and non-causal associations, thereby identifying noise labels. This methodology strengthens the reliability of IDS models by maintaining clear categorical distinctions and mitigating performance degradation in noisy scenarios.

Takeaway. The motivation behind CoLD lies in addressing the spurious associations introduced by noisy labels and local consistency. By reordering and partitioning features to create multiple subsets, CoLD enables the model to learn fine-grained and robust representations. This multi-view strategy, combined with self-supervised learning, enhances the model's ability to distinguish true causal signals from noise, ultimately improving the reliability and robustness of network intrusion detection in noisy environments.

IV. Methodology

A. Preliminary

Problem Setting. Let $\mathbf{x} \in \mathbb{R}^d$ be a network traffic flow comprising d dimensions of features. Each flow in dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ associates with an annotation $y_i \in \{1, 2, ..., K\}$, where $|\mathcal{D}|$ is the total number of flows and K denotes the number of categories. In practice, obtaining pure annotations is challenging and some flows may be mislabeled, i.e., y_i may be a noise label. The goal of noisy label learning is to train a robust model $\mathcal{M}(\theta) : \mathbb{R}^d \rightarrow \mathbb{R}^K$, that is resilient to the noisy labels in training data and generalizes effectively on clean testing data. Generally, $\mathcal{M}(\theta)$ can be expressed as $\mathrm{E} \circ \mathrm{G}$, where E is an encoder network that maps the input $\mathbf{x}$ to its latent representation and G generates predictions coherently based on the representation.

Overview of CoLD. Fig. 5 illustrates our proposed CoLD, designed to enhance the robustness of network intrusion detection systems by effectively filtering noisy labels from datasets. Building on the investigation outlined in Section III, CoLD identifies noise labels by analyzing the divergence of causal associations between multiple representations and their potential true labels. The framework consists of three main components: Feature Reordering, Local Joint Learning, and Causal Collaborative Denoising.

(1) Feature Reordering reorganizes features into subsets with reordered adjacency relationships. The process leverages inter-relationships among input features, grouping those with similar semantics to enhance spatial correlation. Through subset partitioning, each subset maximizes local correlations, thereby retaining essential semantic information. It is assumed that the model typically has incompletely coincident feature subsets, but some level of overlap between feature subsets is also allowed. As visualized in the diagram with connected nodes, given an input traffic sample with several features $\mathbf{x} = \{\mathrm{f}_1, \mathrm{f}_2, \mathrm{f}_3, \mathrm{f}_4, \mathrm{f}_5, \mathrm{f}_6, \mathrm{f}_7\}$, its features are first reordered based on Pearson correlation coefficients, where f_i denotes the i-th feature of sample. After that, the reordered sample is partitioned into a series of subsets $\mathbf{x} = \{x_1, x_2, x_3\}$, where $x_1 = \{\mathrm{f}_1, \mathrm{f}_6, \mathrm{f}_3\}$, $x_2 = \{\mathrm{f}_3, \mathrm{f}_5, \mathrm{f}_2\}$, and $x_3 = \{\mathrm{f}_2, \mathrm{f}_4, \mathrm{f}_7\}$, for subsequent processing.

(2) Local Joint Learning focuses on learning noise-independent representations through local alignment and global reconstruction operations. The high-level idea of Local Joint Learning is to disrupt the local consistency and obtain a global representation that is causally related to the potential true label. It should be noted that the process is self-supervised, thus suppressing spurious associations from noisy labels and

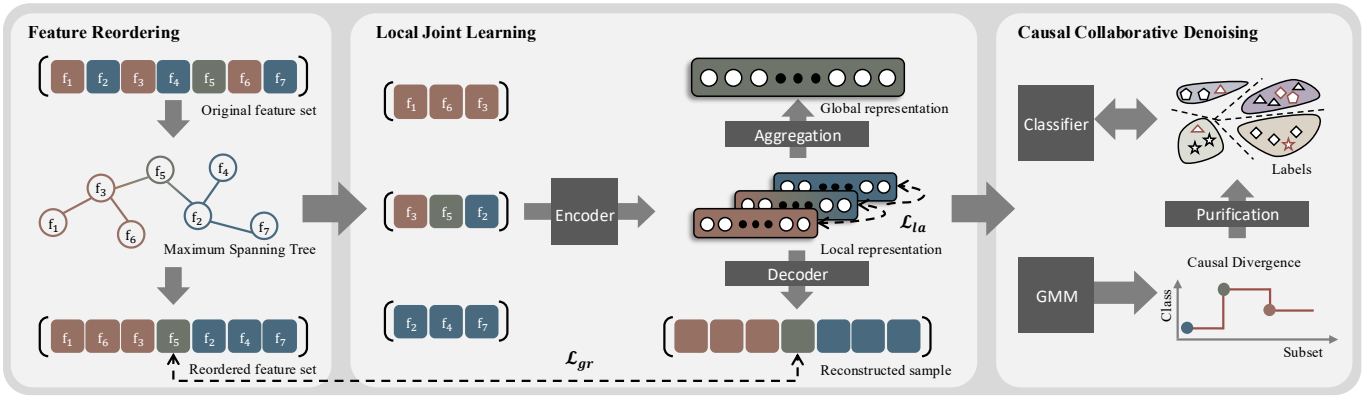


Fig. 5: The Illustration of CoLD. Feature Reordering enhances semantic relevance by rearranging the feature set based on Pearson correlation coefficients. Local Joint Learning partitions the reordered features into multiple subsets and applies self-supervised training to disrupt local consistency, encouraging the model to learn fine-grained and robust representations. Causal Collaborative Denoising purifies the dataset by using a Gaussian Mixture Model (GMM) to analyze the divergence of causal associations between multiple representations and their potential true labels. This process identifies and filters noisy samples, resulting in a purified dataset for training a noise-resilient classifier.

facilitating the learning of deep semantics of traffic flows. Specifically, each subset is fed into a shared encoder to obtain local fine-grained representations. These representations are used to reconstruct the sample through a decoder and generate a global representation by aggregating local representations. Guided by local alignment loss ($\mathcal{L}_{la}$) and global reconstruction loss ($\mathcal{L}_{gr}$), we could obtain a robust encoder to map the input $\mathbf{x}$ to its latent representation.

(3) Causal Collaborative Denoising identifies noisy labels by evaluating causal associations between feature subsets and potential true labels using a Gaussian Mixture Model (GMM). The GMM models causal associations among multiple representations of the same sample and assigns a potential label for each subset, enabling the quantification of causal divergence to detect label noise. To this end, we propose the Causal Divergence Metric, which measures the probability of noise transfer between labels of multi-subsets and the observed label. Samples with significant causal divergence are regarded as noisy. Unlike methods [13], [31] that rely on raw distances or confidence scores, our approach leverages fine-grained representation modeling to uncover causal divergences, ensuring high accuracy in identifying and isolating noisy labels. Following dataset purification, the resulting purified dataset minimizes the impact of noise and facilitates robust training of the final classifier.

B. Feature Reordering

In noisy label learning, local consistency can mislead the model into focusing on non-causal features, establishing associations with noisy labels, and propagating biases. To mitigate the negative impact of these non-causal associations, we propose dividing the original feature set into subsets and performing representation learning on these subsets. Before partitioning, we reorder the features to exploit the inter-relationships among input features, arranging those features with similar semantics together. This reordering endows the sample with spatial correlation, ensuring that each subset maximizes local feature correlation and retains meaningful information.

We are motivated to employ the Pearson correlation coefficient for feature reordering due to its effectiveness in quantifying linear dependencies among features. In many network traffic datasets, a substantial degree of feature redundancy is attributable to linear or near-linear correlations [48], [60], [61]. Furthermore, the Pearson correlation coefficient is computationally efficient and robust to scale differences among features, making it practical for high-dimensional settings. By capturing these relationships, Pearson correlation enables us to efficiently group and partition relevant features, crucial for mitigating the propagation of noise and bias in downstream learning. Specifically, we compute the Pearson correlation coefficients between features to construct a feature correlation matrix $FCM \in \mathbb{R}^{d\times d}$, where d is the dimensionality of the feature set. Each element of FCM is calculated as:

$$FCM_{ij} = \frac{\mathrm{Cov}(\mathbf{x}(\mathrm{f}_i), \mathbf{x}(\mathrm{f}_j))}{\sigma_{\mathrm{f}_i} \cdot \sigma_{\mathrm{f}_j}}, \forall i, j \in \{1, 2, ..., d\}, \quad (1)$$

where $\mathrm{Cov}(\mathbf{x}(\mathrm{f}_i), \mathbf{x}(\mathrm{f}_j))$ is the covariance between feature f_i and feature f_j, σ_{f_i} and σ_{f_j} are the standard deviations of feature f_i and feature f_j, respectively.

Each element FCM_{ij} represents the absolute value of the Pearson correlation coefficient between two features. To group the most correlated features together, we construct a Maximum Spanning Tree (MST) that maximizes the total weight of all edges in the graph. Feature ordering is then determined by performing a Depth-First Search (DFS) traversal on the MST, starting from the feature pair with the highest correlation. This approach ensures that the resulting subsets maximize correlations with adjacent features, providing a robust basis for subsequent learning tasks.

C. Local Joint Learning

The high-level idea of Local Joint Learning is to disrupt the local consistency and derive a global representation that is causally related to the potential true label. The Local Joint Learning is self-supervised, thus avoiding interference from noisy labels and facilitating the learning of deep semantics of traffic flows.

We consider disrupting local consistency on two levels: original feature space and latent representation space. In the original space, we generate a perturbed version of the traffic sample as the model input through the feature obfuscation technique. Given a feature-reordered sample $\mathbf{x}_i$, we can easily construct a series of subsets $\{x_{i,1}, x_{i,2}, \ldots, x_{i,M}\}$, where M is the number of feature subsets. This partitioning is similar to image cropping in image processing [59], transforming the representation learning problem into a multi-view learning task where each subset represents a local view of the full feature set. Theoretically, for traffic flows of the same category, the local view provided by each feature subset should assist the model in learning similar representations, as they embody similar semantics. However, noisy labels conflict with the true semantics and induce locally consistent subsets to establish causal associations with them, which should be avoided as much as possible. To counter this, we propose the Local Joint Learning method that comprehensively learns the true semantic representations of samples from multiple perspectives. This approach aims to mitigate the influence of locally consistent features in the latent space and provide a reliable basis for identifying label noise. Specifically, joint learning is conducted in a self-supervised manner, modeling the relationships between different subsets of a sample through local alignment. Global reconstruction is employed to introduce global information into subsets' representation, thereby obtaining label-independent robust representations.

