Exploiting Self-Refining Normal Graph Structures for Robust Defense against Unsupervised Adversarial Attacks

Bingdao Feng[1] , **Di Jin**[1,2,*] , **Xiaobao Wang**[1,3] , **Dongxiao He**[1] , **Jingyi Cao**[1] and **Zhen Wang**[4]
[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]Key Laboratory of Artificial Intelligence Application Technology, Qinghai Minzu University, Xining, 810007, China
[3]Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen, China
[4]School of Cybersecurity, Northwestern Polytechnical University, Xi'an, Shaanxi, China
{fengbingdao, jindi, wangxiaobao, hedongxiao, caojingyi}@tju.edu.cn, w-zhen@nwpu.edu.cn

Abstract

Defending against adversarial attacks on graphs has become increasingly important. Graph refinement to enhance the quality and robustness of representation learning is a critical area that requires thorough investigation. We observe that representations learned from attacked graphs are often ineffective for refinement due to perturbations that cause the endpoints of perturbed edges to become more similar, complicating the defender's ability to distinguish them. To address this challenge, we propose a robust unsupervised graph learning framework that utilizes cleaner graphs to learn effective representations. Specifically, we introduce an anomaly detection model based on contrastive learning to obtain a rough graph excluding a large number of perturbed structures. Subsequently, we then propose the *Graph Pollution Degree (GPD)*, a mutual information-based measure that leverages the encoder's representation capability on the rough graph to assess the trustworthiness of the predicted graph and refine the learned representations. Extensive experiments on four benchmark datasets demonstrate that our method outperforms nine state-of-the-art defense models, effectively defending against adversarial attacks and enhancing node classification performance.

1 Introduction

Graphs are a ubiquitous form of data structure capable of representing a diverse array of entities and their complex relationships [Zhang *et al.*, 2020]. Graphs can mirror various real-world networks, including protein networks [Vlaic *et al.*, 2018], traffic networks [Huang *et al.*, 2022], social networks [Chang *et al.*, 2023], and Textual Networks [Wang *et al.*, 2025]. Graph Neural Networks (GNNs) [Hamilton *et al.*, 2017; Kipf and Welling, 2017] have emerged as a powerful tool for graph representation, attracting significant attention due to their remarkable performance in tasks such as node classification [Yan *et al.*, 2025].

*Corresponding author

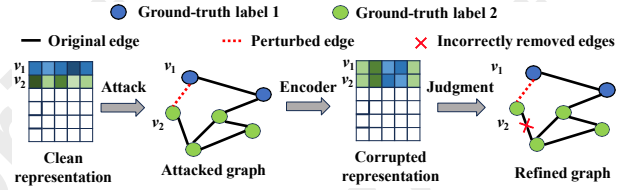


Figure 1: Overview of using corrupted representations to detect perturbed edges. The adversary adds an edge between v_1 and v_2 based on the clean representation, maximizing the disruption to the graph's representation. The defender, trained on the attacked graph, has difficulty correctly removing the perturbed edges because the message passing process causes the two nodes to become more similar.

While supervised learning remains prevalent in practical applications, labeling large volumes of graph data can be costly and prone to errors. Consequently, there have been notable advancements in unsupervised algorithms over the years [Veličković *et al.*, 2019; Zhu *et al.*, 2021]. These methods focus on learning an encoder from unlabeled graph data, which can be used to generate representations for downstream tasks.

Despite their advantages in many tasks, unsupervised learning models are generally more vulnerable to adversarial attacks than supervised models due to the absence of labeled data. Supervised models use labeled data to correct predictions and learn more robust representations, whereas unsupervised models lack this mechanism, making them more susceptible to structural perturbations [Xu *et al.*, 2022; Jin *et al.*, 2023]. Adversarial attacks introduce small perturbations to the graph structure, which can significantly alter prediction outcomes [Madry *et al.*, 2018; Zügner *et al.*, 2019; Sun *et al.*, 2022; Zhu *et al.*, 2024; He *et al.*, 2025].

Existing research primarily focuses on defending against evasion attacks by training robust representations to withstand adversarial attacks during the inference phase [Zhuang and Al Hasan, 2022; Feng *et al.*, 2024b]. However, compared to evasion attacks, poisoning attacks are more destructive because they directly alter the model during training, while evasion attacks only degrade performance by modifying the structure around target nodes [Li *et al.*, 2023]. Research on defenses against poisoning attacks remains limited. Common defense strategies often rely on the homophily

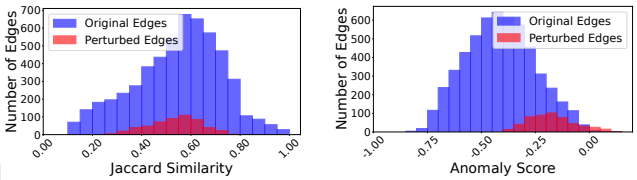
assumption, which seeks to remove edges between nodes with dissimilar representations [Zhang and Zitnik, 2020; Zhang *et al.*, 2019]. However, due to the presence of perturbed edges, the endpoints of these edges are influenced by the perturbations, causing their representations to become more similar and, consequently, harder to detect. In other words, the large number of normal edges causes the effect of a few perturbed edges to be diluted, resulting in similar similarity distributions for both perturbed and original edges. To illustrate, we design a simple process for detecting adversarial perturbations based on similarity as shown in Figure 1. The adversary aims to maximize disruption by connecting distant nodes, v_1 and v_2, with a perturbed edge. However, the perturbation causes their representations to become more similar, making it difficult to distinguish between them. To verify this, we examine the similarity distribution of perturbed edges. As shown in Figure 2(a), the similarity distributions of normal and perturbed edges are highly similar, which makes it challenging for similarity-based detection methods to effectively distinguish the perturbed edges.

