Refine then Classify: Robust Graph Neural Networks with Reliable Neighborhood Contrastive Refinement

Shuman Zhuang[1], **Zhihao Wu**[2*], **Zhaoliang Chen**[3], **Hong-Ning Dai**[3], **Ximeng Liu**[1]

[1]College of Computer and Data Science, Fuzhou University, Fuzhou, China
[2]College of Computer Science and Technology, Zhejiang University, Hangzhou, China
[3]Department of Computer Science, Hong Kong Baptist University, Hong Kong, China
shumanzhuang@163.com, zhihaowu1999@gmail.com, chenzl23@outlook.com, henrydai@hkbu.edu.hk, snbnix@gmail.com

Abstract

Graph Neural Networks (GNNs) have exhibited remarkable capabilities for dealing with graph-structured data. However, recent studies have revealed their fragility to adversarial attacks, where imperceptible perturbations to the graph structure can easily mislead predictions. To enhance adversarial robustness, some methods attempt to learn robust representation through improving GNN architectures. Subsequently, another approach suggests that these GNNs might taint feature information and have poor classifier performance, leading to the introduction of Graph Contrastive Learning (GCL) methods to build a refining-classifying pipeline. However, existing methods focus on global-local contrastive strategies, which fails to address the robustness issues inherent in the contexts of adversarial robustness. To address these challenges, we propose a novel paradigm named GRANCE to enhance the robustness of learned representations by shifting the focus to local neighborhoods. Specifically, a dual neighborhood contrastive learning strategy is designed to extract local topological and semantic information. Paired with a neighbor estimator, the strategy can learn robust representations that are resilient to adversarial edges. Additionally, we also provide an improved GNN as classifier. Theoretical analyses provide a stricter lower bound of mutual information, ensuring the convergence of GRANCE. Extensive experiments validate the effectiveness of GRANCE compared to state-of-the-art baselines against various adversarial attacks.

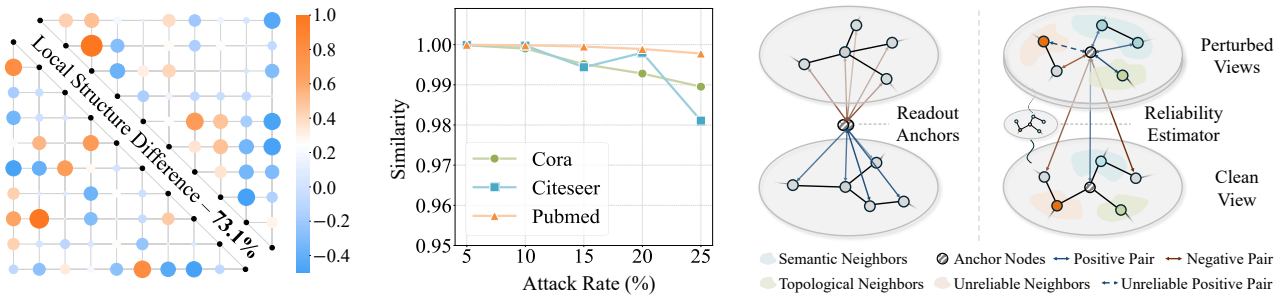
Introduction

Graph Neural Networks (GNNs), with their effective message passing mechanisms, have excelled in numerous graph-related tasks (Chen et al. 2023b; Wan et al. 2024; Wu, Zhang, and Fan 2024) such as anomaly detection (Pan et al. 2023; Cai et al. 2024; Zhang et al. 2024; Liu et al. 2024), genomics (Li et al. 2022a; Hickey et al. 2023), network analysis (Yu et al. 2024; Wu et al. 2024; Chen et al. 2023a), and disease propagation modeling (Lao et al. 2022; Gao et al. 2023), thanks to their ability to learn and uncover complex patterns from neighbor exchanges (Li et al. 2024; Wu et al. 2023; Chen et al. 2024). However, despite their successes, GNNs demonstrate vulnerabilities to adversarial attacks involving noisy or deliberately crafted edges (Zhu et al. 2022; Zügner, Akbarnejad, and Günnemann 2018; Wang et al. 2024c), exposing significant robustness flaws that threaten security in critical environments. Addressing these vulnerabilities is essential for broadening GNNs' applications.

Due to the unique characteristics of graphs and mechanisms of GNNs (Gilmer et al. 2017; Zhuang et al. 2024; Lu et al. 2024; Zheng et al. 2023), structural attacks (Xu et al. 2019; Zugner and Gunnemann 2019) have emerged as a more effective and prevalent method of targeting GNNs compared to traditional feature attacks. To defend against structural attacks, some researchers re-engineered the GNN models to refine the representation through an in-processing way (Zhu et al. 2019; Wang et al. 2024a). Nonetheless, the inherent message-passing mechanism within these models permitted interactions between representations and the contaminated graph structure, which inevitably taints feature information so that capping performance potential. In response to these challenges, there has been a shift towards decoupling this process, i.e., first refining and then classifying using a GNN (Wu et al. 2019). Typically, the refining is a static procedure, driven solely by feature information. This approach, while preventing contamination, tends to underutilize the untainted segments of the structure. To overcome these drawbacks, Graph Contrastive Learning (GCL), which learns invariant information from perturbed graphs (Wang et al. 2024b; Zheng et al. 2022b,a), is being considered as a representation refining technique. Although GCL also suffered severely from structural attacks, STABLE (Li et al. 2022b) surprisingly revealed that by integrating GCL with supervised GNNs, exceptional performance can be achieved. The core idea involves a rough preprocessing to construct contrastive views and refining representations via GCL. Ultimately, an improved GNN is performed to derive robust classification results.

Following this idea, some research (Tao et al. 2024) has made further progress. However, the GCL-based refining-classifying pipeline is still in its infancy, where STABLE and other existing methods have not explored the intrinsic challenges of adversarial robustness. They directly adopt the well-known global-local contrastive strategy (Velickovic et al. 2019), which is contrary to the principle of adversarial defense as it disrupts the intrinsic geometric structure in the feature space. It is worth mentioning that decoupling aims to protect feature information, but these methods still poten-

*Corresponding Authors.



(a) (b) (c)

Figure 1: (a) Similarity matrices and neighbor overlap rate for original and learned features on Cora using the global-local contrastive strategy; (b) Similarity curves of global representations at different attack rates versus unattacked representations; (c) Descriptions of the global-local and proposed dual neighborhood contrastive strategies.

tially compromise it. They generate global embedding for the perturbed graph through readout, which unreasonably assumes the read global information can effectively reflect subtle attacks, and also neglects the locality of operators in GNNs as well as the effects of structural attacks on node neighborhoods. We detail the discussion later, where we argue that the global-local strategy is not suitable enough.

To tackle this, we design a new refining-classifying framework called robust GNNs with ReliAble Neighborhood Contrastive rEfinement (GRANCE), which deeply into the insight of adversarial robustness on graphs through a fine-grained perspective. Instead of focusing on global information, we recognize that adversarial attacks are exactly designed to corrupt vulnerable local neighborhoods in globally imperceptible ways. Unlike traditional GCL which uses artificially perturbed graphs as augmentations, GRANCE creates a relatively clean view alongside several perturbed views through varied pre-processing, leveraging the natural "perturbations" by attacks as augmentations. Our neighborhood contrastive strategy then maximizes the agreement between the same node and its neighborhoods across different views, and meanwhile forces them to be distinguishable from other nodes, thereby emphasizing local robustness against attacks. Neighborhoods are defined both topologically and semantically, with topology based on graph structure and semantics derived from local geometry in feature space. The reliability of all these neighbors is estimated in the clean view to fully exploit feature semantics. Based on this framework, multiple views within the refinement module collaborate during training. This collaboration leads to a robust representation that is resilient to adversarial edges within neighborhoods.