Feature Obfuscation. According to Section III-B, local consistency indicates that the feature subsets of different traffic flow categories often share a similar distribution. Therefore, we aim to increase the diversity of local features through feature obfuscation. To achieve this, we apply random masking to the flow $\mathbf{x}_i$ along the feature dimension. The mask vector $\mathbf{m}$ is randomly sampled from a Bernoulli distribution with a predefined probability parameter δ. Subsequently, the sample $\mathbf{x}_j$ from the same batch as $\mathbf{x}_i$ and the mask vector $\mathbf{m}$ are used jointly as inputs. The perturbed version of each sample is generated as follows:

$$\tilde{\mathbf{x}}_i = \mathbf{m} \odot \mathbf{x}_i + (1 - \mathbf{m}) \odot \mathbf{x}_j, \tag{2}$$

where $\odot$ denotes element-wise multiplication. The perturbed sample $\tilde{\mathbf{x}}_i$ is then processed by an encoder to transform it into several subset representations, and a corresponding decoder reconstructs the original input data.

Local Alignment. Aligning representations of each subset ensures that the extracted representations are consistent within their local neighborhoods, thus mitigating the impact of local consistency features on the model. To achieve this goal, we feed each feature subset into a shared encoder E to obtain the corresponding local latent representations. Subsequently, by aligning these feature representations of each subset, we encourage the encoder to derive representations from multiple local views, better exposing the true semantics and eliminating confounding features. Specifically, for multiple subsets of $\tilde{\mathbf{x}}_i$, the task could be turned into a multi-view representation learning problem. Therefore, we design a local alignment loss function $\mathcal{L}_{la}$ based on contrastive learning [62].

Given a sample from a single feature subset with input $x_{i,j}$, which represents features of the j-th subset in the i-th sample, the encoded latent representation of this subset is denoted as $\mathbf{z}_{i,j} = \mathrm{E}(x_{i,j})$. The positive pair is defined as the feature representations $\{\mathbf{z}_{i,j}, \mathbf{z}_{i,p}\}$ from different subsets of the same sample, while all other samples are considered negative pairs. The local alignment loss is shown as follows:

$$\mathcal{L}_{la} = -\sum_{i=1}^{N}\sum_{j=1}^{M} \log \frac{\exp\left(\mathrm{sim}\left(\mathbf{z}_{i,j}, \mathbf{z}_{i,p}\right)/\tau\right)}{\sum_{n=1}^{N}\sum_{q=1}^{M} \exp\left(\mathrm{sim}\left(\mathbf{z}_{i,j}, \mathbf{z}_{n,q}\right)/\tau\right)}, \tag{3}$$

where $\mathrm{sim}\left(\mathbf{z}_{i,j}, \mathbf{z}_{i,p}\right)$ measures similarity between the latent representations $\mathbf{z}_{i,j}$ and $\mathbf{z}_{i,p}$, here dot product is employed due to its stability. $\mathbf{z}_{n,q}$ denotes all subsets from other samples that are treated as negative pairs. τ is an adjustable temperature parameter. N and M denote the total number of samples in the batch and the number of feature subsets, respectively. $\mathcal{L}_{la}$ encourages the encoder to learn fine-grained representations by leveraging multiple local views of the same sample.

Global Reconstruction. We hypothesize that an effective local representation should be a partial sampling of the global feature, which should reflect as much as possible the overall structure and relationships of the global feature. To achieve this, we propose a global reconstruction loss. This loss minimizes the distance between the reconstructed local features and the original input global features. For a sample $\tilde{\mathbf{x}}_i$, the reconstructed version from subset j is represented as $\hat{\mathbf{x}}_{i,j} = \mathrm{D}(\mathbf{z}_{i,j}) \in \mathbb{R}^d$, where D is a decoder. The global reconstruction loss is defined as:

$$\mathcal{L}_{gr} = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M} ||\hat{\mathbf{x}}_{i,j}, \tilde{\mathbf{x}}_i||_2. \tag{4}$$

To ensure compatibility, the decoder's output layer is extended to match the dimensions of the reconstructed local features with those of the global features. Finally, the overall objective of Local Joint Learning combines local alignment and global reconstruction losses:

$$\mathcal{L} = \mathcal{L}_{la} + \mathcal{L}_{gr}. \tag{5}$$

By minimizing this objective, the encoder is encouraged to learn discriminative features from multiple local perspectives while ensuring global perception, thereby obtaining label-independent true semantic information. For a single sample $\mathbf{x}_i$ with a given subset number M, we can obtain M representations $\mathbf{z}_{i,j}, j \in [1, M]$ from all subsets based on the

encoder output. We then aggregate the representations of all feature subsets to produce an aggregated representation $\mathbf{z}_i$ for collaborative reasoning. The process can be expressed as:

$$\mathbf{z}_{i,0} = \mathrm{agg}(\mathbf{z}_{i,1}, \mathbf{z}_{i,2}, ..., \mathbf{z}_{i,j}), j \in [1, M], \tag{6}$$

Here, agg indicates the aggregation function, which can be MEAN, SUM, CANCAT, or other methods. A comparison of these aggregation methods is provided in Section V-C. Once the global representation is obtained, it is utilized for causal collaborative inference alongside multiple subset representations to identify noisy labels.

D. Causal Collaborative Denoising

Existing dataset purification methods, such as [13], detect noisy labels by measuring the distance between a sample and its confidence sample. However, this approach is limited because obtaining a reliable confidence sample is challenging, especially in high-noise scenarios. To overcome these limitations, we leverage a Gaussian Mixture Model (GMM) to establish the causal relationship between the multi-view representations of a sample and its potential label.

GMM is well-suited for this task as it can effectively model complex distributions of network traffic samples, which often exhibit class overlap and diverse attack behaviors. Moreover, GMM provides soft probabilistic assignments for each sample rather than hard binary decisions, enabling the model to capture subtle distinctions between clean and noisy samples. This capability is crucial for accurately identifying the potential causal relationship between feature subsets and their assigned labels. Samples that exhibit significant causal divergence-meaning they cannot be well characterized by the Gaussian component corresponding to their label-are identified as potentially having noisy labels.

To model the distribution of each representation $\mathbf{z}_{i,j}$, we first map it through a linear head network denoted by H, resulting in a transformed representation $\widetilde{\mathbf{z}}_{i,j} = \mathrm{H}(\mathbf{z}_{i,j})$. This transformed representation is then modeled using the GMM. To facilitate this, we introduce discrete latent variables $y \in \{1, 2, ..., K\}$, which are responsible for assigning the observations $\widetilde{\mathbf{z}}_{i,j}$ to one of the K mixture components. Consequently, the GMM operates in an unsupervised manner, defining the probability distributions over the transformed data points $\widetilde{\mathbf{z}}_{i,j}$:

$$\gamma_{i,j,k} = \frac{\pi_k \mathcal{N}\left(\widetilde{\mathbf{z}}_{i,j} \mid \mu_k, \sigma_k\right)}{\sum_{l=1}^{K} \pi_l \mathcal{N}\left(\widetilde{\mathbf{z}}_{i,j} \mid \mu_l, \sigma_l\right)}, \tag{7}$$

where π_k represents the weight of the k-th Gaussian component, satisfying $\sum_{k=1}^{K} \pi_k = 1$. $\mathcal{N}(\widetilde{\mathbf{z}}_{i,j} | \mu_k, \sigma_k)$ is the probability density function of this component with mean μ_k and covariance matrix σ_k. Each representation $\widetilde{\mathbf{z}}_{i,j}$ is assigned a predicted label $\widetilde{y}_{i,j} = \arg\max_k \gamma_{i,j,k}$, thus the labels from multiple feature subsets are treated as multi-labels for the sample $\mathbf{x}_i$.

In an ideal scenario where all samples have clean labels, the transformed latent vector $\widetilde{\mathbf{z}}_{i,j}$ would be identical to the annotation y_i, and the parameters μ_k and σ_k can be solved through a standard Expectation-Maximization (EM) algorithm. However, in the presence of noisy labels, $\widetilde{\mathbf{z}}_{i,j}$ is expected to be estimated in an unsupervised manner, independent of label y_i. To bridge this gap, we use predictions $\overline{y}_i$ from a downstream classifier to update parameters of the linear head θ_h, effectively linking y_i and $\widetilde{\mathbf{z}}_{i,j}$. When the process is controlled by cross-entropy loss, it can be written as:

$$\theta_h^* = \min_{\theta_h} \left[-\sum_{i=1}^{N} \overline{y}_i \log y_i \right]. \tag{8}$$

As mentioned in Section III-B, the subset with local consistency will establish associations with the noisy label under its guidance, whereas the other subsets will maintain relative stability. By establishing the multiple causal associations between the latent representation $\widetilde{\mathbf{z}}_{i,j}$ and clustering label $\widetilde{y}_{i,j}$, we employ collaborative inference to identify incorrectly labeled samples. Subset representations influenced by noisy labels tend to form conflicting causal associations. Inspired by this, we leverage the divergence in causal associations across subsets to detect potentially mislabeled samples. If the causal association of a subset is significantly different from others, it indicates that the sample may be noisy. To identify these inconsistencies, we propose a Causal Divergence Metric (CDM) to quantify the probability of noise transfer between a sample's multi-label $\widetilde{y}_{i,j}$ and the original observed label y_i.

$$\mathrm{CDM}(\mathbf{x}_i) = \frac{1}{M} \sum_{j=1}^{M} P(\widetilde{y}_{i,j} \neq y_i | \mathbf{x}_i), \tag{9}$$

where $P(\widetilde{y}_{i,j} \neq y_i | x_i)$ signifies the transition probability between the cluster label and the observed label, indicating their divergence of causal associations. We use this metric to determine if the sample $\mathbf{x}_i$ has been mislabeled. For a binary probability model, $\mathrm{CDM}(\mathbf{x}_i)$ can be simplified to:

$$\mathrm{CDM}(\mathbf{x}_i) = \frac{1}{M} \sum_{j=1}^{M} \mathbb{1}(\widetilde{y}_{i,j} \neq y_i | \mathbf{x}_i). \tag{10}$$

Here, $\mathbb{1}(\cdot)$ is an indicator function that returns 1 if the condition is true, otherwise 0. By comparing the divergence metric results with a predefined threshold ϵ, we identify samples with noisy original labels, enabling their isolation from the dataset. In this paper, we adopt a rigorous evaluation by setting $\epsilon = 0$, ensuring that a sample $\mathbf{x}_i$ is retained only if all its subsets are causally associated with the observed label.

$$\mathcal{D}_{\mathrm{p}} \leftarrow \mathcal{D} \setminus \{\mathbf{x}_i : \mathrm{CDM}(\mathbf{x}_i) > \epsilon\}. \tag{11}$$

Samples with lower causal divergence are selected to construct a clean subset of data. This purified dataset $\mathcal{D}_{\mathrm{p}}$ is then used to train a downstream classifier, ensuring the model learns from accurate and representative samples. By isolating noisy labels and leveraging clean data, the classifier achieves significantly improved robustness and accuracy in network intrusion detection.