Currently, several self-supervised methods based on contrastive learning have been developed to detect anomalous structures in graphs, where anomalies are defined as edge connection patterns or the existence of edges that significantly deviate from typical graph structures, such as edges connecting distant nodes or altering the inherent properties of nodes [Peng *et al.*, 2018; Liu *et al.*, 2021; Duan *et al.*, 2023]. Unlike similarity-based detection methods, which are influenced by information propagation, these methods learn anomaly patterns by capturing the normal patterns between a large number of nodes and their neighbors (e.g., the homophily assumption), without being affected by information propagation. Can anomaly detection methods be leveraged to better identify perturbed edges? Figure 2(b) shows the distribution of anomaly edge scores on the Cora dataset computed by CoLA. We find that most normal edges can be reliably trusted; however, some normal edges with higher anomaly scores overlap with perturbed edges, making it difficult to distinguish between them. This raises the following challenge:

How to effectively utilize the trusted normal edges to distinguish those within the overlapping region and accurately identify perturbed edges.

To address this challenge, we first apply anomaly detection techniques to generate a preliminary graph composed of "true" (trusted) edges. In the next step, we employ an information-theoretic measure called Graph Pollution Degree (GPD) to quantitatively assess the purity of the predicted graph. This measure allows us to identify candidate edges that remain perturbed, pinpointing clean subgraphs that are suitable for model training. Through this two-stage process, we iteratively refine the graph and fine-tune the graph representation learning model, ultimately achieving effective representations. In summary, the main contributions of this work are as follows:

- We propose a robust unsupervised graph representation learning framework that leverages a clean graph to iteratively refine the graph structure for effective representation learning.



(a) Representation similarity (b) Anomaly detection

Figure 2: (a) The representation similarity for each edge is calculated using Jaccard Similarity, with DGI trained on the attacked graph. (b) The anomaly scores for each edge are computed using CoLA. Both are based on attacked graphs generated by PGD on the Cora dataset.

- We propose a novel graph refinement process that leverages anomaly detection to identify perturbed edges and utilizes our designed information-theoretic measure, Graph Pollution Degree (GPD), to infer the clean graph by focusing on edges most likely to be "true".
- Extensive experiments on four real-world datasets demonstrate that our model defends against various types of attacks and outperform nine state-of-the-art defense models.

2 Preliminaries

2.1 Graph Representation Learning

We are provided with an attribute graph $G = \{V, A, X\}$, where $V = \{v_1, v_2, \ldots, v_n\}$ represents the set of nodes in the graph, $A \in \mathbb{R}^{N\times N}$ denotes the adjacency matrix of the graph G, and $X \in \mathbb{R}^{N\times d}$ is the node feature matrix, where each row x_i corresponds to the d-dimensional feature vector of node v_i. In the adjacency matrix A, the element $A_{i,j} \in \{0, 1\}$ indicates whether there is an edge between nodes v_i and v_j: if an edge exists, the value is 1; otherwise, it is 0. The objective of the graph representation learning task is to train an encoder $e : \mathbb{R}^d \to \mathbb{R}^{d'}$ and obtain high-level representations $H_i \in \mathbb{R}^{d'}$ for every node v_i.

A lot of studies [Veličković *et al.*, 2019; Sun *et al.*, 2019; Qiu *et al.*, 2020] train a graph encoder to maximize the mutual information between the node representations and the global graph representation in the graph G. Since the marginal distribution is not estimated, optimizing mutual information directly can be challenging, these methods transform the problem into a standard binary cross-entropy loss between positive and negative examples:

$$l = \frac{1}{N+M}\left\{\sum_{i=1}^{N}\mathbb{E}_G \log D(h_i, s) + \sum_{j=1}^{M}\mathbb{E}_{\widetilde{G}} \log\left[1 - D(\widehat{h}_j, s)\right]\right\}, \quad (1)$$

where N and M respectively represent the number of positive and negative samples, h_i and h_j respectively denote the representations of positive G and negative $\widetilde{G}$ samples, encoded by the encoder e, s represents graph-level summary vectors,

and $D(\cdot)$ represents a discriminator composed of a bilinear scoring function:

$$D(h_i, s) = \sigma(h^{\mathrm{T}} W s),$$

where W represents the scoring matrix that needs to be learned in the discriminator, and $\sigma(\cdot)$ represents the logistic sigmoid function.

2.2 Adversarial Attacks and Defense

Adversarial attacks [Madry *et al.*, 2018; Bojchevski and Günnemann, 2019] on graphs may involve alterations to both the graph structure and node attributes, potentially leading to failures in tasks such as node classification, graph classification, and more. Essentially, adversarial attacks can be mathematically expressed as a bi-level attack form:

$$\begin{array}{l} \max\limits_{G_{\text{attack}} \in \varphi(G)} l\left(f_{\theta^*}(G_{\text{attack}})\right) \\ \text{s.t.} \theta^* = \arg\min\limits_{\theta} l\left(f_\theta(G)\right) \end{array}, \tag{2}$$

where $\varphi(G)$ denotes the space of perturbation, f_θ is the surrogate model, and l is the loss associated with the surrogate model, such as cross-entropy or contrastive loss.

Depending on whether the attackers generate adversarial samples during the training or inference phase, attacks can be categorized into poisoning attacks and evasion attacks. In response to poisoning attacks, a substantial body of research has adopted attack detection methods, aiming to proactively identify and eliminate malicious nodes and edges in graph data [Li *et al.*, 2022; Wu *et al.*, 2019; Ding *et al.*, 2019].

Graph-based anomaly detection aims to identify abnormal patterns that deviate from the majority of the data, particularly anomalous structures. Adversarial attacks can be considered a distinct and dangerous scenario of anomalies [Abusnaina *et al.*, 2021; Ma *et al.*, 2021; Wang *et al.*, 2023]. In this paper, we propose a graph refinement method based on the CoLA [Liu *et al.*, 2021] framework, recognized as the state-of-the-art in contrastive learning for unsupervised anomaly detection. Our approach predicts the anomaly score of a node by calculating the contrastive instance pair labels between the node v_i and its corresponding sub-graph and thus obtains a preliminarily filtered rough graph.