In summary, we have the following main contributions:

- Articulate the refining-classifying graph defending paradigm and provide detailed analyses on drawbacks of the existing methods, revealing their inherent challenges of achieving adversarial robustness.
- Propose an effective and robust GCL framework, shifting focus from the global perspective to the crucial local neighborhoods under the context of structural attacks.
- Extensive experiments demonstrate the outstanding adversarial robustness and performance of our method, surpassing state-of-the-art defense models such as STABLE.

Proposed Method

Notations Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with adjacency matrix $\mathbf{A} \in \{0,1\}^{n\times n}$, where $\mathcal{V}$ is the set of nodes with $|\mathcal{V}| = n$ and $\mathcal{E}$ is the set of edges. $\mathbf{X} \in \mathbb{R}^{n\times m}$ is the feature matrix containing the feature information, where each node is associated with an m-dimensional feature vector. The degree matrix is defined as $\mathbf{D}$, where the i-th diagonal element is $d_i = \sum_j \mathbf{A}_{ij}$ and $\mathbf{A}_{ij} = 1$ denotes the existence of edge $(v_i, v_j) \in \mathcal{E}$ linking nodes $v_i, v_j \in \mathcal{V}$.

Emprical Findings We analyze the issues of existing GCL-based defenders and address the following questions:

Does global-local strategy hurt features? Taking STABLE on the Cora dataset as an example, we tested the nearest neighbors of nodes in the original features and the learned representation under 25% attack rate, respectively. Figure 1 (a) illustrates partial similarity matrices of the original feature and representation in the lower left and upper right triangles, respectively. It can be observed that the node similarities in the original and latent spaces are significantly different. We also discovered that 73.1% of the nearest neighbors had changed after training with the global-local strategy, clearly indicating that the representations do hurt node features by disrupting the semantic information. A possible reason is the global-local strategy neglects attention to local geometry and forces the nodes to learn global information.

Can global information capture subtle attacks? We employed the Cora dataset, subjected to various degrees of structural attacks, to generate the global representation through GNNs and an average readout function (following STABLE). Taking the global representation at 0% attack rate as a baseline, we tested and plotted the similarity of global representations at other attack rates to it in Figure 1 (b). It can be seen that the global representation hardly changes as the attack rate increases, indicating that it struggles to capture the effects of structural attacks. Therefore, existing methods do not effectively learn attack-resilient embedding from contrastive strategies with global representations.

What do these two findings imply? Structural adversarial attacks are destructive, only leaving two subtle openings

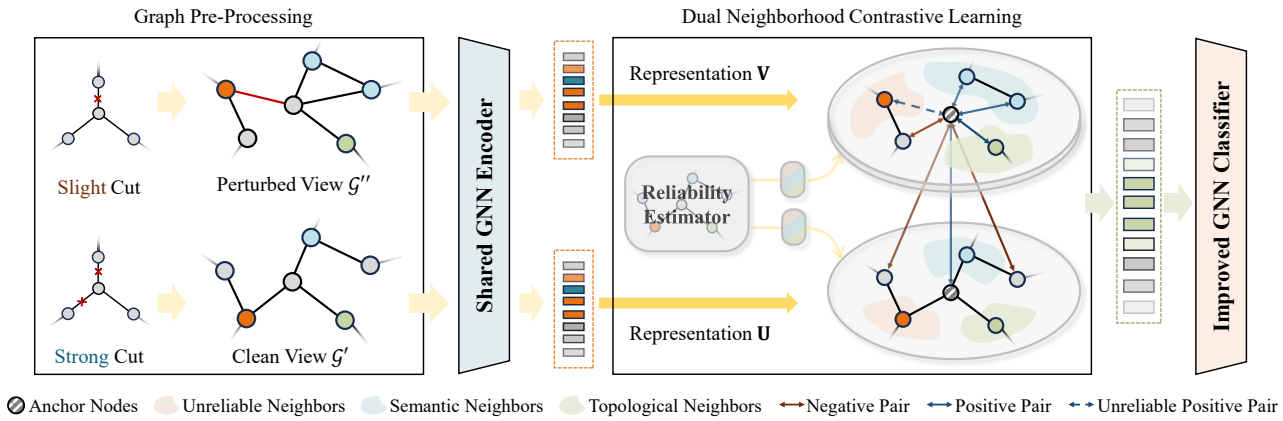


Figure 2: Overview of the proposed GRANCE. It generates relatively clean and perturbed views through semantic-based preprocessing, then conducts dual neighborhood graph contrastive learning, considering both semantic and topological neighborhoods. The reliability of neighbors is calculated by an estimator. Finally, results are obtained using an improved GNN.

to capitalize on: the uncorrupted or easily defensible feature information and the partially correct structural information. Maximizing the use of these two aspects is critical for effective defense, which has led to the development of refining-classifying pipelines. We illustrate the global-local contrastive strategy on the left side of Figure 1 (c), which disrupts the feature space's geometric structure while compromising reliable information and failing to capture local nuances through global representations. This deficiency makes model training against such destruction challenging. In contrast, our method focuses on "local" and develops a fine-grained way. We introduces a dual neighborhood contrastive strategy, considering the critical role of local neighborhoods in graphs and GNNs. By integrating an estimator, our method effectively leverages accurate neighborhood information, fully protecting and utilizing the feature space.

Overall Framework As shown in Figure 2, GRANCE first generates relatively clean and perturbed views via semantic-based preprocessing, and then implements dual neighborhood GCL that accounts for both semantic and topological neighborhood contexts. The reliability of these neighbors is determined by an estimator. Ultimately, it yields results through an enhanced GNN classifier.

Semantic-based Graph Pre-processing According to the preliminary experiments and analyses, existing graph defense methods do not sufficiently utilize node features, despite the critical importance of feature information for adversarial robustness. Beyond the topological structure provided by graphs, node features carry rich real-world semantics and form an inherent semantic geometric structure within the feature space, representing the intrinsic associations among samples. However, classical methods may not effectively preserve semantic structures and adversarial attacks may delete edges, so it is necessary to complete the graph. Formally, we begin by calculating the similarity between samples

$$s_{i,j} = \mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j), \tag{1}$$

where sim is a certain similarity measure defined over $\mathbf{X}$, and $\mathbf{x}_i$ denotes the feature of the i-th sample. Then we define a completion operation. For an given graph $\mathcal{G}$ with the edge set $\mathcal{E}$, the edge completion is performed by

$$\mathcal{G}_{\mathrm{Com}} = \mathrm{T}_{\theta_1}(\mathcal{G}, s) = \big(\mathcal{V}, \mathcal{E} \cup \mathcal{E}_{\mathrm{Com}}\big), \tag{2}$$

where $\mathcal{E}_{\mathrm{Com}} = \{(v_i, v_j) \notin \mathcal{E} \mid s_{i,j} > \theta_1\}$ is the edge set to be completed, θ_1 is the threshold for completing and $\cup$ is the set union operation. As optional, it can also be defined as $\mathcal{E}_{\mathrm{Com}} = \{(v_i, v_j) \notin \mathcal{E} \mid s_{i,j} \in \mathrm{Top}_k(s_{i,:})\}$, where the function Top_k returns the set of k largest values. Completion integrates the inherent geometric structure of the feature space into the graph via constructing the semantic neighborhood, achieving feature information preservation during subsequent learning processes. For generalization, we set a trade-off parameter α to measure the importance of semantic versus topological edges.