In summary, CoLD is a collaborative label denoising framework specifically designed to enhance the robustness of data-driven IDS models in noisy environments. The complete algorithm is outlined in Algorithm 1. CoLD comprises three integrated components: Feature Reordering, which reorganizes input features to optimize semantic coherence and prepare meaningful subsets for learning; Local Joint Learning, which applies self-supervised strategies to disrupt local consistency and extract fine-grained and robust representations; and Causal Collaborative Denoising, which utilizes GMM along with a Causal Divergence Metric to identify and isolate noisy labels. By combining these components, CoLD effectively filters mislabeled samples and generates a purified dataset for downstream classifier training, enabling the construction of high-performance, noise-resilient IDS models.

V. Experiments on Benchmark Datasets

A. Experimental Setting

TABLE I: Dataset Description.

Dataset	CICIDS-2017	MALTLS-22
Benign	32.50%	36.94%
Mal. (Head-3)	44.90%	12.33%
Mal. (Tail-3)	9.70%	4.10%
Mal. (Others)	12.9%	46.63%
# of Classes	9	23
Gini coefficient	0.82	0.84

Mal. is the abbreviation of Malicious.

Datasets. In this work, we use a refined version of CICIDS-2017 [14] and MALTLS-22 [13] to evaluate the performance of the proposed CoLD. The reason for choosing these two datasets is that they are widely used for intrusion detection tasks [4], [5], [6] and noisy label learning [13]. They cover diverse attack categories, which facilitates the construction of a more challenging noise environment. In real-world scenarios, benign traffic significantly outweighs malicious traffic, leading to an imbalanced distribution of classes within the dataset. This disparity can be quantified using the Gini coefficient, which is calculated as:

$$\text{GiniCo.} = 1 - \sum_{i=1}^{k} p_i^2, \tag{12}$$

where p_i represents the probability of each class within the dataset. A higher Gini coefficient signifies greater imbalance, closer to 1, whereas a coefficient closer to 0 indicates a more balanced class distribution. The dataset descriptions are summarized in Table I. For simplicity, we present the proportion of benign traffic and group the malicious categories into Head-3, Tail-3, and Others. The data is split into training and testing sets using an 8:2 ratio. Controlled proportions of label noise are introduced only in the training set, while the test set remains clean and unchanged for evaluation.

The MALTLS-22 dataset [13] contains 22 types of realistic encrypted malicious traffic. The traffic was captured over four

Algorithm 1 CoLD: Collaborative Label Denoising

Input: Dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|}$, number of categories K, number of feature subsets M, mask probability δ, threshold ϵ, number of iterations T, encoder network e, decoder network d, projection head h, classifier g.

Output: Purified dataset $\mathcal{D}_\text{p}$, encoder network e.

Feature Reordering

compute feature correlation matrix $FCM \in \mathbb{R}^{d\times d}$: $FCM_{ij} = \frac{\text{Cov}(\mathbf{x}(\text{f}_i),\mathbf{x}(\text{f}_j))}{\sigma_{\text{f}_i}\cdot\sigma_{\text{f}_j}}, \forall i,j \in \{1,2,...,d\}$.

construct Maximum Spanning Tree (MST) using FCM.

reorder features through depth-first search (DFS) traversal of MST.

Local Joint Learning

for $t = 1$ to T **do**

  for each sample $\mathbf{x}_i \in \mathcal{D}$ **do**

    generate perturbed version of $\mathbf{x}_i$: $\tilde{\mathbf{x}}_i = \mathbf{m}\odot\mathbf{x}_i + (1-\mathbf{m})\odot\mathbf{x}_j$, where $\mathbf{m} \sim \text{Bernoulli}(\delta)$.

    partition $\tilde{\mathbf{x}}_i$ into M subsets $\{x_{i,1}, x_{i,2}, ..., x_{i,M}\}$ based on feature order.

    encode latent representations: $\mathbf{z}_{i,j} = \text{e}(x_{i,j})$, where $j \in [0, M]$.

    align local views using local alignment loss: $\mathcal{L}_{la} = -\sum_{i=1}^{N}\sum_{j=1}^{M} \log \frac{\exp(\text{sim}(\mathbf{z}_{i,j},\mathbf{z}_{i,p})/\tau)}{\sum_{n=1}^{N}\sum_{q=1}^{M}\exp(\text{sim}(\mathbf{z}_{i,j},\mathbf{z}_{n,q})/\tau)}$.

    reconstruct global features from local views: $\hat{\mathbf{x}}_{i,j} = \text{d}(\mathbf{z}_{i,j})$, where $j \in [0, M]$.

    compute reconstruction loss: $\mathcal{L}_{gr} = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M} ||\hat{\mathbf{x}}_{i,j}, \tilde{\mathbf{x}}_i||_2$.

    update encoder and decoder parameters via: $\mathcal{L} = \mathcal{L}_{la} + \mathcal{L}_{gr}$.

    aggregate representations of subsets to get global representation: $\mathbf{z}_{i,0} = \text{agg}(\mathbf{z}_{i,1}, \mathbf{z}_{i,2}, ..., \mathbf{z}_{i,j})$, where $j \in [1, M]$.

  end for

end for

Causal Collaborative Denoising

for each sample $\mathbf{x}_i \in \mathcal{D}$ **do**

  for each subset representation $\mathbf{z}_{i,j}, j \in [1, M]$ **do**

    project to latent space using h: $\widetilde{\mathbf{z}}_{i,j} = \text{h}(\mathbf{z}_{i,j})$

    fit Gaussian Mixture Model (GMM) and compute posterior probabilities: $\gamma_{i,j,k} = \frac{\pi_k\mathcal{N}(\widetilde{\mathbf{z}}_{i,j}|\mu_k,\sigma_k)}{\sum_{l=1}^{K}\pi_l\mathcal{N}(\widetilde{\mathbf{z}}_{i,j}|\mu_l,\sigma_l)}$.

    obtain cluster labels: $\widetilde{y}_{i,j} = \arg\max_k \gamma_{i,j,k}$

  end for

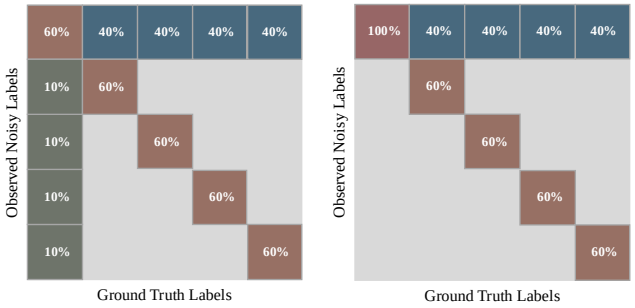
  predict label for $\mathbf{z}_{i,0}$ using g: $\overline{y}_i = \text{g}(\mathbf{z}_{i,0})$

  update head parameters via cross-entropy loss: $\theta_h^* = \min_{\theta_h}\left[-\sum_{i=1}^{N}\overline{y}_i \log y_i\right]$

  computer causal divergence: $\text{CDM}(\mathbf{x}_i) = \frac{1}{M}\sum_{j=1}^{M} P(\widetilde{y}_{i,j} \neq y_i|\mathbf{x}_i)$

  remove noisy samples to obtain a purified dataset: $\mathcal{D}_\text{p} \leftarrow \mathcal{D} \setminus \{\mathbf{x}_i : \text{CDM}(\mathbf{x}_i) > \epsilon\}$

end for



(a) Symmetric Noise (b) Asymmetric Noise

Fig. 6: Example of Label Conversion Matrices with Noise Ratio of 40%.

years from 2018 to 2021 and labeled with precise threat intelligence to ensure ground truth. Within this dataset, all traffic was encrypted with TLS, resulting in similar behavior across local features. From Table I, it can be seen that the benign traffic and Head-3 malicious traffic account for nearly 50%, which reveals its imbalance. We have maintained the original data distribution in the experiments.

The updated CICIDS-2017 dataset [14] has been refined by removing errors and inconsistencies. In its original form, over 80% of the samples are labeled as benign, making it unsuitable for evaluating robustness under noisy conditions. In such skewed distributions, a naive model could achieve deceptively high accuracy by predicting all samples as benign. To reduce this bias, we downsampled dominant classes and excluded those with fewer than 1000 instances. This filtering process resulted in a representative subset containing 9 categories, with a class distribution more closely aligned with the Gini coefficient observed in MALTLS-22.