3 Methods

In this section, we introduce a robust representation learning framework composed of two main components: normality analysis and robust representation learning. As illustrated in Figure 3, we initially employ graph anomaly detection methods to ascertain the anomaly scores of each edge, identifying potential adversarial structures. Subsequently, we implement a progressive refinement process that iteratively updates the graph and fine-tunes the model, thereby transforming corrupted representations into effective ones and enabling the learning of robust graph representations.

3.1 Anomaly Studying

We present methods for obtaining a rough graph that contains a significantly lower proportion of perturbed edges. This process involves two main steps: computing anomaly scores for nodes and constructing the rough graph based on these scores.

Anomaly Score Learning

Adversarial attacks often connect two unrelated and distant nodes, resulting in inconsistency between the attributes of a node and its neighbors, which causes these nodes to appear anomalous. Therefore, we compute an anomaly score for each node to identify these anomalies caused by adversarial attacks. Inspired by [Liu *et al.*, 2021], we model the anomaly patterns using contrastive self-supervised learning. Specifically, in each epoch, we randomly select a target node i, and adopt the random walk with restart (RWR) method to sample positive examples associated with i, represented as the positive subgraph G_i^+. First, we compute the embeddings H_i of the nodes in the subgraph G_i^+. To achieve this, we aggregate information from each node's local neighborhood to capture both structural and attribute information. For this purpose, we use a Graph Convolutional Network (GCN), as it effectively learns representations by combining features from neighboring nodes. The embeddings $H_i^{(l)}$ at layer l are computed as:

$$H_i^l = \varphi(D_i^{-\frac{1}{2}} A_i D_i^{-\frac{1}{2}} H_i^{(l-1)} W^{(l-1)}), \tag{3}$$

where D_i represents the degree of node i, and A_i represents the adjacency matrix of the subgraph to which node i belongs. W denotes the learnable parameters, and $\varphi(\cdot)$ is the ReLU activation function. Then, we apply an average pooling function as the readout function to obtain the overall embedding h_i^g of the subgraph G_i.

$$h_i^g = \sum_{k=1}^{K} \frac{(H_k)}{K}, \tag{4}$$

where k represents the k-th node in G_i^+, and K represents the number of nodes in G_i^+.

Next, we calculate the representation of node i by mapping it into the same embedding space as the subgraph embedding h_i^g. Specifically, we use a deep neural network (DNN) to compute the embedding of node i as:

$$h_i^{(l)} = \varphi(h_i^{(l-1)} W^{(l-1)}), \tag{5}$$

where $h_i^{(l-1)}$ is the representation of node i at layaer $l-1$, and W represents the corresponding learnable parameters. Based on the embeddings of node i and the subgraph G_i^+, we predict the anomaly score of node i. We adopt a bilinear scoring function defined as:

$$s_i^+ = \sigma(h_i^g W h_i), \tag{6}$$

where W is a learnable weight matrix, and $\sigma(\cdot)$ is the sigmoid activation function. For positive instance pairs, the score s_i^+ should be close to 1. Learning exclusively from positive instance pairs can result in model collapse [Zhuang *et al.*, 2024]. To address this issue, we introduce the score s_i^- for negative instance pairs. The negative instance pairs G_i^- are obtained by randomly selecting positive subgraphs from other nodes. Consequently, s_i^- should be close to 0 and is defined as:

$$s_i^- = \sigma(h_j^g W h_i), \tag{7}$$

where j represents any random node other than node i. During the testing phase, the anomaly score of node i is defined

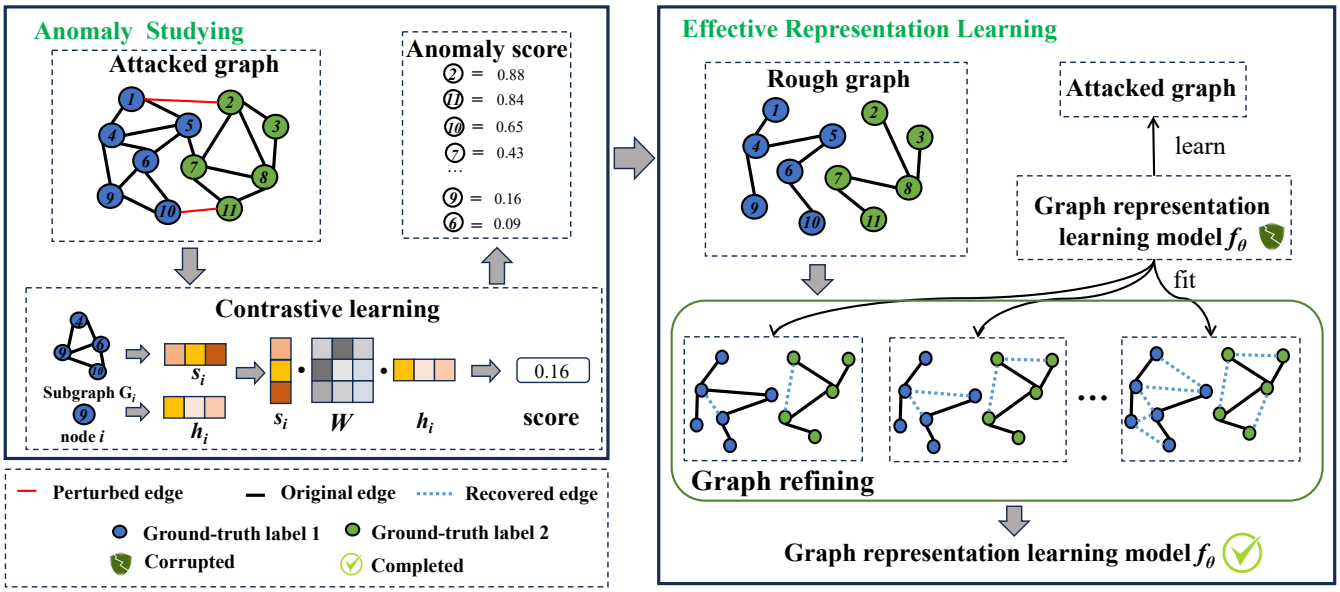


Figure 3: The overall framework of effective representation learning, consists of two modules: Anomaly Studying and Effective Representation Learning. First, the anomaly score for each node is computed to identify anomalous edges, and the top-k anomalous edges are pruned based on similarity to obtain a rough graph. Then, the rough graph is refined step by step, while the representation learning model is fine-tuned.