On the other hand, graph structure attacks tend to add edges against this semantic structure. Therefore, by leveraging semantic information, we can also effectively eliminate a portion of the easily detectable adversarial edges. In a similar manner, the similarity between pairs of samples is first calculated using a similarity function. We then define the following cut operations:

$$\mathcal{G}_{\mathrm{Cut}} = \mathrm{T}_{\theta_2}(\mathcal{G}, s) = \big(\mathcal{V}, \mathcal{E} \backslash \mathcal{E}_{\mathrm{Cut}}\big), \tag{3}$$

where $\mathcal{E}_{\mathrm{Cut}} = \{(v_i, v_j) \in \mathcal{E} \mid s_{i,j} < \theta_2\}$ is the edge set to be cut, θ_2 is the threshold for cutting and $\backslash$ is the set difference operation. Consequently, we can sever all edges connecting nodes with a similarity below a predetermined threshold, resulting in a graph that is cleaner relative to the original.

After defining the completion and cut operations, we explore how to generate views for GCL. Initially, we construct a relatively clean graph, called the clean view, through completion and cutting. This view employs strong cutting—setting a high threshold—to ensure it contains a higher proportion of normal edges, though some structural information may be lost. Subsequently, we obtain a set of

edges with slight cutting by performing random sampling within the set $\mathcal{E}_{\text{Cut}}$, as

$$\mathcal{E}'_{\text{Cut}} = \{(v_i, v_j) \in \mathcal{E}_{\text{Cut}} \mid t_{i,j} = 1, t_{i,j} \sim \mathcal{B}(p)\}, \quad (4)$$

where $t_{i,j}$ is a random variable that follows the Bernoulli distribution $\mathcal{B}(p)$ with parameter p. Through slight cutting, the resulting graph has more noise compared to the clean view but also retains more information, termed the perturbed view. By conducting V random samplings, we can generate V distinct perturbed views $\{\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_V\}$. Without loss of generality, we consider the scenario with one clean view and one perturbed view, denoted as $\mathcal{G}'$ and $\mathcal{G}''$.

The proposed pre-processing is illustrated in Figure 2, where completion constructs semantic neighborhood by adding edges between semantically similar nodes while strong and slight cutting generate contrastive views by removing suspicious edges in different levels. Based on the semantic information, the completion enhances the protection of the intrinsic geometric structure of nodes. Afterward, strong cut leads to a clean view, while slight cuts produce perturbed but informative views, the difference between them is mainly composed of adversarial edges, thus this approach utilizes the natural adversarial edges as self-supervised information. Again, we note that perturbed views are generated by completion and slight cut, and "clean" view refers to the relatively clean graph obtained by completion and strong cut, which does not imply that unattacked graphs are available in this paper.

Dual Neighborhood Contrastive Learning Formally, we specify the relatively clean and perturbed graphs as $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ with adjacency matrix $\mathbf{A}'$ and $\mathcal{G}'' = (\mathcal{V}, \mathcal{E}'')$ with adjacency matrix $\mathbf{A}''$. To encode the two graphs $\mathcal{G}'$ and $\mathcal{G}''$, we typically define the following GNN encoder to learn corresponding node representations:

$$\mathbf{U} = f_{\mathcal{W}}(\mathbf{A}', \mathbf{X}), \ \mathbf{V} = f_{\mathcal{W}}(\mathbf{A}'', \mathbf{X}), \quad (5)$$

where $f_{\mathcal{W}}$ is the GNN encoder with trainable parameter set $\mathcal{W}$, $\mathbf{U}$ and $\mathbf{V}$ are the clean and perturbed view node representations, respectively. Existing GCL-based defenders are built upon the local-global Mutual Information (MI) maximization framework (Li et al. 2022b), which potentially makes several unreasonable assumptions in the context of adversarial attacks and has been found to compromise valuable semantic information. We instead consider a more fine-grained contrastive learning framework. Inspired by InfoNCE and previous GCL work (Zhu et al. 2020), we first construct a basic node contrastive loss, i.e., for node i in representation $\mathbf{U}$ selected as the anchor, we have

$$\ell_{f_{\mathcal{W}}}(\boldsymbol{u}_i) = \log \frac{\text{pos}_{f_{\mathcal{W}}}(\boldsymbol{u}_i)}{\text{pos}_{f_{\mathcal{W}}}(\boldsymbol{u}_i) + \text{neg}_{f_{\mathcal{W}}}(\boldsymbol{u}_i)}. \quad (6)$$

where the $\text{pos}_{f_{\mathcal{W}}}$ and $\text{neg}_{f_{\mathcal{W}}}$ are typically specified as

$$\begin{aligned} \text{pos}_{f_{\mathcal{W}}}(\boldsymbol{u}_i) &= e^{\kappa(\boldsymbol{u}_i, \boldsymbol{v}_i)/\tau}, \\ \text{neg}_{f_{\mathcal{W}}}(\boldsymbol{u}_i) &= \sum_{i \neq j} (\underbrace{e^{\kappa(\boldsymbol{u}_i, \boldsymbol{u}_j)/\tau}}_{\text{intra-view}} + \underbrace{e^{\kappa(\boldsymbol{u}_i, \boldsymbol{v}_j)/\tau}}_{\text{inter-view}}). \end{aligned} \quad (7)$$

where $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ are the embedding of node i from views $\mathcal{G}'$ and $\mathcal{G}''$, respectively. κ is a specified similarity measure

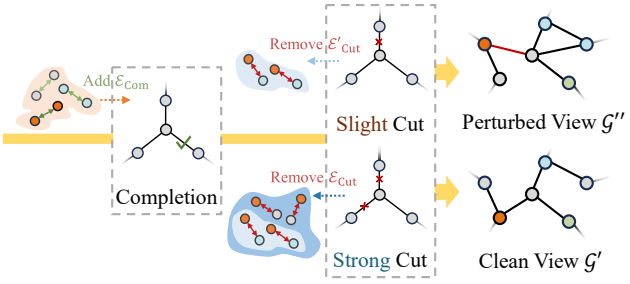


Figure 3: An illustration of semantic-based graph pre-processing, where nodes and bidirectional arrows with shading represent semantic information.

defined over node representations and τ is a temperature parameter. However, this strategy treats nodes as independent entities, which still leaves them vulnerable to corruption under adversarial attacks. Structural attacks aim to inject anomalous edges in a globally imperceptible manner, causing nodes to acquire incorrect neighborhood information via message passing, which leads to reversed prediction outcomes. The locality of graphs and GNNs makes them particularly susceptible to structural attacks. Conversely, by thoroughly analyzing local neighborhoods, we can construct a robust framework in response.