Noise Setting. Following established setups in prior noise modeling studies [13], [31], [37], we implement two distinct label corruption schemes: symmetric and asymmetric noise. In the symmetric noise setting, label corruption is applied uniformly across both benign and malicious samples. In contrast, the asymmetric noise setting introduces corruption exclusively within the malicious class. This setting simulates a more realistic adversarial scenario in which malicious behavior is intentionally disguised as benign. Fig. 6 presents the label transition matrices under a 40% noise ratio for both settings. The horizontal axis denotes the ground truth labels, while the vertical axis represents the observed noise labels. The first row and column correspond to benign traffic, and the remaining entries represent various types of malicious traffic. In the left figure, 40% of benign samples are randomly mislabeled as various types of malicious traffic, and vice versa. In the right figure, 40% of malicious samples are mislabeled as benign, while all benign labels remain unchanged. To enable comprehensive evaluation, we vary the noise ratio across $\{10\%, 20\%, 40\%, 50\%, 60\%, 80\%\}$.

Baselines. We consider three categories of work for comparison: intrusion detection methods, robust training methods, and dataset purification methods. Representative works are summarized in Table II, which includes six state-of-the-art noisy label learning methods [36], [27], [28], [31], [18], [13] and two of the latest intrusion detection models [5], [6]. Here is a brief introduction to them.

TABLE II: Summary of Selected Compared Methods.

Category	Method
Intrusion Detection	ACID [5], CLEID [6]
Robust Training	Decoupling [36], Co-Teaching [27], Co-Teaching+ [28]
Dataset Purification	FINE [31], MORSE [18], MCRe [13]

- ACID [5] utilizes supervised adaptive clustering to learn cluster centers that serve as extensions of the input features, which enhances the robustness against outliers;
- CLEID [6] leverages data augmentation and self-supervised contrastive learning to extract semantic information of different traffic types, thereby enhancing the model's robustness;
- Decoupling [36] updates the parameters based on samples that receive different predictions from two classifiers;
- Co-Teaching [27] trains two networks with distinct initial states simultaneously and prompts them to select clean samples from each other;
- Co-Teaching+ [28] is similar to Co-Teaching, but it has a greater tendency to select samples of disagreement for another network;
- FINE [31] filters noisy labels by employing eigen decomposition of their gram matrix in the latent space;
- MORSE [18] is a representative semi-supervised noisy learning work. It takes possibly incorrectly labeled data as unlabeled data and thus avoids their potential negative impact on model training.
- MCRe [13] is a framework for handling noise malicious traffic based on self-supervised representation learning and distance measurement, capable of filtering samples with noise labels to purify datasets.

Model Structure and Optimizer. In Local Joint Learning, the encoder consists of a two-layer fully connected network, with each hidden layer having the same dimensionality as the input features. The decoder adopts a symmetric structure to reconstruct the original input. For Causal Collaborative Reasoning, a linear head is appended to the encoder, followed by a fully connected layer that serves as the classifier. Both training stages utilize the Adam optimizer with an initial learning rate of 0.001 and a batch size of 128. All experiments were conducted on NVIDIA RTX 3090 GPUs and repeated three times using different random seeds. The reported results represent the average metrics across these three trials. The code will be released soon.

Evaluation Metric. To evaluate the performance, the F1-score is selected as the evaluation metric. Considering the imbalance of network traffic datasets, the macro measurement is adopted. As the experiments of the synthetic noisy datasets are repeated

three times, the mean and standard deviation of the test F1-score are calculated. With the detection result and the ground truth of one testing dataset, we can calculate the number of true positive samples (TP), the number of false positive samples (FP), and the number of false negative samples (FN). Then, the three metrics can be computed as follows: precision $= \frac{\text{TP}}{\text{TP+FP}}$, recall $= \frac{\text{TP}}{\text{TP+FN}}$, and F1-score $= \frac{2\times\text{precision}\times\text{recall}}{\text{precision+recall}}$.

Paired t-test. Following previous work [18], we use a paired t-test to measure the statistical significance of performance comparison between MCRe and CoLD. Specifically, given two sets of F1-score obtained by MCRe and CoLD, we first compute their difference, i.e., $\text{Diff} = \text{F1}_{\text{CoLD}} - \text{F1}_{\text{MCRe}}$. With the Diff, we set the null hypothesis as $H_0 : \mathbb{E}(\text{Diff}) < 0$ and compute the p-value. If the p-value is smaller than a threshold (e.g., 0.05), we can reject H_0 and conclude that CoLD outperforms MCRe with statistical significance.

B. Comparison Results

Tables III and IV present results on MALTLS-22 dataset and CICIDS-2017 dataset, respectively. From the results, the following observations and conclusions can be drawn:

(1) Overall, model performance consistently declines as the level of label noise increases. For example, under the 'None' setting (i.e., 0% noise), ACID and CLEID achieve 92.43% and 91.42% performance on the MALTLS-22 dataset, respectively. However, when 20% symmetric noise is introduced, their performance drops significantly to 77.46% and 60.61%. Robust training methods show minimal degradation at lower noise levels, but performance deteriorates rapidly when noise exceeds 50%. In contrast, dataset purification methods exhibit greater resilience to high levels of noise. Notably, MCRE and MORSE maintain robust performance even at a 60% noise level. These results highlight the effectiveness of dataset purification in mitigating the impact of label noise, particularly in intrusion detection tasks. This observation provides strong empirical motivation for our proposed CoLD framework, which builds upon and extends these strengths to further improve robustness in noisy environments.

(2) CoLD demonstrates superior and consistent performance across a variety of settings, highlighting its resilience to label noise. Notably, its robustness becomes increasingly evident as noise levels rise. For instance, even under 60% symmetric noise, CoLD's performance drops by only 3.4% on the MALTLS-22 dataset and 1.6% on the CICIDS-2017 dataset compared to the 20% noise setting. This stability is primarily attributed to the carefully designed Local Joint Learning module, which disrupts local consistencies in the feature space, thereby reducing the model's reliance on spurious, non-causal associations induced by noisy labels. In addition, the Causal Collaborative Denoising module enhances robustness by analyzing causal associations between representations and latent true labels. Through this mechanism, CoLD identifies and filters out samples with significant causal divergence, resulting in more accurate and noise-resilient model training.

(3) Intrusion detection techniques such as ACID [5] and CLEID [6] exhibit subpar performance in the presence of label noise. When exposed to the MALTLS-22 dataset with the symmetric noise ratio escalating from 20% to 40%, ACID and CLEID experience performance drops of 15.75% and 53.68%, respectively. This notable decline underscores the significant challenge that label noise poses to intrusion detection models, highlighting the limitations of current robust intrusion detection technologies in handling such scenarios. Of particular concern is the rapid performance deterioration observed in CLEID, likely stemming from its use of mixups for generating augmented samples. In the presence of label noise, the distinctions between different categories will become blurred, potentially exacerbating the propagation of errors induced by label noise throughout the model's decision-making process.

(4) Robust training methods designed for label noise, such as Decoupling [36], Co-Teaching [27], and Co-Teaching+ [28], aim to prevent models from overfitting to noisy labels by designing robust loss functions and training strategies. While these methods outperform traditional robust intrusion detection methods [5], [6] in low-noise settings, their performance deteriorates significantly in high-noise environments. For instance, on the CICIDS-2017 dataset, the F1-score of Co-Teaching+ drops from over 99% at a 20% noise ratio to less than 10% when the noise ratio reaches 60%. These results suggest that such methods struggle to extract effective domain knowledge and lack positive feedback mechanisms in high-noise scenarios, resulting in ineffective parameter updates and limited generalization capabilities. Consequently, these methods face significant challenges in their application to network intrusion detection, particularly in noisy real-world environments.

(5) FINE [31] detects noisy labels by leveraging feature decomposition and representation similarity, achieving competitive performance in high-noise settings. We guess the reason is that the main eigenvector amplifies the difference between clean and noisy samples, making them easier to identify and remove. While FINE has shown success in image recognition tasks, its performance in traffic classification is limited due to fundamental domain differences. Unlike images, network traffic lacks spatial structure and often exhibits local consistency, where features from different classes share similar distributions. This reduces the separability of feature vectors and weakens the effectiveness of similarity-based noise detection. As a result, FINE struggles to maintain stable performance in the context of network traffic data.

(6) MORSE [18] utilizes a pre-trained model on noisy datasets to distinguish between noisy and clean samples, which relies on a predefined splitting ratio. It then applies a semi-supervised learning approach for classification. For fairness in comparison, we set the splitting ratio to match the actual noise ratio. The results show that MORSE maintains relatively stable performance across different scenarios. We believe that its semi-supervised design enables better utilization of all samples for representation learning, giving it a significant advantage over FINE, which directly filters out suspected noisy samples. However, the pre-trained model in MORSE can be easily misguided by noisy labels, leading to inaccurate noisy sample segmentation. In contrast, CoLD employs a self-supervised

TABLE III: Results on MALTLS-22 Dataset.

Noise Type	None	Symmetric					Asymmetric				
Noise Ratio	0%	10%	20%	40%	50%	60%	10%	20%	40%	50%	60%
ACID	92.43/1.83	81.51/3.21	77.46/4.55	61.71/3.84	31.74/4.02	4.38/1.32	80.01/1.85	79.63/2.18	69.48/3.65	39.30/1.78	2.46/0.19
CLEID	91.42/1.12	80.98/0.02	60.61/1.02	6.93/1.74	2.73/0.14	2.38/0.03	85.39/0.05	60.71/1.25	4.93/0.54	3.42/0.35	2.35/0.03
Decoupling	91.30/0.62	89.12/0.79	88.11/1.13	72.66/2.02	31.73/2.35	3.01/1.14	89.54/0.43	90.00/0.37	75.18/0.90	38.37/4.04	3.54/0.42
Co-Teaching	93.47/0.16	92.85/0.17	87.26/0.19	47.50/0.80	7.95/0.53	2.85/0.15	92.57/0.07	90.75/0.29	73.25/0.64	32.98/4.74	2.66/0.11
Co-Teaching+	91.12/0.31	89.49/0.26	89.18/0.56	74.73/0.60	35.50/1.64	3.50/0.12	90.33/0.15	89.23/0.53	86.59/1.71	44.86/5.47	3.02/0.06
FINE	75.76/0.13	64.97/1.35	65.61/0.54	65.37/0.46	57.21/0.27	46.91/1.54	65.43/0.01	64.96/0.51	61.69/1.00	59.19/1.54	59.04/1.04
MORSE	82.04/1.46	77.91/0.13	76.33/0.71	75.71/0.13	74.39/0.85	74.71/1.20	79.36/0.09	77.63/1.90	74.13/0.28	73.92/2.77	70.13/1.09
MCRe	88.49/3.18	87.73/1.94	88.19/0.68	87.03/0.44	86.96/0.38	86.07/0.82	85.66/1.39	85.56/0.73	85.49/0.64	84.97/0.83	84.37/1.13
CoLD (Ours)	92.97/0.32 p=0.043	93.11/0.19 p=0.017	92.14/0.53 p=0.000	91.82/0.42 p=0.000	90.07/0.67 p=0.000	88.75/0.76 p=0.000	93.55/0.34 p= 0.002	91.91/0.35 p=0.000	90.84/0.38 p=0.000	88.08/0.40 p=0.003	86.48/0.65 p=0.008

TABLE IV: Results on CICIDS-2017 Dataset.