by the difference between the scores from the negative and positive instance pairs:

$$s_i = -(s_i^+ - s_i^-). \tag{8}$$

A higher value of s_i suggests that node i is more likely to be anomalous.

Obtaining Rough Graph

We first develop a method to exclude most of the perturbed edges from the graph. Previous research shows that adversarial attacks often connect distantly located nodes to disrupt the graph's homogeneity, typically resulting in high anomaly scores for the two connected endpoints [Tang *et al.*, 2020; Jin *et al.*, 2023; Li *et al.*, 2023]. Therefore, if the anomaly scores of the endpoints of an edge are high, it suggests a higher probability that the edge has been perturbed. In practice, we rely on the anomaly scores of the endpoints to derive the edge anomaly matrix $E^*_{i,j}$, defined as:

$$E^*_{i,j} = \begin{cases} s_i + s_j, \ if \ A_{i,j} = 1 \\ 0, \ \text{otherwise} \end{cases}. \tag{9}$$

We sort the edges according to their anomaly scores in descending order and remove the top-k most anomalous edges to generate the rough graph G. Although this edge removal strategy does not entirely eliminate all perturbed edges, it significantly mitigates the influence of most "false" edges, thereby increasing the proportion of original edges in G.

In addition, averaging the anomaly scores of the endpoints is a straightforward way to handle results across multiple rounds. Theoretically, more statistical methods could be employed, such as using the minimum or maximum values. However, since perturbed edges tend to exhibit different distribution characteristics, we believe that averaging remains an effective solution.

3.2 Effective Representation Learning

Although we obtain a cleaner graph in the previous section, this comes at the cost of discarding many "true" edges that are mistakenly identified as "false", thereby reducing the richness of the graph's structural information. To address this issue, we propose a method in this section to progressively expand the rough graph to include more "true" edges while refining the corrupted representations into effective ones. This method consists of two key components: the establishment of the *Graph Pollution Degree* (GPD) and the overall framework for obtaining effective representations.

Quantifying Graph Pollution Degree

We propose a method to ascertain whether the edges in a predicted graph have been perturbed by measuring changes in mutual information. Intuitively, the aim of adversarial attacks on graphs is to damage the representations, thereby reducing the performance of downstream tasks. This indicates that representations learned from clean graphs do not adequately reflect graphs containing adversarial edges. Therefore, we propose the Graph Pollution Degree (GPD) as a measure to quantify the level of pollution in the predicted graph.

$$GPD(\theta) = I\left(G, f_\theta\left(G\right)\right) - I\left(G', f_\theta\left(G'\right)\right), \tag{10}$$

where G denotes the clean graph and G' denotes the predicted graph, f_θ is an encoder trained on the clean graph G, and $I\left(G, f_\theta\left(G\right)\right)$ represents the mutual information between the G and its representation $f_\theta\left(G\right)$. The GPD quantifies the difference in mutual information between the clean graph and the predicted graph, reflecting how much the predicted graph's structure deviates from the clean graph in terms of its learned representations. A lower value of $GPD(\theta)$ indicates

that the predicted graph under examination contains a higher level of pollution due to perturbed edges.

Depending on the value of GPD, problem (10) can guide two simple sub-problems:

$$\begin{cases} G' \textit{ contains adversarial edges, if } GPD > \alpha \\ G' \textit{ can be trusted, otherwise} \end{cases}.$$

A highly expressive encoder, which undergoes continuous fine-tuning based on clean graphs, fails to capture the interdependencies between polluted graphs and their representations, resulting in $GPD > \alpha$.

Although some studies [Xu *et al.*, 2022; Jin *et al.*, 2023; Zhu *et al.*, 2020] propose the concept of Representation Vulnerability (RV) to quantify the impact of adversarial attacks on representations and develop corresponding defense strategies, their primary focus is on learning robust representations from clean graphs. In contrast, the goal of GPD is to quantify the level of pollution in the predicted graph and refine the polluted graph into a cleaner one that is more suitable for learning effective representations.

Algorithm 1 Overall Procedure of Algorithm

Input: Attacked graph $G^* = (A^*, X)$, rough graph G, learning rate δ, hyperparameters α
Output: Refined model θ, effective representation $f_\theta(G)$

```
Calculate model θ according Eq. (1)
While not early-stop do
    Randomly select E', and combine it with G to form G' = (A', X), where E' ∈ G* & E' ∉ G
    If I(G, f_θ(G)) − I(G', f_θ(G')) < α
        // updating clean graph
        Let G = G'
        // fine-tuning the model
        θ ← θ − δ∇_θ I(G; f_θ(G))
    Else
        // fine-tuning the model
        θ ← θ − δ∇_θ I(G; f_θ(G))
End while
Calculate effective representation f_θ(G)
Return θ, f_θ(G)
```