Therefore, in the neighborhood-based contrastive loss, the terms $\text{pos}_{f_{\mathcal{W}}}$ and $\text{neg}_{f_{\mathcal{W}}}$ can be improved as

$$\begin{aligned} &\text{pos}_{f_{\mathcal{W}}}(\boldsymbol{u}_i) = \\ &e^{\kappa(\boldsymbol{u}_i, \boldsymbol{v}_i)/\tau} + \underbrace{\sum_{j \in \mathcal{N}^{\text{t}}_{\boldsymbol{u}_i}} e^{\kappa(\boldsymbol{u}_i, \boldsymbol{u}_j)/\tau}}_{\text{topological}} + \underbrace{\sum_{j \in \mathcal{N}^{\text{s}}_{\boldsymbol{u}_i}} e^{\kappa(\boldsymbol{u}_i, \boldsymbol{u}_j)/\tau}}_{\text{semantic}}, \end{aligned} \quad (8)$$

$$\begin{aligned} &\text{neg}_{f_{\mathcal{W}}}(\boldsymbol{u}_i) = \\ &\underbrace{\sum_{j \notin \mathcal{N}^{\text{t}}_{\boldsymbol{u}_i} \cup \mathcal{N}^{\text{s}}_{\boldsymbol{u}_i}} e^{\kappa(\boldsymbol{u}_i, \boldsymbol{u}_j)/\tau}}_{\text{intra-view}} + \underbrace{\sum_{j \notin \mathcal{N}^{\text{t}}_{\boldsymbol{v}_i} \cup \mathcal{N}^{\text{s}}_{\boldsymbol{v}_i}} e^{\kappa(\boldsymbol{u}_i, \boldsymbol{v}_j)/\tau}}_{\text{inter-view}}. \end{aligned} \quad (9)$$

where $\mathcal{N}_{\boldsymbol{u}_i}$ denotes the neighborhood, i.e., the collection of neighbors of node i defined over graph $\mathcal{G}'$. $\mathcal{N}^{\text{t}}_{\boldsymbol{u}_i}$ and $\mathcal{N}^{\text{s}}_{\boldsymbol{u}_i}$ are topological and semantic neighborhoods respectively and we have $\mathcal{N}_{\boldsymbol{u}_i} = \mathcal{N}^{\text{t}}_{\boldsymbol{u}_i} \cup \mathcal{N}^{\text{s}}_{\boldsymbol{u}_i}$. This dual neighborhood contrastive learning objective fully considers neighborhood information across dual spaces, enabling the model to learn node representations that are robust to anomalous edges within neighborhoods in a simple and effective manner.

Neighborhood Reliability Estimation Although the aforementioned contrastive strategy provides foundational robustness, there remains a potential risk of the clean view's information being influenced by the perturbed view. Therefore, we explore how to protect as well as use the integrated feature information and relatively accurate structural information in the clean view to further enhance robustness. To achieve this, we define neighbor reliability within neighborhoods:

$$r_{i,j} = g_{\Phi}(\boldsymbol{u}_i, \boldsymbol{u}_j), \ \forall j \in \mathcal{N}_{\boldsymbol{u}_i}, \quad (10)$$

where $r_{i,j}$ is the reliability score of neighbor $\boldsymbol{u}_j$ to anchor $\boldsymbol{u}_i$ and g_Φ is an estimator with learnable parameter set Φ. The estimated reliabilities are leveraged to adjust the aforementioned dual neighborhood contrastive objective. Taking the objective on topological positive pairs in Equation (8) as an example, it is redefined as

$$\sum_{j\in\mathcal{N}^{\mathrm{t}}_{\boldsymbol{u}_i}} e^{h_{\kappa,g_\Phi}(\boldsymbol{u}_i,\boldsymbol{u}_j)\cdot/\tau} = \sum_{j\in\mathcal{N}^{\mathrm{t}}_{\boldsymbol{u}_i}} r_{i,j}\cdot e^{\kappa(\boldsymbol{u}_i,\boldsymbol{u}_j)/\tau}, \quad (11)$$

for convenience, where h_{κ,g_Φ} is a composite function of κ and g_Φ. Consequently, we obtain the final contrastive objective of $\boldsymbol{u}_i$:

$$\ell_{f_\mathcal{W},g_\Phi}(\boldsymbol{u}_i) = \log \frac{\mathrm{pos}_{f_\mathcal{W},g_\Phi}(\boldsymbol{u}_i)}{\mathrm{pos}_{f_\mathcal{W},g_\Phi}(\boldsymbol{u}_i)+\mathrm{neg}_{f_\mathcal{W}}(\boldsymbol{u}_i)}. \quad (12)$$

where we have

$$\mathrm{pos}_{f_\mathcal{W},g_\Phi}(\boldsymbol{u}_i) = e^{\kappa(\boldsymbol{u}_i,\boldsymbol{v}_i)/\tau} + \underbrace{\sum_{j\in\mathcal{N}^{\mathrm{t}}_{\boldsymbol{u}_i}} e^{h_{\kappa,g_\Phi}(\boldsymbol{u}_i,\boldsymbol{u}_j)/\tau}}_{\text{reliable topological}} + \underbrace{\sum_{j\in\mathcal{N}^{\mathrm{s}}_{\boldsymbol{u}_i}} e^{h_{\kappa,g_\Phi}(\boldsymbol{u}_i.\boldsymbol{u}_j)/\tau}}_{\text{reliable semantic}}, \quad (13)$$

This mechanism exploits feature semantics and clean topology, enabling the neighborhood contrastive learning to maintain resilience against neighborhood information influenced by anomalous edges, thereby enhancing robustness. Our robust dual neighborhood contrastive learning prompts the model to learn better representations and continuously refine the estimator. Together, these elements collaborate effectively to achieve refined representations on attacked graphs. For specific implementations of the estimator, we choose a simple soft clustering algorithm (Wilder et al. 2019), whose details are deferred to Appendix A[1].

Improving GNN Classifier The refinement module produces a robust representation $\mathbf{U}$. This section discusses improvements to the subsequent GNN classifier. A basic form of message passing in GNN classifiers is defined as

$$\boldsymbol{h}_i^{(l+1)} = (1-\varepsilon)\sum_{j\in\mathcal{N}_i}\gamma_{i,j}^{(l)}\boldsymbol{h}_j^{(l)} + \varepsilon\boldsymbol{h}_i^{(0)}, \quad (14)$$

where $\gamma_{i,j}^{(l)}$ is the weight for message between node i and j in the l-th layer. For the vanilla GCN, the weight is $\gamma_{i,j}^{(l)} = 1/\sqrt{d_i d_j}$, where the message from each neighbor is dependent on only the degrees. That may cause the survival adversarial edges to still affect the classifier. Existing studies suggest that the message passing in traditional GNNs can be viewed as a form of Laplacian smoothing (Li, Han, and Wu 2018): Solely relying on this approach, which indiscriminately smooths all node representations within a neighborhood, inevitably leads to the vulnerability of GNNs. Since we have already obtained high-quality node representation during the refinement stage, nodes have the potential to adaptively select the information they receive during the classification phase. Therefore, an adaptive reweighting factor $\eta_{i,j}^{(l)}$ is introduced, which leads to $\gamma_{i,j}^{(l)} = \eta_{i,j}^{(l)}/\sqrt{d_i d_j}$. Intuitively, we set its range between -1 and 1, allowing the messages passed through adversarial or noisy edges to be treated as "negative messages." Essentially, the key insight behind this improvement is to introduce Laplacian sharpening (Park et al. 2019), such that each node is able to adaptively balance between the smoothing and sharpening. For implementation, the factor is learned by a gating mechanism (refer to Appendix A). This simple manner fully leverages the refined representations and further enhances adversarial robustness and classification performance.

Training Objective In summary, the overall training objective that needs to be maximized is defined as follows,

$$\mathcal{L}(\mathbf{U},\mathbf{V}) = \frac{1}{2N}\sum_{i=1}^{N}(\ell_{f_\mathcal{W},g_\Phi}(\boldsymbol{u}_i) + \ell_{f_\mathcal{W},g_\Phi}(\boldsymbol{v}_i)). \quad (15)$$

To explore the effectiveness of this objective function, we provide an analysis of the connection between maximizing this objective and maximizing mutual information.