Noise Type	None	Symmetric					Asymmetric				
Noise Ratio	0%	10%	20%	40%	50%	60%	10%	20%	40%	50%	60%
ACID	99.27/0.45	97.13/2.88	97.03/0.83	88.01/1.68	52.27/8.43	5.56/0.27	99.15/2.13	98.05/0.64	88.96/3.27	24.29/2.34	5.67/0.23
CLEID	96.59/0.17	86.82/6.46	81.25/1.89	58.50/0.74	11.56/0.26	5.88/0.12	93.83/0.26	80.86/1.45	34.66/1.78	10.62/1.45	7.46/0.80
Decoupling	99.01/0.16	98.76/0.18	98.61/0.67	95.79/0.85	30.60/4.26	5.82/0.12	98.99/0.22	98.83/0.22	94.88/1.26	46.18/1.58	6.02/0.21
Co-Teaching	99.74/0.33	99.40/0.29	99.32/0.13	72.30/1.36	30.83/1.76	30.26/2.45	99.41/0.12	98.34/0.11	85.75/2.35	45.98/5.35	6.17/0.61
Co-Teaching+	99.70/0.48	99.43/0.52	99.35/0.12	98.15/0.52	41.48/3.53	6.61/5.65	99.58/0.43	99.19/0.17	97.10/0.43	54.22/5.47	6.29/0.31
FINE	88.54/0.21	80.87/0.38	78.85/0.23	81.88/0.24	82.45/0.63	77.93/0.38	85.15/0.11	84.27/0.21	74.25/1.03	88.99/1.41	85.13/2.42
MORSE	98.48/0.84	95.06/0.47	90.25/0.52	82.04/0.44	77.34/1.12	72.11/1.23	89.32/0.51	85.55/0.69	82.82/1.38	74.91/2.17	74.60/1.14
MCRe	98.86/1.30	98.81/1.01	99.13/0.10	98.97/0.32	98.99/0.22	96.13/0.41	97.90/0.92	97.06/0.34	93.66/0.13	90.59/1.04	83.25/2.66
CoLD (Ours)	99.71/0.36 p =0.128	99.51/0.23 p=0.130	99.31/0.18 p=0.029	98.98/0.21 p=0.445	99.07/0.31 p=0.132	97.76/0.57 p=0.002	99.55/0.31 p=0.021	99.48/0.23 p=0.000	98.32/0.34 p=0.000	96.67/0.76 p=0.000	95.43/0.97 p=0.003

TABLE V: Comparision Results with 80% Noise.

Dataset	MALTLS-22		CICIDS-2017	
Noise Type	Sym.	Asym.	Sym.	Asym.
FINE	6.42±0.34	39.66±1.05	6.73±0.12	50.32±2.17
MORSE	55.80±1.83	63.57±2.09	51.82±2.17	62.09±2.35
MCRe	85.97±1.44	79.44±1.47	90.97±1.38	84.63±4.23
CoLD (Ours)	87.04±1.12 p=0.014	84.02±0.89 p=0.003	95.16±1.51 p=0.000	93.54±1.06 p=0.020

approach during the Local Joint Learning phase, creating label-independent representations from multiple views, which are then used in the subsequent denoising process. Moreover, our method does not require a predefined noisy label ratio, which significantly broadens its applicability.

(7) While MCRe [13] falls short of CoLD in overall performance, it still achieves notable results by leveraging carefully designed representation constraints and distance-based metrics, outperforming several baseline methods. However, its reliance on distance-based noisy label detection leads to the removal of clean samples that are distant from the confidence center, resulting in the loss of valuable information. This limitation is reflected in the model's sharp performance fluctuations in the clean setting (i.e., large standard deviations in "None" setting) and its inferior performance compared to robust training methods under low noise settings, like Sym-10%. In contrast, CoLD avoids this issue by employing causal divergence to identify noisy labels. This approach is grounded in an understanding of traffic properties, allowing CoLD to more effectively distinguish noise from clean data and preserving informative samples to achieve strong performance.

We evaluate performance under the challenging 80% noise scenario. While such extreme noise rates are uncommon in static datasets, this setting is critical for validating robustness in open-world environments [13], [31], [35], [43]. In security systems, annotation models often introduce systematic noise due to concept drift (e.g., a legacy model may mislabel a large proportion of unseen attack samples as benign) [50], [63]. Under traditional supervised learning frameworks, if noisy labels dominate the class distribution (e.g., 80% of benign samples flipped to malicious in binary classification), models tend to overfit erroneous distributions due to label dependency. However, this work focuses on multi-class scenarios, where noise is typically more dispersed (see Fig. 6). Even with an overall noise rate of 80%, label flips for individual classes are generally sparse (e.g., 10% "Benign→Dos Attack", 10% "Benign→Bot Attack") rather than concentrating on a single mislabeling direction.

As shown in Table V, CoLD maintains strong performance even under such extreme settings. This result underscores the advantages of Causal Collaborative Denoising over distance-based dataset purification methods like MCRe. Specifically, MCRe assumes clear separability between clean and noisy samples, which becomes unreliable at high noise rates. In

such settings, local consistency across traffic categories can mislead the model into learning spurious associations, blurring the boundary between clean and mislabeled data. CoLD overcomes this by employing Local Joint Learning, which creates multi-view representations from feature subsets in a self-supervised manner. This disrupts spurious correlations caused by local consistency and encourages the model to learn causal patterns shared across views. While causal features across categories may still offer weak learning signals, CoLD amplifies this signal through multi-view comparison. By analyzing the causal discrepancies between noisy labels and GMM-based predictions, CoLD isolates noisy samples without relying on label distribution statistics. This design enables robust learning even in high-noise environments.

Takeaway. The experimental results highlight the effectiveness and robustness of CoLD in handling noisy labels, even under extreme conditions. Compared to state-of-the-art methods, CoLD consistently demonstrates superior performance across various noise scenarios, including the challenging 80% noise setting. Its superior performance and generalization highlight the strength of Causal Collaborative Denoising in isolating noisy samples and enabling noise-resilient network intrusion detection in real-world environments.

C. Ablation Study

TABLE VI: Ablation Study with 40% Symmetric Noise. FR, LA, GR, and LP denote Feature Reordering, Local Alignment, Global Reconstruction, and Label Purification, respectively.

FR	MASK	LA	GR	LP	MALTLS-22	CICIDS-2017
●	0.1	●	●	●	90.96	**98.98**
●	0.3	●	●	●	**91.82**	98.52
●	0.5	●	●	●	91.36	98.34
○	0.3	●	●	●	91.69	96.39
●	0.3	○	●	●	89.14	97.01
●	0.3	●	○	●	88.55	96.60
●	0.3	○	○	●	84.67	91.08
●	0.3	●	●	○	85.85	89.32

● with; ○ without.

To evaluate the contributions of each proposed component, we conduct an ablation study on the MALTLS-22 and CICIDS-2017 datasets under 40% symmetric noise. The results are presented in Table VI, where FR, LA, GR, and LP denote Feature Reordering, Local Alignment, Global Reconstruction, and Label Purification, respectively. Across both datasets, CoLD consistently outperforms its ablated variants, confirming the effectiveness of the full framework. Each component contributes to overall performance improvements, with the best results achieved through their integration. Among these components, LP stands out as a key contributor, highlighting the strength of the proposed Causal Collaborative Denoising module. LA and GR constitute the Local Joint Learning module, and omitting these operations led to a significant decline in performance. This demonstrates that the Local Joint Learning module effectively maps features to the hidden representation space, enhancing the model's ability to capture causal association. Additionally, we observe that FR provides a smaller performance gain on MALTLS-22 compared to CICIDS-2017. This can be attributed to the inherent feature correlations in the MALTLS-22 dataset, which reduce the need for reordering. Nevertheless, we recommend retaining FR, as it enhances adaptability in scenarios where the original feature arrangement lacks correlation.

TABLE VII: Comparison of Different Aggregation Methods.

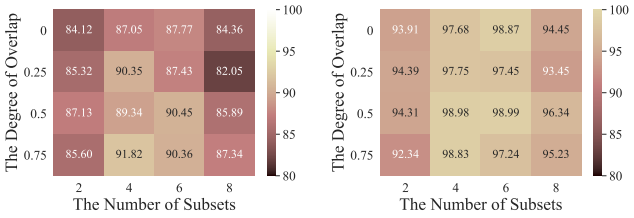
Method	Sym-40%	Sym-60%	Asym-40%	Asym-60%
NULL	91.73	89.99	**90.98**	86.23
MAX	85.65	84.54	85.88	83.14
MIN	85.15	85.03	83.41	84.33
SUM	87.12	87.46	86.02	85.36
CANCAT	90.45	**90.32**	90.11	85.45
MEAN	**91.82**	90.07	90.84	**86.48**

During the Causal Collaborative Denoising phase, local representations are aggregated to form the final global representation. We evaluated several aggregation functions to assess their impact on model performance. The experimental results on the MALTLS-22 dataset are summarized in Table VII, where NULL denotes the use of the best-performing individual subset without aggregation. Overall, the model maintains stable performance across different aggregation strategies. However, aggregation using MAX or MIN functions tends to degrade performance, likely due to the loss of nuanced feature information. In contrast, MEAN and CONCAT consistently deliver strong results, with MEAN achieving a favorable balance between performance and computational efficiency. While CONCAT can offer slightly better accuracy, it incurs additional memory and computation costs. Therefore, we recommend MEAN as the default aggregation function due to its effectiveness and lower overhead in deriving final representations after Local Joint Learning.