Graph Representation Learning Framework

The overall graph representation learning framework is outlined in Algorithm 1. We begin by randomly initializing the parameters θ for the unsupervised graph contrastive learning method, which employs the DGI (Deep Graph Infomax) approach defined in Eq. (1). Next, we randomly select edges that belong to the polluted graph but not to the clean graph and sequentially add them to the clean graph to generate the predicted graph G'. We then calculate the difference in mutual information between the clean graph G and its representation $f_\theta(G)$, and between the predicted graph G' and its representation $f_\theta(G')$, denoted as GPD(θ) (step 4). If GPD(θ) is less than the threshold α, the predicted graph G' is deemed trustworthy and used to fine-tune the model via Eq. (1). Conversely, if GPD(θ) exceeds α, it indicates that G' contains a significant number of perturbed edges. In this case, we use the cleaner graph G to fine-tune the model, repeating the process until convergence is achieved. The algorithm ultimately returns the refined model and robust representation $f_\theta(G)$, which can then be used for downstream tasks such as node classification, selecting only the graph G' with the smallest GPD as the refined graph for each round.

4 Experiments

In this section, we demonstrate that our model can train robust, high-quality representations on graphs under adversarial attacks. We focus on addressing the following research questions. Q1: Can our model learn robust and efficient representations under PGD attacks with varying perturbation rates? Q2: Can our model effectively defend against different types of graph adversarial attacks? Q3: How sensitive is the model to parameter variations? Q4: What is the impact of each component on the overall model performance?

4.1 Experimental Settings

Datasets

We conduct experiments on four widely used benchmark datasets: Cora, Citeseer, Pubmed [Kipf and Welling, 2017; Sen *et al.*, 2008], and Polblogs [Adamic and Glance, 2005]. The first three are citation networks, where nodes represent documents and edges denote citation links between them. Polblogs is a political blog network, in which nodes are blogs and edges indicate hyperlinks. Since Polblogs lacks node features, we use an identity matrix as its attribute matrix following [Xu *et al.*, 2022]. Detailed dataset statistics are provided in Table 1.

	$\|V\|$	$\|E\|$	$\|Feature\|$	$\|Class\|$
Cora	2,708	5,429	1,433	7
Citeseer	3,327	4,732	3,703	6
Pubmed	19,717	44,338	500	3
Polblogs	1,490	16,714	-	2

Table 1: Statistics of the experimental data.

Baselines

We compare our methods with nine state-of-the-art defense Graph Neural Networks (GNNs). The baselines are categorized into two main groups: embedding-based methods, including DGI [Veličković *et al.*, 2019], Jaccard [Wu *et al.*, 2019], SVD [Entezari *et al.*, 2020], STABLE [Li *et al.*, 2022], and STRG [Li *et al.*, 2023]; and anomaly detection-based methods, such as Dominant [Ding *et al.*, 2019], CoLA [Liu *et al.*, 2021], GRADATE [Duan *et al.*, 2023], and GFCN [Mesgaran and Hamza, 2024]. Additionally, we implement three non-targeted structural adversarial attack methods: PGD [Madry *et al.*, 2018], MetaAttack [Zügner and Günnemann, 2019], and Random.

Implementation Details

For each dataset, we randomly divided 10% of the nodes for training and 80% for testing, as in [Feng *et al.*, 2024a]. In our work, we set the hyperparameters $\alpha = 5e-4$. Regarding the attack setup, the polluted graph is generated using the

Dataset	Ptb rate	Model									
		DGI	JACCARD	Dominant	CoLA	SVD	STABLE	STRG	GRADATE	GFCN	Ours
Cora	5%	72.3	73.8	74.6	73.8	72.6	73.5	75.6	74.2	72.4	**77.3**
	10%	66.5	67.2	68.7	68.3	66.9	68.8	69.0	69.2	65.8	**71.8**
	20%	60.3	60.5	61.1	62.5	59.8	61.9	63.5	62.8	60.2	**67.0**
	30%	54.1	54.5	54.3	58.9	55.2	56.1	61.8	58.6	52.8	**64.3**
	40%	49.9	50.2	49.2	54.8	50.4	54.9	57.0	53.4	50.6	**59.2**
Citeseer	5%	63.1	64.7	66.3	66.2	64.2	65.4	66.7	67.2	63.5	**69.0**
	10%	60.4	62.5	63.0	64.5	61.4	63.3	64.5	64.1	59.8	**67.4**
	20%	54.2	53.6	53.8	58.8	54.5	58.0	58.0	57.4	57.7	**62.3**
	30%	47.3	49.8	45.4	51.8	47.7	49.0	53.5	50.5	48.6	**57.7**
	40%	41.8	43.5	41.4	46.6	42.5	42.9	49.9	45.1	43.5	**53.1**
Pubmed	5%	60.5	60.1	59.7	62.2	59.6	59.6	64.5	62.2	62.2	**66.5**
	10%	52.7	53.5	52.9	55.6	51.4	51.9	57.9	54.2	53.8	**60.4**
	20%	42.8	41.5	42.3	43.3	41.6	41.1	50.1	43.8	43.6	**54.7**
	30%	36.3	37.8	38.3	39.9	36.5	38.8	44.8	38.3	37.6	**48.2**
	40%	34.9	36.4	36.4	37.9	11.1	34.5	40.5	37.7	36.2	**43.6**
Polblogs	5%	85.5	85.4	86.2	86.4	85.2	86.0	86.5	86.4	85.6	**87.4**
	10%	82.0	82.8	82.1	83.6	81.5	83.6	84.8	83.3	83.1	**85.7**
	20%	80.3	81.5	81.8	81.6	80.1	82.4	83.4	81.6	81.8	**84.1**
	30%	78.3	79.6	78.6	78.8	77.5	80.7	82.1	79.9	79.8	**83.1**
	40%	76.5	77.2	77.1	77.3	75.3	79.2	80.8	77.0	78.2	**82.2**

Table 2: Node classification performance (Acc%) on Cora, Citeseer, Pubmed and Polblogs under PGD attack.

aforementioned three structural perturbation methods. We set the perturbation ratio δ, which is tuned from $\{0.05, 0.1, 0.2, 0.3, 0.4\}$, based on the default parameters. For the parameter design of the baselines, besides using the default parameters, we set a threshold for JACCARD, and STABLE from $\{0.2, 0.3, 0.4, 0.5, 0.6\}$ and select the best-performing one. For Dominant, CoLA, GRADATE, and GFCN, we delete edges based on node or structural anomaly rankings, ensuring that the number of removed edges corresponds to the number of edges targeted in the attack.