Theorem 1 *Given two node representations $\mathbf{U},\mathbf{V}\in\mathbb{R}^F$ obtained by encoding views $\mathcal{G}'$ and $\mathcal{G}''$, our proposed contrastive objective $\mathcal{L}(\mathbf{U},\mathbf{V})$ is a lower bound of mutual information between encoder input feature matrix $\mathbf{X}$ and node representations $\mathbf{U},\mathbf{V}$ from two graph views:*

$$\mathcal{L}(\mathbf{U},\mathbf{V}) \le I(\mathbf{X};\mathbf{U},\mathbf{V}). \quad (16)$$

The proof is deferred to Appendix A. Theorem 1 establishes the relationship between objective $\mathcal{L}(\mathbf{U},\mathbf{V})$ and mutual information $I(\mathbf{X};\mathbf{U},\mathbf{V})$. According to this theorem, the objective $\mathcal{L}(\mathbf{U},\mathbf{V})$ serves as a stricter lower bound on MI. Therefore, maximizing $\mathcal{L}(\mathbf{U},\mathbf{V})$ is equivalent to maximizing a lower bound on the mutual information shared between input node features $\mathbf{X}$ and learned representations $\mathbf{U},\mathbf{V}$, ensuring model convergence (Bachman, Hjelm, and Buchwalter 2019; Tian, Krishnan, and Isola 2020).

Experiments

Experiment Setup *Dataset:* We evaluate GRANCE[1] and baselines on several widely used real-world datasets, including Cora, Citeseer, Cora-ML, and BlogCatalog. *Compared Methods:* We compare GRANCE with ten representative methods to assess its robustness under adversarial attacks, including: 1) baseline methods: GCN (Kipf and Welling 2017) and GAT (Veličković et al. 2017); 2) state-of-the-art GNN defenders: RGCN (Zhu et al. 2019), Jaccard (Wu et al. 2019), Pro-GNN (Jin et al. 2020), SimPGCN (Jin et al. 2021), ElasticGNN (Liu et al. 2021), STABLE (Li et al. 2022b), Mid-GCN (Huang et al. 2023), and GraphRSP (Wang et al. 2024a). *Attack Methods:* The experiments are conducted under five attack strategies, including: 1) Non-targeted attack: Mettack (Zugner and Gunnemann 2019), PGD (Xu et al. 2019), DICE (Waniek et al. 2018), and Random attack; 2) Targeted Attack: Nettack (Zügner, Akbarnejad, and Günnemann 2018). See Appendix B for details.

[1]The appendix and code are available at https://github.com/shumanzhuang/GRANCE.

	Attacks / Ptb rate (%)	Clean	Poisoning (Acc%) 5	Poisoning (Acc%) 15	Poisoning (Acc%) 25	Evasion (Acc%) 5	Evasion (Acc%) 15	Evasion (Acc%) 25
Cora	GCN	83.51 (0.36)	76.65 (0.78)	66.18 (1.64)	47.34 (1.92)	79.88 (0.81)	78.86 (1.11)	77.21 (1.42)
	GAT	83.50 (0.72)	79.83 (0.67)	71.18 (1.31)	55.29 (0.80)	80.36 (0.78)	69.78 (1.48)	53.61 (1.44)
	RGCN	83.24 (0.31)	76.16 (0.39)	69.16 (0.73)	52.36 (1.06)	82.40 (0.37)	81.79 (0.45)	80.83 (0.57)
	Jaccard	81.84 (0.35)	79.28 (0.40)	72.66 (0.74)	62.75 (1.46)	79.34 (1.01)	78.33 (1.00)	76.63 (1.55)
	Pro-GNN	83.10 (0.40)	80.93 (0.57)	72.44 (0.59)	68.65 (0.64)	78.97 (1.24)	76.94 (1.36)	72.88 (0.72)
	SimPGCN	82.30 (0.46)	78.07 (0.53)	73.94 (1.03)	65.44 (1.76)	81.98 (0.57)	81.97 (0.57)	81.02 (0.46)
	ElasticGNN	84.62 (0.72)	82.06 (0.93)	73.17 (0.85)	65.32 (1.03)	–	–	–
	STABLE	84.46 (0.42)	81.15 (0.66)	78.48 (1.64)	75.89 (1.72)	82.56 (0.28)	80.96 (0.42)	80.98 (0.33)
	Mid-GCN	83.96 (0.45)	**82.31 (0.57)**	76.63 (0.75)	72.15 (1.03)	82.38 (0.90)	80.69 (0.73)	77.56 (1.29)
	GraphRSP	81.03 (0.37)	80.10 (0.23)	78.15 (0.31)	70.77 (0.46)	82.18 (0.17)	81.65 (0.24)	81.13 (0.30)
	GRANCE	**84.71 (0.40)**	81.33 (0.78)	**80.30 (0.39)**	**78.50 (0.92)**	**82.60 (0.75)**	**82.07 (1.09)**	**81.38 (1.28)**
Citeseer	GCN	71.63 (0.61)	70.29 (0.77)	64.36 (1.30)	56.31 (0.85)	69.38 (0.79)	68.32 (0.57)	67.08 (1.04)
	GAT	73.76 (0.69)	72.57 (0.74)	68.07 (1.24)	61.32 (1.03)	73.11 (1.32)	69.29 (1.26)	61.79 (0.64)
	RGCN	73.43 (0.20)	71.33 (0.31)	64.45 (0.60)	57.82 (0.93)	72.77 (0.51)	71.78 (0.40)	72.11 (0.31)
	Jaccard	72.23 (0.11)	71.27 (0.34)	67.18 (0.89)	61.37 (1.12)	69.06 (0.71)	68.89 (0.81)	67.58 (0.78)
	Pro-GNN	73.18 (0.18)	72.23 (0.32)	65.61 (1.05)	55.84 (0.63)	71.84 (0.24)	68.03 (1.23)	68.47 (1.25)
	SimPGCN	73.10 (0.75)	72.99 (0.84)	70.44 (1.56)	67.83 (3.21)	74.03 (0.74)	73.86 (0.61)	73.63 (0.84)
	ElasticGNN	73.71 (0.46)	72.54 (0.56)	71.08 (0.70)	62.87 (1.17)	–	–	–
	STABLE	74.07 (1.34)	73.75 (0.84)	72.86 (1.32)	71.17 (1.02)	73.77 (0.84)	73.66 (0.62)	73.52 (0.45)
	Mid-GCN	73.82 (0.39)	73.36 (0.16)	72.89 (0.56)	68.55 (1.31)	73.83 (0.18)	74.16 (0.22)	71.24 (0.26)
	GraphRSP	73.88 (0.62)	72.52 (0.35)	67.68 (0.50)	62.06 (0.71)	70.29 (0.21)	70.25 (0.20)	69.51 (0.31)
	GRANCE	**75.07 (0.61)**	**74.80 (1.05)**	**73.79 (0.56)**	**74.36 (0.53)**	**75.43 (0.74)**	**75.04 (0.42)**	**74.77 (0.51)**
Cora-ML	GCN	85.31 (0.23)	80.17 (0.30)	54.15 (0.58)	49.42 (0.64)	82.26 (0.56)	81.41 (0.84)	81.35 (0.65)
	GAT	85.52 (0.35)	81.45 (0.68)	57.53 (1.03)	45.41 (3.36)	81.03 (0.64)	57.41 (1.16)	45.60 (2.70)
	RGCN	85.66 (0.49)	81.48 (0.38)	55.93 (0.63)	50.85 (0.35)	82.18 (0.15)	56.05 (0.23)	46.02 (0.24)
	Jaccard	84.64 (0.34)	80.46 (0.51)	57.24 (1.45)	50.03 (0.42)	82.29 (1.24)	79.98 (0.68)	78.59 (0.84)
	Pro-GNN	85.38 (0.36)	83.32 (0.57)	53.57 (0.32)	51.32 (0.73)	81.59 (0.69)	72.91 (3.41)	67.34 (2.54)
	SimPGCN	85.32 (0.42)	83.34 (0.39)	76.69 (6.03)	69.67 (10.31)	**84.15 (0.25)**	81.96 (1.01)	81.74 (2.31)
	ElasticGNN	85.60 (0.84)	83.46 (1.05)	71.35 (1.69)	54.07 (0.42)	–	–	–
	STABLE	85.81 (0.34)	81.62 (0.23)	76.22 (0.49)	70.19 (3.11	82.01 (0.29)	81.18 (0.27)	79.59 (0.54)
	Mid-GCN	82.77 (0.39	78.14 (0.29)	74.13 (0.45)	71.27 (0.65	82.82 (0.39)	80.62 (0.60)	79.60 (0.33)
	GraphRSP	84.19 (0.17)	81.28 (0.22)	78.76 (0.25)	76.38 (0.62)	83.04 (0.47)	81.07 (0.18)	74.09 (0.36)
	GRANCE	**86.17 (0.30)**	**84.11 (0.20)**	**81.21 (0.40)**	**79.66 (0.70)**	83.36 (0.37)	**82.12 (0.63)**	**82.30 (0.79)**
BlogCatalog	GCN	85.96 (1.16)	69.51 (1.34)	48.65 (1.46)	36.51 (3.23)	70.84 (2.72)	50.50 (1.68)	39.67 (3.79)
	GAT	69.63 (1.64)	63.96 (2.05)	52.09 (9.03)	39.14 (6.06)	65.57 (1.55)	49.86 (11.04)	32.26 (5.45)
	RGCN	72.71 (2.87)	67.16 (3.20)	53.20 (7.09)	59.19 (8.93)	68.47 (2.36)	63.83 (1.74)	60.86 (1.72)
	Jaccard	–	–	–	–	–	–	–
	Pro-GNN	76.27 (1.34)	67.33 (2.03)	60.74 (3.45)	63.59 (5.32)	71.26 (0.78)	48.70 (0.59)	38.19 (3.36)
	SimPGCN	90.03 (0.22)	86.52 (0.30)	85.73 (0.21)	84.53 (1.20)	87.09 (1.05)	86.22 (1.13)	85.26 (3.76)
	ElasticGNN	88.05 (0.58)	87.74 (0.18)	82.88 (1.03)	80.54 (1.90)	–	–	–
	STABLE	86.21 (0.71)	80.55 (0.65)	70.80 (1.27)	67.37 (0.35)	84.34 (1.28)	76.56 (1.39)	70.99 (1.03)
	Mid-GCN	86.39 (0.69)	85.68 (1.28)	81.45 (6.27)	79.79 (5.72)	85.69 (1.97)	84.98 (1.92)	83.21 (3.88)
	GraphRSP	89.64 (1.86)	85.10 (1.63)	80.52 (2.66)	76.95 (3.45)	85.83 (1.25)	84.72 (1.30)	83.83 (2.22)
	GRANCE	**91.91 (0.20)**	**90.31 (0.17)**	**88.55 (0.43)**	**88.39 (0.38)**	**87.56 (0.20)**	**86.87 (0.26)**	**85.76 (0.42)**