D. Parameter Study

In this paper, we introduce the method of partitioning raw features into subsets for joint learning and collaborative denoising. This section delves into exploring the influence of the number of subsets and the degree of overlap. Fig. 7 presents the findings of this investigation. Overall, increasing the degree of overlap enhances performance. However, excessively detailed partitioning would diminish the representation capabilities of subsets, while overly coarse partitioning may lead to insufficient collaborative reasoning. For the MALTLS-22 dataset, the best results are achieved with 4 subsets and an overlap of 0.75. In contrast, for the CICIDS-2017 dataset, performance is relatively insensitive to variations in overlap when using 4 or 6 subsets. Fig. 8 illustrates the performance of subsets with varying degrees of overlap on the MALTLS-22 dataset, with the number of subsets fixed at 4. It is evident that at lower degrees of overlap, there is significant performance disparity among subsets, resulting in unstable performance.

Based on these observations, we recommend adopting a higher degree of overlap alongside a moderate partition granularity.



(a) MALTLS-22 Dataset (b) CICIDS-2017 Dataset

Fig. 7: The influence of different partition settings.

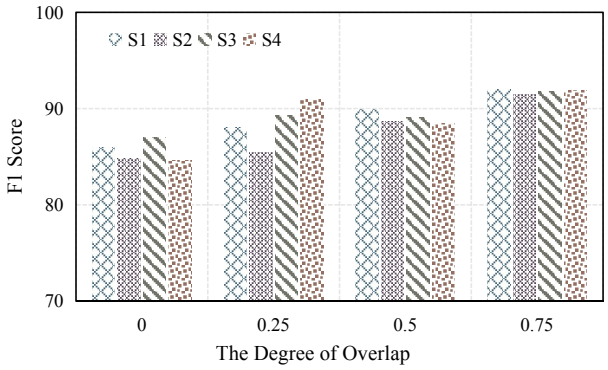


Fig. 8: The performance of subsets over varying overlaps.

E. Complexity Analysis

We analyze the computational complexity of CoLD with respect to its three primary components: Feature Reordering, Local Joint Learning, and Causal Collaborative Denoising. The complexity of Feature Reordering arises from computing pairwise correlations among features, resulting in a time complexity of $\mathcal{O}(Nd^2)$, where N is the number of samples and d is the number of features. In the Local Joint Learning stage, both the encoder and decoder are implemented as simple fully connected networks, so the per-subset complexity is $\mathcal{O}(d)$. Given M feature subsets and N samples, the total cost for this step is $\mathcal{O}(NMd)$. Additionally, pairwise similarity computations for loss terms introduce an extra $\mathcal{O}(NM^2)$ cost, making the overall complexity for this stage $\mathcal{O}(N(Md+M^2))$. For Causal Collaborative Denoising, the main computational cost comes from GMM fitting, which incurs a complexity of $\mathcal{O}(NKd)$ per EM iteration, where K is the number of mixture components.

Overall, the complexity of CoLD is primarily influenced by the number of subsets M, the number of samples N, the feature dimensionality d, and the number of mixture components K. Since Feature Reordering can be performed offline, it does not affect the computational complexity during inference. In practice, the training cost can be further reduced by using a small number of subsets (e.g., $M = 4$) and by adopting mini-batch training, which allows the computational cost to scale approximately linearly with the batch size. These design choices make CoLD computationally efficient and scalable for large-scale datasets.

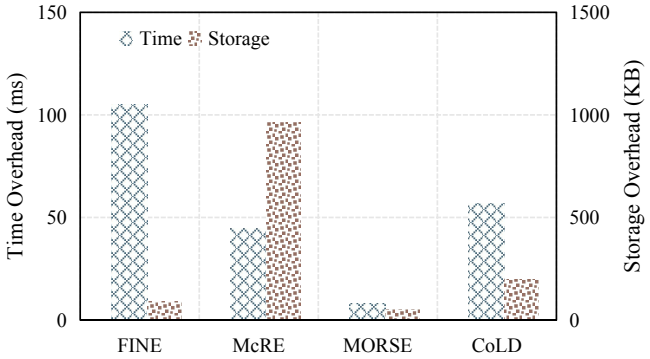


Fig. 9: Time and storage overhead of comparison methods.

We present the empirical time and storage overhead of CoLD and compare it with other methods, as shown in Fig. 9. Time overhead is reported as the average processing time per sample. The results show that CoLD incurs lower time overhead than FINE, primarily because FINE involves a more complex feature decomposition process. While MORSE demonstrates lower storage and computational requirements, its performance is notably inferior. In contrast, CoLD processes multiple feature subsets in the Local Joint Learning phase to achieve fine-grained representations, accepting a moderate increase in complexity for enhanced noise purification. Overall, CoLD's time cost is moderate and comparable to that of McRE, while its storage overhead remains significantly lower. These results highlight that CoLD achieves robust performance and noise resistance with a favorable trade-off between efficiency and accuracy, making it a practical choice for real-world applications.

VI. Evaluation in Realistic Enterprise Network

While CoLD has shown strong performance on benchmark datasets, it is crucial to further evaluate its effectiveness and robustness in real-world environments. In this section, we present the initial deployment and evaluation of CoLD in a large-scale enterprise network, aiming to assess its practical applicability and reliability.

A. Task Description

Intrusion detection systems (IDSs) are crucial for protecting enterprise networks from sophisticated attacks such as Advanced Persistent Threats (APTs) [64], which often evade traditional defenses. Provenance-based IDSs [65], [66], [67], [68] leverage contextual information from system logs to detect APTs, but their performance is challenged by unreliable ground truth and pervasive label noise [69]. In this task, we integrate CoLD with several representative enterprise-level IDSs [67], [68] and evaluate its effectiveness on the OpTC dataset [70]. This evaluation extends CoLD's applicability to realistic enterprise networks and systematically investigates

its ability to improve intrusion detection performance under various levels of label noise.

B. Comparative Discussion

A variety of IDSs have emerged in recent years to identify malicious activities on enterprise networks, leveraging network traffic and log data. In large-scale enterprise settings, network architectures are highly dynamic, involving a diverse set of active entities—including hosts, users, virtual machines, and applications—that interact in complex and evolving patterns.

To address these challenges, Jbeil [66] modeled authentication events as temporal graphs, leveraging Temporal Graph Networks for inductive learning. This approach allows for the detection of static phases of lateral movement in APT attacks as the graph structure changes dynamically. However, modules tailored for specific attack phases may prove ineffective when applied to complex real-world scenarios. Flash [67] employed Word2Vec to transform node attributes into semantically rich, time-sensitive feature vectors and then utilizes Graph Neural Networks to capture both local and global graph structures. This enables the model to effectively encode complex temporal dependencies within the provenance graph. However, Flash's approach requires a large volume of high-quality log data to ensure the completeness and reliability of the constructed provenance graphs. Argus [68] introduced a dynamic graph representation learning framework that integrates Graph Convolutional Networks with Long Short-Term Memory networks for feature extraction. By embedding timestamp information and supporting dynamic updates, Argus can track and model real-time changes in graph topology. Despite their effectiveness, these methods face persistent challenges related to the collection and annotation of enterprise-scale event data, particularly under conditions of label noise.

In contrast to the aforementioned approaches that are tailored for specific attack patterns or scenarios, CoLD serves as a more general and versatile causal collaborative denoising framework. Its primary goal is to purify noisy samples and enhance the robustness of IDSs in the presence of label noise. Although CoLD is mainly applied to network traffic in our previous experiments, its modular design and causal denoising mechanism are broadly applicable to a wide range of data modalities. In fact, CoLD can be seamlessly integrated with existing IDSs such as Flash and Argus, complementing their strengths and further improving their resilience to noisy labels. This flexibility facilitates the practical deployment of CoLD in diverse and large-scale enterprise environments.

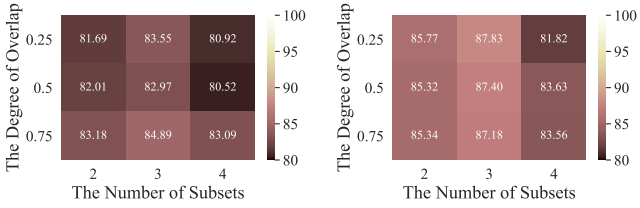
C. Evaluation Settings

To integrate CoLD with existing IDSs, we adopted a decoupled approach, dividing the enterprise-level intrusion detection task into a feature extraction module and a denoising learning module. Specifically, we utilized established provenance graph representation learning methods to extract features from enterprise network logs [70], which were then used as inputs for downstream classifiers. To meet real-time processing requirements, we selected XGBoost as a lightweight and efficient classifier. CoLD operates as a plug-in module that enhances the classifier's training process by effectively filtering out noisy samples. This modular design not only broadens CoLD's applicability but also significantly reduces resource consumption during deployment. For empirical evaluation, we conduct a comprehensive comparison between CoLD and two representative provenance-based IDSs, Flash [67] and Argus [68], using the F1-score as the evaluation metric. Specifically, we integrated CoLD into their established pipelines by preserving the original provenance graph construction and feature extraction of each system. This ensures a fair and consistent assessment of CoLD's effectiveness in enhancing intrusion detection performance under varying noise settings, while maintaining the unique characteristics and strengths of the baseline methods.

D. Evaluation Results

TABLE VIII: Results on OpTC Dataset.

Method	Sym-10%	Sym-40%	Asym-10%	Asym-40%
Falsh	93.57	79.49	94.08	85.15
Falsh+CoLD	94.04 ↑ 0.47	84.89 ↑ 5.40	94.30 ↑ 0.22	93.78 ↑ 8.63
Argus	91.45	81.81	93.94	86.28
Argus+CoLD	93.73 ↑ 2.28	87.83 ↑ 6.02	94.70 ↑ 0.76	93.51 ↑ 7.23



(a) Falsh+CoLD (b) Argus+CoLD

Fig. 10: Hyperparameter optimization of integrated systems.