4.2 Performances against PGD Attack (Q1)

Table 2 presents the node classification accuracy of our proposed model compared to nine baselines under PGD adversarial attacks. Our method consistently outperforms all baselines across different perturbation ratios, achieving ACC gains of 77.3%, 69.0%, 66.5%, and 87.4% on Cora, Citeseer, PubMed, and Polblogs at a 5% perturbation rate. Compared to similarity-based approaches, our model's advantage lies in its ability to identify more accurate structures from the outset. In contrast to anomaly detection-based methods, we leverage trusted structures to effectively identify the correct structures within untrusted regions. Additionally, as perturbation increases, our model's performance decreases more slowly across all datasets, similar to the STABLE baseline. This resilience is attributed to our approach of initially selecting a rough graph and fine-tuning the model based on refined graph structures, reducing sensitivity to adversarial perturbations.

4.3 Performance against Different Attacks (Q2)

In this section, we evaluate the performance of our model under MetaAttack and random perturbations on the Cora and Citeseer datasets, as shown in Figure 4. Our experiments reveal that adversarial attacks primarily aim to connect distant nodes, resulting in both structural anomalies and mismatches between the polluted graph and its learned representations. By leveraging mutual information, our model effectively mitigates these issues, consistently outperforming baseline methods across different perturbation techniques. Moreover, as the perturbation rate increases, our model's performance declines more gradually than other methods. This resilience can be attributed to the reduced number of perturbed edges, which allows the accurate structures to provide more reliable information and making it easier to identify and exclude false edges.

4.4 Sensitivity Analysis (Q3)

To verify the effectiveness of our framework, we investigate two parameters under a 5% PGD attack on different datasets: the GPD threshold α, the proportion of edges removed in advance, and the number of edge verifications q. First, we vary α to observe changes in classification accuracy. As shown in Figure 5(a), extremely small α values lead to premature termination, while excessively large values can cause severely polluted graphs to be misclassified as clean. Hence, we set α to 5×10^{-4} for a robust balance. Second, increasing the proportion of edges removed in advance generally improves model robustness, indicating that screening out more potentially perturbed edges helps stabilize subsequent refinements. Finally, we examine the effect of q. As shown in Figure 5(b), a higher q yields better refinement, as too few queries risk incomplete structural information. We find that $q = 10$ sufficiently captures most accurate structures for downstream tasks. Overall, our model remains relatively stable within reasonable parameter ranges, confirming that thresholding polluted edges and adjusting verification times effectively enhance both robustness and representation quality.

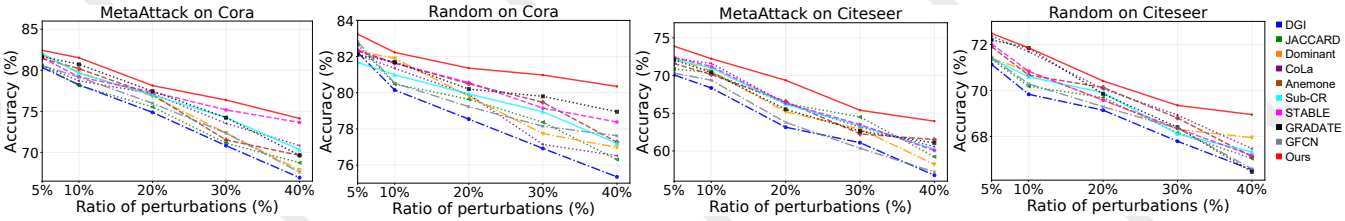
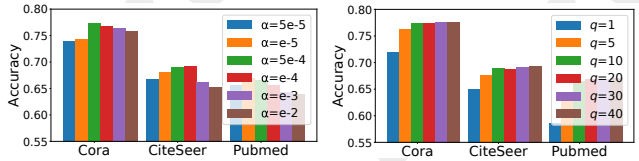


Figure 4: Node classification performance (Acc %) on Cora and Citeseer under different attacks.



(a) Parameter α. (b) Parameter q.

Figure 5: Experimental results for parameters α and q.

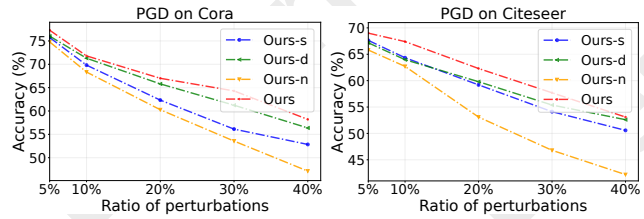


Figure 6: Evaluation of different anomaly detection methods.

4.5 Ablation Study (Q4)

In this section, we provide a detailed analysis of our method's performance under various conditions on the Cora dataset. Specifically, we compare three variants: Ours-similarity **(Ours-s)**, which first learns the representations of the attacked graph using DGI and retains edges with higher similarity; Ours-dominant **(Ours-d)**, which uses autoencoders to compute the anomaly score and selects edges with lower scores; and Ours-none **(Ours-n)**, which randomly selects edges from the polluted graph. Figure 6 shows the results of these variants. We make the following observations: using these methods to obtain the rough graph is less effective than employing the CoLA method. This is likely because CoLA's anomaly detection approach identifies anomaly patterns rather than relying on structural information, making it less influenced by the presence of anomalous edges.