Table 1: Node classification accuracy (mean% and standard deviation%) under different perturbation of Mettack. The top two results are highlighted in bold and underlined. Some methods encounter errors, with the corresponding results marked as "–".

Performance against Non-targeted Attacks We first present the classification accuracy of all the models against non-targeted adversarial attacks at varying perturbation rates. The results for the Mettack, in both poisoning (attack during training) and evasion (attack during testing) settings, are shown in Table 1. Based on these results, the following observations can be made: 1) GRANCE consistently outperforms other baselines against both attacks across most datasets, indicating its effectiveness. 2) GRANCE and STABLE show superior performance over most end-to-end defenders, validating the advantages of the GCL-based refining-classifying pipeline. Notably, GRANCE performed

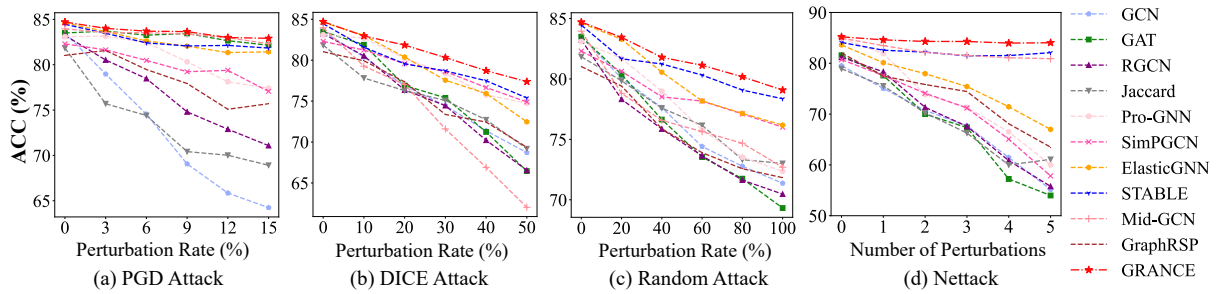
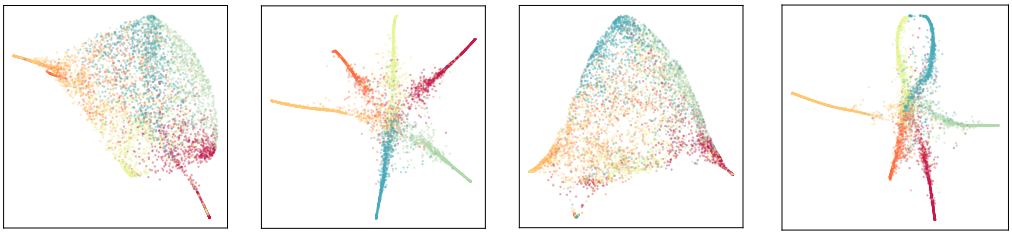


Figure 4: Accuracy on Cora under PGD, DICE, Random, and Nettack attack.



(a) $\text{STABLE}_{0\%}$ (b) $\text{GRANCE}_{0\%}$ (c) $\text{STABLE}_{25\%}$ (d) $\text{GRANCE}_{25\%}$

Figure 5: t-SNE visualization of node embedding in BlogCatalog dataset under Mettack.

especially well under severe perturbation ratios. We further evaluate the performance of GRANCE under additional non-targeted attacks, including PGD, DICE, and random attack. Using Cora dataset as an example, GRANCE exhibits slower accuracy degradation across varying attack rates and outperforms most competitors, as shown in Figure 4 (a)-(c).

Performance against Targeted Attacks. For the targeted attack experiments, we follow the default settings of (Jin et al. 2020) using Nettack (Zügner, Akbarnejad, and Günnemann 2018) to generate attacks. Specifically, we perturb the neighbors of each target node by incrementally increasing the number of adversarial edges from 1 to 5. Following (Jin et al. 2020), we select target nodes with a degree greater than 10 from the test set. In Figure 4 (d), we observe that GRANCE achieves the best performance on Cora dataset. Similar to the results under other attacks, the performance gains of GRANCE over baselines become more significant as perturbations increase. As the number of perturbations per target node grows, GRANCE maintains stable performance, showing effectiveness in fully utilizing correct neighborhood structures to defend against attacks.