Based on the analysis in Section V-D, we perform hyperparameter optimization for CoLD within an appropriate range under the Sym-40% setting, with the results shown in Fig. 10.. The comparative performance of CoLD on the enterprise dataset under various noise settings is shown in Table VIII. From these results, we draw the following observations and conclusions:

(1) As label noise increases, the performance of baseline IDSs is notably challenged. However, integrating CoLD consistently leads to significant improvements. For instance, at a symmetric noise level of 40%, CoLD boosts the performance of Flash and Argus by 5.40% and 6.02%, respectively, demonstrating its effective denoising capability in enterprise-level detection scenarios.

(2) Flash exhibits superior performance under low noise conditions compared with Argus, as it incorporates more features to represent graph nodes, such as process names, command line arguments, file paths, and IP addresses. Nevertheless, as noise levels rise, Flash's performance degrades

more rapidly, largely because many of its features lack class discriminability and are more susceptible to being misled by local consistency.

(3) Argus demonstrates greater robustness to high levels of label noise. Its use of lower-dimensional feature representations tends to favor smaller, more overlapping feature subsets, which helps maintain stability. In contrast, due to the limited quality of the dataset, many features leveraged by Flash are sparse, requiring a larger overlap to preserve the representational capability for sample discrimination.

Overall, these results confirm the effectiveness of CoLD in real-world enterprise environments and demonstrate its ability to enhance the noise resilience of existing IDSs.

VII. Discussion

A. Application in Online/Streaming Network Environments

Online or streaming IDSs are required to process large volumes of network data in real time while adapting to continuously evolving behavior patterns. The modular architecture of CoLD makes it well-suited for extension to such scenarios in real-world deployments. Specifically, the Feature Reordering module supports both offline computation and periodic updates, whereas the Local Joint Learning and Causal Collaborative Denoising modules can be incrementally updated as new data arrives. Leveraging a sliding window technology, CoLD efficiently updates its model parameters on streaming data, thus maintaining computational efficiency and low latency. As the underlying data distribution shifts, the encoder, decoder, and classifier can be periodically fine-tuned using continual learning techniques, thereby preventing catastrophic forgetting and ensuring sustained performance. Furthermore, as discussed in Section VI, CoLD can be seamlessly integrated as a denoising module within existing online IDS pipelines, filtering incoming training data streams before model updates. This design enables the IDS to remain robust against evolving attack strategies and dynamic label noise, making CoLD a practical and adaptive solution for real-world intrusion detection systems operating in dynamic environments.

B. Behavior of CoLD across Different Datasets

In this paper, we conduct a comprehensive evaluation of CoLD across multiple representative datasets to provide insights into its generalization and behavior under diverse real-world network settings. Specifically, we evaluate CoLD on MALTLS-22, which consists of encrypted traffic and a wide range of attack types, and CICIDS-2017, which covers both benign and malicious behaviors across multiple protocols. These datasets embody key practical challenges such as traffic encryption, attack diversity, and severe class imbalance.

Across all noise settings and datasets, CoLD consistently demonstrates robust performance and significantly outperforms baseline methods. Notably, its advantages are especially pronounced on MALTLS-22, highlighting CoLD's ability to disrupt misleading local consistency and capture fine-grained representations, which are essential for handling high feature redundancy and subtle inter-class differences. Ablation studies confirm that all components of CoLD contribute to performance gains, with Feature Reordering being particularly impactful on CICIDS-2017, where the original feature arrangement shows lower correlation. Parameter analysis indicates that while CoLD prefers different parameter settings across datasets, it maintains strong robustness to parameter variations under moderate partition granularities. We also observe that factors such as feature dimensionality and sparsity should be considered when selecting feature subset size and overlap degree for optimal results.

Beyond these two benchmarks, our evaluation on the OpTC dataset (Section VI) further demonstrates CoLD's adaptability and effectiveness in realistic enterprise network environments. Overall, these results confirm that the collaborative causal denoising mechanisms in CoLD are generalizable, enabling the framework to adapt effectively to heterogeneous data modalities and address the diverse and complex challenges of real-world network intrusion detection.

C. Limitation and Future Work

First, we currently employ Pearson correlation to reorder features, which primarily captures linear statistical relationships between features. Future work should explore more advanced techniques to capture complex, nonlinear, or higher-order dependencies among features. Second, while CoLD achieves robust representations, it introduces additional computational complexity during the joint learning and denoising process. Improving the framework's efficiency will be a focus of subsequent research. Third, integrating CoLD with incremental or continual learning strategies represents a promising direction to further enhance its adaptability to evolving network environments and streaming data. We leave these improvements as important avenues for future work.

VIII. Conclusion

In this paper, we presented CoLD, a **Co**llaborative **L**abel **D**enoising framework designed to address the challenges posed by noisy labels in network intrusion detection. Leveraging causal analysis, we identify the impact of noisy labels and attribute it to local consistency across different categories in network traffic. Based on these insights, CoLD integrates three key components: Feature Reordering organizes features based on their statistical properties to preserve correlation, Local Joint Learning utilizes multiple views to learn robust representations that are less susceptible to label noise, and Causal Collaborative Denoising employs Gaussian Mixture Model to identify and filter mislabeled traffic flows, resulting in a purified dataset for training classifiers. Extensive experiments on benchmark datasets demonstrate that CoLD significantly improves classification performance and robustness to label noise, outperforming state-of-the-art approaches. These findings underscore the potential of CoLD to enhance the performance of IDSs in noisy environments, paving the way for more reliable and secure network infrastructures.

IX. Ethics Considerations

This research aims to advance the field of network intrusion detection by addressing the challenges posed by noisy labels. The proposed framework, CoLD, is designed to improve the accuracy and robustness of intrusion detection systems, contributing to enhanced cybersecurity for organizations and individuals. However, we acknowledge that such advancements can have dual-use implications. While the technology is intended to strengthen defenses against malicious activities, it could potentially be misused to evade detection by malicious actors. To mitigate this risk, we emphasize that this research is strictly intended for defensive applications, and we advocate for its deployment in ethically governed environments.

Additionally, the datasets used in this study are publicly available and do not contain personally identifiable information (PII). We ensure compliance with all relevant data privacy regulations. By using anonymized and publicly accessible datasets, we minimize the potential for harm or misuse of sensitive information.

Acknowledgment

This work was supported by the UGC General Research Fund no. 17209822 and the Innovation and Technology Commission Fund no. ITS/383/23FP from Hong Kong.

References

[1] M. R. Ayyagari, N. Kesswani, M. Kumar, and K. Kumar, "Intrusion detection techniques in network environment: a systematic review," *Wireless Networks*, vol. 27, no. 2, pp. 1269–1285, 2021.

[2] H. Tahaei, F. Afifi, A. Asemi, F. Zaki, and N. B. Anuar, "The rise of traffic classification in iot networks: A survey," *Journal of Network and Computer Applications*, vol. 154, p. 102538, 2020.

[3] Y. Chen, Q. Yin, Q. Li, Z. Liu, K. Xu, Y. Xu, M. Xu, Z. Liu, and J. Wu, "Learning with semantics: Towards a Semantics-Aware routing anomaly detection system," in *33rd USENIX Security Symposium (USENIX Security 24)*. Philadelphia, PA: USENIX Association, Aug. 2024, pp. 5143–5160. [Online]. Available: https://www.usenix.org/conference/usenixsecurity24/presentation/chen-yihao

[4] N. Wang, Y. Chen, Y. Hu, W. Lou, and Y. T. Hou, "Feco: Boosting intrusion detection capability in iot networks via contrastive learning," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2022, pp. 1409–1418.

[5] A. F. Diallo and P. Patras, "Adaptive clustering-based malicious traffic classification at the network edge," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[6] Y. Yue, X. Chen, Z. Han, X. Zeng, and Y. Zhu, "Contrastive learning enhanced intrusion detection," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4232–4247, 2022.

[7] D. Chou and M. Jiang, "A survey on data-driven network intrusion detection," *ACM Computing Surveys (CSUR)*, vol. 54, no. 9, pp. 1–36, 2021.

[8] S. Yang, X. Zheng, Z. Xu, and X. Wang, "A lightweight approach for network intrusion detection based on self-knowledge distillation," in *ICC 2023-IEEE International Conference on Communications*. IEEE, 2023, pp. 3000–3005.

[9] Y. Chen, Q. Yin, Q. Li, Z. Liu, K. Xu, Y. Xu, M. Xu, Z. Liu, and J. Wu, "Learning with semantics: Towards a Semantics-Aware routing anomaly detection system," in *33rd USENIX Security Symposium (USENIX Security 24)*. Philadelphia, PA: USENIX Association, Aug. 2024, pp. 5143–5160. [Online]. Available: https://www.usenix.org/conference/usenixsecurity24/presentation/chen-yihao

[10] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE transactions on neural networks and learning systems*, vol. 34, no. 11, pp. 8135–8153, 2022.

[11] D. Arp, E. Quiring, F. Pendlebury, A. Warnecke, F. Pierazzi, C. Wressnegger, L. Cavallaro, and K. Rieck, "Dos and don'ts of machine learning in computer security," in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 3971–3988. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/arp

[12] J. L. Guerra, C. Catania, and E. Veas, "Datasets are not enough: Challenges in labeling network traffic," *Computers & Security*, vol. 120, p. 102810, 2022.

[13] Q. Yuan, G. Gou, Y. Zhu, Y. Zhu, G. Xiong, and Y. Wang, "Mcre: A unified framework for handling malicious traffic with noise labels based on multidimensional constraint representation," *IEEE Transactions on Information Forensics and Security*, 2023.

[14] G. Engelen, V. Rimmer, and W. Joosen, "Troubleshooting an intrusion detection dataset: the cicids2017 case study," in *2021 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2021, pp. 7–12.

[15] C. Fu, Q. Li, and K. Xu, "Detecting unknown encrypted malicious traffic in real time via flow interaction graph analysis," *arXiv preprint arXiv:2301.13686*, 2023.

[16] S. Yang, X. Zheng, J. Li, J. Xu, and E. C. H. Ngai, "Multi-scale contrastive attention representation learning for encrypted traffic classification," in *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 2024, p. 4173–4177.