5 Related Works

Numerous studies have demonstrated that graph neural networks (GNNs) are susceptible to attacks, particularly through adversarial perturbations. Nettack [Zügner *et al.*, 2019] is a pioneering study in the domain of graph adversarial attacks, pioneering targeted attacks through subtle changes to the structure and node attributes of graphs. Building on this, Mettack [Zügner and Günnemann, 2019] employs a meta-learning approach to tackle the bi-level optimization problem in adversarial attacks. This method of perturbation persistently diminishes the representational capacity of graph convolutional networks, enabling maximally effective untargeted attacks on graph neural network learning. PGD [Madry *et al.*, 2018] method uses a projected gradient descent topology attack for dual optimization, effectively applying convex relaxation optimization to enable gradient-based adversarial attacks on discrete graph data. These graph attack algorithms can be directly applied during the graph training phase, leading to extensive research on defending against adversarial perturbations. For defense against different attacks, [Wu *et al.*, 2019] propose using the Jaccard similarity of node representations to remove adversarial edges. [Entezari *et al.*, 2020] propose defending against adversarial attacks by using Singular Value Decomposition (SVD) for low-rank approximation, which retains the high-rank components of the graph and effectively reduces the impact of attacks. [Li *et al.*, 2022] propose removing edges between dissimilar nodes and additionally selecting top-k edges to reduce the impact of residual perturbations. STRG [Li *et al.*, 2023] leverages the local structures of test nodes and pseudo-labels to train a GCN, reducing the distribution shift between the training and test sets, thereby enhancing robustness. Unlike these studies, we recognize the potential impact of perturbed edges on attack detection models. Consequently, we design a method to refine contaminated graphs using clean graphs, enabling more detailed refinement of the contaminated graphs to better defend against adversarial attacks.

6 Conclusion

In this paper, we propose a novel robust representation learning method for unsupervised learning through graph refining. Our framework iteratively refines graphs using cleaner structures, guided by the Graph Pollution Degree (GPD) metric to quantify graph pollution and direct the refinement process. By employing contrastive learning and anomaly detection to identify and exclude perturbed edges, we progressively improve the quality of the learned representations. Extensive experiments demonstrate our model's effectiveness in learning representation, leading to significant improvements in node classification. One limitation of our method is its reliance on the homophily assumption, which may reduce effectiveness on graphs with low homophily. For future work, we aim to adapt our defense technique to effectively handle additional attack types, specifically injection and backdoor attacks, and assess our framework on more complex and varied graph data, such as heterogeneous and multimodal graphs.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (Nos. 62272340, 92370111, 62422210, 62276187, 62302333, U22B2036, 62261136549), the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) (No. GML-KF-24-16), the Hebei Natural Science Foundation (F2024202047), the Technological Innovation Team of Shaanxi Province (No. 2025RS-CXTD-009), the International Cooperation Project of Shaanxi Province (No. 2025GH-YBXM-017) and the Tencent Foundation and XPLORER PRIZE.

References

[Abusnaina *et al.*, 2021] Ahmed Abusnaina, Yuhang Wu, Sunpreet Arora, Yizhen Wang, Fei Wang, Hao Yang, and David Mohaisen. Adversarial example detection using latent neighborhood graph. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7687–7696, 2021.

[Adamic and Glance, 2005] Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery*, pages 36–43, 2005.

[Bojchevski and Günnemann, 2019] Aleksandar Bojchevski and Stephan Günnemann. Adversarial attacks on node embeddings via graph poisoning. In *International Conference on Machine Learning*, pages 695–704, 2019.

[Chang *et al.*, 2023] Chao Chang, Junming Zhou, Yu Weng, Xiangwei Zeng, Zhengyang Wu, Chang-Dong Wang, and Yong Tang. Kgtn: Knowledge graph transformer network for explainable multi-category item recommendation. *Knowledge-Based Systems*, 278:110854, 2023.

[Ding *et al.*, 2019] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 594–602, 2019.

[Duan *et al.*, 2023] Jingcan Duan, Siwei Wang, Pei Zhang, En Zhu, Jingtao Hu, Hu Jin, Yue Liu, and Zhibin Dong. Graph anomaly detection via multi-scale contrastive learning networks with augmented view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7459–7467, 2023.

[Entezari *et al.*, 2020] Negin Entezari, Saba A Al-Sayouri, Amirali Darvishzadeh, and Evangelos E Papalexakis. All you need is low (rank) defending against adversarial attacks on graphs. In *Proceedings of the 13th international conference on web search and data mining*, pages 169–177, 2020.

[Feng *et al.*, 2024a] Bingdao Feng, Di Jin, Xiaobao Wang, Fangyu Cheng, and Siqi Guo. Backdoor attacks on unsupervised graph representation learning. *Neural Networks*, 180:106668, 2024.

[Feng *et al.*, 2024b] Shengyu Feng, Baoyu Jing, Yada Zhu, and Hanghang Tong. Ariel: Adversarial graph contrastive learning. *ACM Transactions on Knowledge Discovery from Data*, 18(4):1–22, 2024.

[Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[He *et al.*, 2025] Meixia He, Peican Zhu, Keke Tang, and Yangming Guo. Hypergraph attacks via injecting homogeneous nodes into elite hyperedges. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 282–290, 2025.

[Huang *et al.*, 2022] Feihu Huang, Peiyu Yi, Jince Wang, Mengshi Li, Jian Peng, and Xi Xiong. A dynamical spatial-temporal graph neural network for traffic demand prediction. *Information Sciences*, 594:286–304, 2022.

[Jin *et al.*, 2023] Di Jin, Bingdao Feng, Siqi Guo, Xiaobao Wang, Jianguo Wei, and Zhen Wang. Local-global defense against unsupervised adversarial attacks on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8105–8113, 2023.

[Kipf and Welling, 2017] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Machine Learning*, 2017.