Embedding Visualization To further evaluate our model's capabilities, we visualize the generated embeddings using the t-SNE (Van der Maaten L 2008) algorithm, as shown in Figure 5. We focus on both a non-adversarial scenario (ptb rate = 0%) and a heavily perturbed scenario (ptb rate = 25%) on BlogCatalog, which has the highest number of edges and a relatively large number of classes. We observe that the embeddings produced by GRANCE are more tightly clustered and exhibit clearer class boundaries, while those generated by STABLE are more dispersed with blurred class separations. This difference may arises from STABLE's limited ability to preserve semantic information and effectively leverage correct topological structures.

Parameter Sensitivity We conduct a sensitivity analysis of GRANCE under Mettack (25%) on Cora and Citeseer, focusing on its four key parameters: α, θ_2, c, and ε. Specifically, α controls the weight of semantic edges, θ_2 denotes the threshold for cutting, c is the number of clusters in soft-clustring-based estimator, and ε is used in the GNN classifier. Figure 6 (a)-(b) illustrate GRANCE's performance w.r.t. (α, θ_2) while c and ε is fixed. Key findings include: 1) Optimal performance occurs when α is between 0.2 and 0.3, highlighting the benefit of the semantic neighborhood, especially in heavily perturbed scenarios. 2) Any θ_2 improves performance compared to no threshold ($\theta_2 = 0$), affirming the effectiveness of semantic-based edge cutting. For c and ε, varying c from 5 to 30 and ε from 0 to 0.9 shows general robustness within specific ranges (Figure 6 (c)-(d)). Key observations are: 1) Performance generally improves as ε increases up to an optimal point, supporting the benefits of preserving semantic structure for robustness. 2) Performance stability across different values of c, suggests that GRANCE's effectiveness is not significantly impacted by the number of clusters in reliable estimation.

Ablation Study To validate the importance of each component in GRANCE, we perform an ablation study to assess their impact under Mettack (ptb rate $= 25\%$). Specifically, we construct four variants of GRANCE by removing individual components: **w/o ES** (without semantic-based edge supplementation), **w/o EC** (without semantic-based edge cutting), **w/o RNE** (without reliable neighborhood estima-

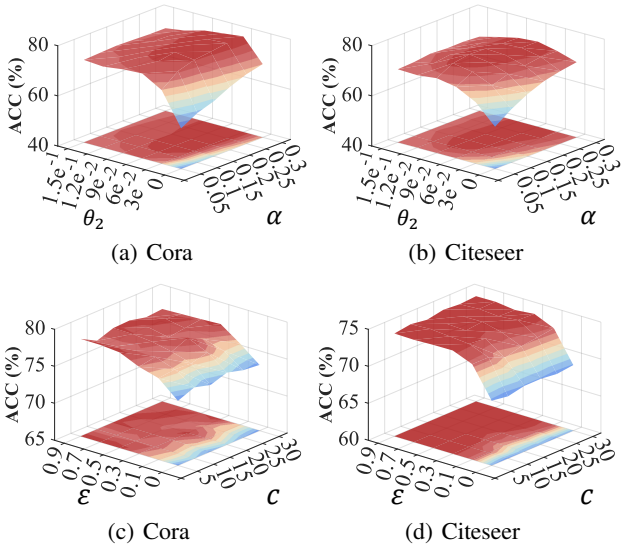


Figure 6: Parameter sensitivity on Cora and Citeseer.

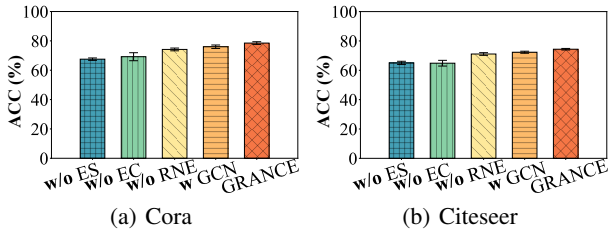


Figure 7: Ablation study (mean% and standard deviation%)

tion), and **w GCN** (using vanilla GCN as the classifier). Figure 7 presents the performance comparison of these variants with GRANCE on Cora and Citeseer. We observe: 1) The absence of semantic-based pre-processing leads to the most significant performance drop. This clearly illustrates the effectiveness of leveraging semantic information in defending against structural adversarial attacks. 2) Removing reliable neighborhood estimation also results in a noticeable performance decline. Such mechanism better utilizes correct topological information, allowing fine-grained evaluation that reduces interference from noisy adversarial edges. 3) Our classifier is more robust compared to vanilla GCN.

Related Work

Adversarial Attacks on Graphs Many studies have explored adversarial attacks of graph structures to expose the vulnerability of GNNs (Zügner, Akbarnejad, and Günnemann 2018; Waniek et al. 2018; Zugner and Gunnemann 2019; Xu et al. 2019; Geisler et al. 2021). These attack methods fall into two categories based on their attack targets: 1) *Non-targeted attacks* aim to degrade the model's performance on the test set by attacking the entire dataset. For instance, (Waniek et al. 2018) adopts the principle of internal disconnection and external connection. (Zugner and Gunnemann 2019) introduces a meta-learning-based global attack (Mettack) that treats graph structures as hyperparameters. (Xu et al. 2019) proposes a first-order topology attack (PGD) to identify minimal edge perturbations in the global graph structure. 2) *Targeted attacks* focus solely on perturbing specific target nodes to mislead the model. A prominent targeted attack method is Nettack (Zügner, Akbarnejad, and Günnemann 2018), which manipulates graph structures and node attributes while preserving degree distribution and feature co-occurrence. The above adversarial attacks pose significant challenges to GNNs, highlighting the critical need to design robust GNNs to withstand such attacks.

Adversarial Defense on Graphs Extensive work has been devoted to developing robust GNN-based models, we divide defense methods into three main categories. 1) *Refine and classify in-process:* These methods design robust architectures for GNNs, so that the refinement and classification are performed simultaneously. (Zhu et al. 2019) uses variance-based attention to mitigate adversarial influences, while (Zhang and Zitnik 2020) assesses neighbor importance via cosine similarity. (Liu et al. 2021) introduces an elastic message passing mechanism. But these models allow for interactions between representations and perturbed graph structures, inevitably contaminating feature information. 2) *Decoupling refining and classifying:* A more direct approach is to separately refine the graph structure before training. For example, (Wu et al. 2019) removes suspicious edges using Jaccard similarity. Similarly, (Entezari et al. 2020) applies low-rank approximations to reduce noise. (Zhu et al. 2023) employs bi-level structural learning to sanitize the input. Nonetheless, they cannot learn the topology adaptively. Given the lack of knowledge regarding adversarial edges, mis-deletions can undermine their effectiveness. Therefore, these methods, while preventing contamination, often under-utilize the uncontaminated parts of the structure. 3) *GCL-based Refining then classifying pipeline:* To overcome these drawbacks, GCL learns invariant information from perturbed graphs, which is being considered as a refining technique. (Li et al. 2022b) proposes an unsupervised method to optimize the graph structure, while Tao et al. (Tao et al. 2024) also adopts a GCL strategy to refine representations. However, these approaches fail to address the robustness of the unsupervised learning process itself.