[17] Y. Chen, Z. Ding, and D. Wagner, "Continuous learning for android malware detection," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 1127–1144.

[18] X. Wu, W. Guo, J. Yan, B. Coskun, and X. Xing, "From grim reality to practical solution: Malware classification in real-world noise," in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 2602–2619.

[19] F. Dong, L. Wang, X. Nie, F. Shao, H. Wang, D. Li, X. Luo, and X. Xiao, "DISTDET: A Cost-Effective distributed cyber threat detection system," in *32nd USENIX Security Symposium (USENIX Security 23)*. Anaheim, CA: USENIX Association, Aug. 2023, pp. 6575–6592. [Online]. Available: https://www.usenix.org/conference/usenixsecurity23/presentation/dong-feng

[20] W. U. Hassan, S. Guo, D. Li, Z. Chen, K. Jee, Z. Li, and A. Bates, "Nodoze: Combatting threat alert fatigue with automated provenance triage," in *network and distributed systems security symposium*, 2019.

[21] Z. Huang, J. Zhang, and H. Shan, "Twin contrastive learning with noisy labels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 11 661–11 670.

[22] A. Ghosh, H. Kumar, and P. S. Sastry, "Robust loss functions under label noise for deep neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, no. 1, 2017.

[23] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 322–330.

[24] A. Vahdat, "Toward robustness against label noise in training deep discriminative neural networks," *Advances in neural information processing systems*, vol. 30, 2017.

[25] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *International conference on machine learning*. PMLR, 2018, pp. 2304–2313.

[26] I. Jindal, M. Nokleby, and X. Chen, "Learning deep networks from noisy labels with dropout regularization," in *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 2016, pp. 967–972.

[27] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. W. Tsang, and M. Sugiyama, "Co-teaching: robust training of deep neural networks with extremely noisy labels," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 8536–8546.

[28] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, and M. Sugiyama, "How does disagreement help generalization against label corruption?" in *International conference on machine learning*. PMLR, 2019, pp. 7164–7173.

[29] C. Liu, H. Yu, B. Li, Z. Shen, Z. Gao, P. Ren, X. Xie, L. Cui, and C. Miao, "Noise-resistant deep metric learning with ranking-based instance selection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6811–6820.

[30] X. Xia, T. Liu, B. Han, M. Gong, J. Yu, G. Niu, and M. Sugiyama, "Sample selection with uncertainty of losses for learning with noisy labels," *arXiv preprint arXiv:2106.00445*, 2021.

[31] T. Kim, J. Ko, J. Choi, S.-Y. Yun *et al.*, "Fine samples for learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 137–24 149, 2021.

[32] Y. Li, H. Han, S. Shan, and X. Chen, "Disc: Learning from noisy labels via dynamic instance-specific selection and correction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 24 070–24 079.

[33] Z. Zhao, R. Birke, R. Han, B. Robu, S. Bouchenak, S. B. Mokhtar, and L. Y. Chen, "Enhancing robustness of on-line learning models on highly noisy data," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2177–2192, 2021.

[34] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," in *International conference on machine learning*. PMLR, 2020, pp. 6543–6553.

[35] S. Liu, J. Niles-Weed, N. Razavian, and C. Fernandez-Granda, "Early-learning regularization prevents memorization of noisy labels," *Advances in neural information processing systems*, vol. 33, pp. 20 331–20 342, 2020.

[36] E. Malach and S. Shalev-Shwartz, "Decoupling" when to update" from" how to update"," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 961–971.

[37] C. Tan, J. Xia, L. Wu, and S. Z. Li, "Co-learning: Learning from noisy labels with self-supervision," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 1405–1413.

[38] J. Goldberger and E. Ben-Reuven, "Training deep neural-networks using a noise adaptation layer," in *International conference on learning representations*, 2022.

[39] R. Tanno, A. Saeedi, S. Sankaranarayanan, D. C. Alexander, and N. Silberman, "Learning from noisy labels by regularized estimation of annotator confusion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 11 244–11 253.

[40] X. Xia, T. Liu, N. Wang, B. Han, C. Gong, G. Niu, and M. Sugiyama, "Are anchor points really indispensable in label-noise learning?" *Advances in neural information processing systems*, vol. 32, 2019.

[41] E. Arazo, D. Ortego, P. Albert, N. O'Connor, and K. McGuinness, "Unsupervised label noise modeling and loss correction," in *International conference on machine learning*. PMLR, 2019, pp. 312–321.

[42] K. Yi and J. Wu, "Probabilistic end-to-end noise correction for learning with noisy labels," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 7017–7025.

[43] S. Zheng, P. Wu, A. Goswami, M. Goswami, D. Metaxas, and C. Chen, "Error-bounded correction of noisy labels," in *International Conference on Machine Learning*. PMLR, 2020, pp. 11 447–11 457.

[44] C. Northcutt, L. Jiang, and I. Chuang, "Confident learning: Estimating uncertainty in dataset labels," *Journal of Artificial Intelligence Research*, vol. 70, pp. 1373–1411, 2021.

[45] G. Pleiss, T. Zhang, E. Elenberg, and K. Q. Weinberger, "Identifying mislabeled data using the area under the margin ranking," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 044–17 056, 2020.

[46] Z. Zhu, Z. Dong, and Y. Liu, "Detecting corrupted labels without training a model to predict," in *International conference on machine learning*. PMLR, 2022, pp. 27 412–27 427.

[47] K. Fauvel, F. Chen, and D. Rossi, "A lightweight, efficient and explainable-by-design convolutional neural network for internet traffic classification," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 4013–4023.

[48] C. Qiu, Y. Geng, J. Lu, K. Chen, S. Zhu, Y. Su, G. Nan, C. Zhang, J. Fu, Q. Cui *et al.*, "3d-ids: Doubly disentangled dynamic intrusion detection," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 1965–1977.

[49] X. Zheng, S. Yang, and X. Wang, "Sf-ids: An imbalanced semi-supervised learning framework for fine-grained intrusion detection," in *ICC 2023-IEEE International Conference on Communications*. IEEE, 2023, pp. 2988–2993.

[50] 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 *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 2327–2344.

[51] S. Yang, X. Zheng, J. Li, J. Xu, X. Wang, and E. C. H. Ngai, "Recda: Concept drift adaptation with representation enhancement for network intrusion detection," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, p. 3818–3828.

[52] G. Andresini, F. Pendlebury, F. Pierazzi, C. Loglisci, A. Appice, and L. Cavallaro, "Insomnia: Towards concept-drift robustness in network intrusion detection," in *Proceedings of the 14th ACM workshop on artificial intelligence and security*, 2021, pp. 111–122.

[53] X. Wang, "Enidrift: A fast and adaptive ensemble system for network intrusion detection under real-world drift," in *Proceedings of the 38th Annual Computer Security Applications Conference*, 2022, pp. 785–798.

[54] S. Yang, X. Zheng, J. Li, J. Xu, X. Zhang, and E. C. H. Ngai, "Self-supervised adaptation method to concept drift for network intrusion detection," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–15, 2025.

[55] Y. Qing, Q. Yin, X. Deng, Y. Chen, Z. Liu, K. Sun, K. Xu, J. Zhang, and Q. Li, "Low-quality training data only? a robust framework for detecting encrypted malicious network traffic," *arXiv preprint arXiv:2309.04798*, 2023.

[56] X. Wu, B. Jiang, K. Yu, H. Chen, and C. Miao, "Multi-label causal feature selection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 04, 2020, pp. 6430–6437.

[57] Y. Sui, X. Wang, J. Wu, M. Lin, X. He, and T.-S. Chua, "Causal attention for interpretable and generalizable graph classification," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2022, pp. 1696–1705.

[58] F. Zhou, Y. Mao, L. Yu, Y. Yang, and T. Zhong, "Causal-debias: Unifying debiasing in pretrained language models and fine-tuning via causal invariant learning," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 4227–4241.

[59] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, and T. A. Mann, "Data augmentation can improve robustness," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29 935–29 948, 2021.

[60] F. Iglesias and T. Zseby, "Analysis of network traffic features for anomaly detection," *Machine Learning*, vol. 101, no. 1, pp. 59–84, 2015.

[61] F. Gottwalt, E. Chang, and T. Dillon, "Corrcorr: A feature selection method for multivariate correlation network anomaly detection techniques," *Computers & Security*, vol. 83, pp. 234–245, 2019.

[62] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[63] M. Dragoi, E. Burceanu, E. Haller, A. Manolache, and F. Brad, "Anoshift: A distribution shift benchmark for unsupervised anomaly detection," *Advances in Neural Information Processing Systems*, vol. 35, pp. 32 854–32 867, 2022.

[64] K. A. Akbar, Y. Wang, G. Ayoade, Y. Gao, A. Singhal, L. Khan, B. Thuraisingham, and K. Jee, "Advanced persistent threat detection using data provenance and metric learning," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 5, pp. 3957–3969, 2022.

[65] M. Zipperle, F. Gottwalt, E. Chang, and T. Dillon, "Provenance-based intrusion detection systems: A survey," *ACM Computing Surveys*, vol. 55, no. 7, pp. 1–36, 2022.

[66] J. Khoury, . Klisura, H. Zanddizari, G. D. L. T. Parra, P. Najafirad, and E. Bou-Harb, "Jbeil: Temporal graph-based inductive learning to infer lateral movement in evolving enterprise networks," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024, pp. 3644–3660.

[67] M. U. Rehman, H. Ahmadi, and W. U. Hassan, "Flash: A comprehensive approach to intrusion detection via provenance graph representation learning," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024, pp. 3552–3570.

[68] J. Xu, X. Shu, and Z. Li, "Understanding and bridging the gap between unsupervised network representation learning and security analytics," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024, pp. 3590–3608.

[69] J. Liu, M. A. Inam, A. Goyal, A. Riddle, K. Westfall, and A. Bates, "What we talk about when we talk about logs: Understanding the effects of dataset quality on endpoint threat detection research," in *2025 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2025, pp. 112–129.

[70] F. Directions, "Operationally transparent cyber (optc) data release," 2020. [Online]. Available: https://github.com/FiveDirections/OpTC-data