[Li *et al.*, 2022] Kuan Li, Yang Liu, Xiang Ao, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Reliable representations make a stronger defender: Unsupervised structure refinement for robust gnn. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 925–935, 2022.

[Li *et al.*, 2023] Kuan Li, Yang Liu, Xiang Ao, and Qing He. Revisiting graph adversarial attack and defense from a data distribution perspective. In *International Conference on Learning Representations*, 2023.

[Liu *et al.*, 2021] Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2378–2392, 2021.

[Ma *et al.*, 2021] Yao Ma, Xiaorui Liu, Tong Zhao, Yozen Liu, Jiliang Tang, and Neil Shah. A unified view on graph neural networks as graph signal denoising. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 1202–1211, 2021.

[Madry *et al.*, 2018] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[Mesgaran and Hamza, 2024] Mahsa Mesgaran and A Ben Hamza. Graph fairing convolutional networks for anomaly detection. *Pattern Recognition*, 145:109960, 2024.

[Peng *et al.*, 2018] Zhen Peng, Minnan Luo, Jundong Li, Huan Liu, Qinghua Zheng, et al. Anomalous: A joint modeling approach for anomaly detection on attributed networks. In *International Joint Conference on Artificial Intelligence*, volume 18, pages 3513–3519, 2018.

[Qiu *et al.*, 2020] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1150–1160, 2020.

[Sen *et al.*, 2008] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–93, 2008.

[Sun *et al.*, 2019] Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *International Conference on Learning Representations*, 2019.

[Sun *et al.*, 2022] Lichao Sun, Yingtong Dou, Carl Yang, Kai Zhang, Ji Wang, S Yu Philip, Lifang He, and Bo Li. Adversarial attack and defense on graph data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7693–7711, 2022.

[Tang *et al.*, 2020] Xianfeng Tang, Yandong Li, Yiwei Sun, Huaxiu Yao, Prasenjit Mitra, and Suhang Wang. Transferring robustness for graph neural network against poisoning attacks. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 600–608, 2020.

[Veličković *et al.*, 2019] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. In *International Conference on Learning Representations*, 2019.

[Vlaic *et al.*, 2018] Sebastian Vlaic, Theresia Conrad, Christian Tokarski-Schnelle, Mika Gustafsson, Uta Dahmen, Reinhard Guthke, and Stefan Schuster. Modulediscoverer: Identification of regulatory modules in protein-protein interaction networks. *Scientific Reports*, 8(1):1–11, 2018.

[Wang *et al.*, 2023] Hang Wang, David J Miller, and George Kesidis. Anomaly detection of adversarial examples using class-conditional generative adversarial networks. *Computers & Security*, 124:102956, 2023.

[Wang *et al.*, 2025] Xiaobao Wang, Yujing Wang, Dongxiao He, Zhe Yu, Yawen Li, Longbiao Wang, Jianwu Dang, and Di Jin. Elevating knowledge-enhanced entity and relationship understanding for sarcasm detection. *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[Wu *et al.*, 2019] Huijun Wu, Chen Wang, Yuriy Tyshetskiy, Andrew Docherty, Kai Lu, and Liming Zhu. Adversarial examples for graph data: Deep insights into attack and defense. In *International Joint Conference on Artificial Intelligence*, pages 4816–4823, 2019.

[Xu *et al.*, 2022] Jiarong Xu, Yang Yang, Junru Chen, Xin Jiang, Chunping Wang, Jiangang Lu, and Yizhou Sun. Unsupervised adversarially robust representation learning on graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4290–4298, 2022.

[Yan *et al.*, 2025] Fengyu Yan, Xiaobao Wang, Dongxiao He, Longbiao Wang, Jianwu Dang, and Di Jin. Hetergp: Bridging heterogeneity in graph neural networks with multi-view prompting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 21895–21903, 2025.

[Zhang and Zitnik, 2020] Xiang Zhang and Marinka Zitnik. Gnnguard: Defending graph neural networks against adversarial attacks. *Advances in neural information processing systems*, 33:9263–9275, 2020.

[Zhang *et al.*, 2019] Yingxue Zhang, S Khan, and Mark Coates. Comparing and detecting adversarial attacks for graph deep learning. In *International Conference on Learning Representations*, 2019.

[Zhang *et al.*, 2020] Ziwei Zhang, Peng Cui, and Wenwu Zhu. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):249–270, 2020.

[Zhu *et al.*, 2020] Sicheng Zhu, Xiao Zhang, and David Evans. Learning adversarially robust representations via worst-case mutual information maximization. In *International Conference on Machine Learning*, pages 11609–11618, 2020.

[Zhu *et al.*, 2021] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Graph contrastive learning with adaptive augmentation. In *Proceedings of the web conference 2021*, pages 2069–2080, 2021.

[Zhu *et al.*, 2024] Peican Zhu, Zechen Pan, Yang Liu, Jiwei Tian, Keke Tang, and Zhen Wang. A general black-box adversarial attack on graph-based fake news detectors. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 568–576, 2024.

[Zhuang and Al Hasan, 2022] Jun Zhuang and Mohammad Al Hasan. Defending graph convolutional networks against dynamic graph perturbations via bayesian self-supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4405–4413, 2022.

[Zhuang *et al.*, 2024] Jin Zhuang, Xiao-Yuan Jing, and Xiaodong Jia. Mining negative samples on contrastive learning via curricular weighting strategy. *Information Sciences*, 668:120534, 2024.

[Zügner and Günnemann, 2019] Daniel Zügner and Stephan Günnemann. Adversarial attacks on graph neural networks via meta learning. In *International Conference on Learning Representations*, 2019.

[Zügner *et al.*, 2019] Daniel Zügner, Amir Akbarnejad, and Stephan Günnemann. Adversarial attacks on neural networks for graph data. In *International Joint Conference on Artificial Intelligence*, 2019.