Conclusion

In this paper, we reveal the inherent challenges of existing refining-classifying graph defense frameworks through both analytical and empirical studies. The widely adopted global-local strategies inevitably disrupt the intrinsic geometric structure in the feature space and fail to capture subtle attacks. Motivated by these findings, we propose GRANCE, a model that learns robust representations through reliable neighborhood contrastive learning. Extensive experiments on five adversarial attacks against state-of-the-art GNN defenders validate the robustness of GRANCE. One potential limitation of this work may be that we only consider first-order neighbors, and we plan to explore the incorporation of higher-order neighbors in future work.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grants No.62406070 and No.62102422, and the Open Topics from The Lion Rock Labs of Cyberspace Security under the project #LRL24006.

References

Bachman, P.; Hjelm, R. D.; and Buchwalter, W. 2019. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*.

Cai, J.; Zhang, Y.; Fan, J.; and Ng, S.-K. 2024. LG-FGAD: An Effective Federated Graph Anomaly Detection Framework. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 3760–3769.

Chen, Y.; Wu, Z.; Chen, Z.; Dong, M.; and Wang, S. 2023a. Joint learning of feature and topology for multi-view graph convolutional network. *Neural Networks*, 168: 161–170.

Chen, Z.; Wu, Z.; Lin, Z.; Wang, S.; Plant, C.; and Guo, W. 2024. AGNN: Alternating Graph-Regularized Neural Networks to Alleviate Over-Smoothing. *IEEE Transactions on Neural Networks and Learning Systems*, 35(10): 13764–13776.

Chen, Z.; Wu, Z.; Wang, S.; and Guo, W. 2023b. Dual low-rank graph autoencoder for semantic and topological networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4191–4198.

Entezari, N.; Al-Sayouri, S. A.; Darvishzadeh, A.; and Papalexakis, E. E. 2020. All you need is low (rank) defending against adversarial attacks on graphs. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, 169–177.

Gao, C.; Yin, S.; Wang, H.; Wang, Z.; Du, Z.; and Li, X. 2023. Medical-Knowledge-Based Graph Neural Network for Medication Combination Prediction. *IEEE Transactions on Neural Networks and Learning Systems*.

Geisler, S.; Schmidt, T.; Şirin, H.; Zügner, D.; Bojchevski, A.; and Günnemann, S. 2021. Robustness of graph neural networks at scale. In *Advances in Neural Information Processing Systems*, volume 34, 7637–7649.

Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In *Proceedings of the International Conference on Machine Learning*, 1263–1272.

Hickey, G.; Monlong, J.; Ebler, J.; Novak, A. M.; Eizenga, J. M.; Gao, Y.; Marschall, T.; Li, H.; and Paten, B. 2023. Pangenome graph construction from genome alignments with Minigraph-Cactus. *Nature Biotechnology*, 1–11.

Huang, J.; Du, L.; Chen, X.; Fu, Q.; Han, S.; and Zhang, D. 2023. Robust Mid-Pass Filtering Graph Convolutional Networks. In *Proceedings of the ACM Web Conference*, 328–338.

Jin, W.; Derr, T.; Wang, Y.; Ma, Y.; Liu, Z.; and Tang, J. 2021. Node similarity preserving graph convolutional networks. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, 148–156.

Jin, W.; Ma, Y.; Liu, X.; Tang, X.; Wang, S.; and Tang, J. 2020. Graph structure learning for robust graph neural networks. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 66–74.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations*.

Lao, D.; Yang, X.; Wu, Q.; and Yan, J. 2022. Variational inference for training graph neural networks in low-data regime through joint structure-label estimation. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 824–834.

Li, H.; Wang, S.; Chai, S.; Yang, Z.; Zhang, Q.; Xin, H.; Xu, Y.; Lin, S.; Chen, X.; Yao, Z.; et al. 2022a. Graph-based pan-genome reveals structural and sequence variations related to agronomic traits and domestication in cucumber. *Nature Communications*, 13(1): 682.

Li, K.; Liu, Y.; Ao, X.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2022b. Reliable representations make a stronger defender: Unsupervised structure refinement for robust gnn. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 925–935.

Li, Q.; Han, Z.; and Wu, X.-M. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Li, S.; Liu, Y.; Chen, Q.; Webb, G. I.; and Pan, S. 2024. Noise-Resilient Unsupervised Graph Representation Learning via Multi-Hop Feature Quality Estimation. In *Proceedings of the ACM International Conference on Information & Knowledge Management*.

Liu, X.; Jin, W.; Ma, Y.; Li, Y.; Liu, H.; Wang, Y.; Yan, M.; and Tang, J. 2021. Elastic graph neural networks. In *Proceedings of the International Conference on Machine Learning*, 6837–6849.

Liu, Y.; Li, S.; Zheng, Y.; Chen, Q.; Zhang, C.; and Pan, S. 2024. ARC: A Generalist Graph Anomaly Detector with In-Context Learning. In *Advances in Neural Information Processing Systems*.

Lu, J.; Wu, Z.; Chen, Z.; Cai, Z.; and Wang, S. 2024. Towards Multi-view Consistent Graph Diffusion. In *Proceedings of the ACM International Conference on Multimedia*, 186–195.

Pan, J.; Liu, Y.; Zheng, Y.; and Pan, S. 2023. PREM: A Simple Yet Effective Approach for Node-Level Graph Anomaly Detection. In *Proceedings of the IEEE International Conference on Data Mining*, 1253–1258.

Park, J.; Lee, M.; Chang, H. J.; Lee, K.; and Choi, J. Y. 2019. Symmetric graph convolutional autoencoder for unsupervised graph representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6519–6528.

Tao, Q.; Liao, J.; Zhang, E.; and Li, L. 2024. A Dual Robust Graph Neural Network Against Graph Adversarial Attacks. *Neural Networks*, 175: 106276.

Tian, Y.; Krishnan, D.; and Isola, P. 2020. Contrastive multiview coding. In *Proceedings of the European Conference on Computer Vision*, 776–794.

Van der Maaten L, H. G. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. In *Proceedings of the International Conference on Learning Representations*.

Velickovic, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep graph infomax. In *Proceedings of the International Conference on Learning Representations*.

Wan, X.; Liu, J.; Liu, X.; Wen, Y.; Yu, H.; Wang, S.; Yu, S.; Wan, T.; Wang, J.; and Zhu, E. 2024. Decouple then Classify: A Dynamic Multi-view Labeling Strategy with Shared and Specific Information. In *Proceedings of the International Conference on Machine Learning*, volume 235, 49941–49956.

Wang, H.; Zhou, C.; Chen, X.; Wu, J.; Pan, S.; Li, Z.; Wang, J.; and Philip, S. Y. 2024a. Graph Structure Reshaping Against Adversarial Attacks on Graph Neural Networks. *IEEE Transactions on Knowledge and Data Engineering*.

Wang, L.; Zheng, Y.; Jin, D.; Li, F.; Qiao, Y.; and Pan, S. 2024b. Contrastive graph similarity networks. *ACM Transactions on the Web*, 18(2): 1–20.

Wang, Y.; Liu, Y.; Shen, X.; Li, C.; Ding, K.; Miao, R.; Wang, Y.; Pan, S.; and Wang, X. 2024c. Unifying Unsupervised Graph-Level Anomaly Detection and Out-of-Distribution Detection: A Benchmark. *arXiv preprint arXiv:2406.15523*.

Waniek, M.; Michalak, T. P.; Wooldridge, M. J.; and Rahwan, T. 2018. Hiding individuals and communities in a social network. *Nature Human Behaviour*, 2(2): 139–147.

Wilder, B.; Ewing, E.; Dilkina, B.; and Tambe, M. 2019. End to end learning and optimization on graphs. In *Advances in Neural Information Processing Systems*.

Wu, H.; Wang, C.; Tyshetskiy, Y.; Docherty, A.; Lu, K.; and Zhu, L. 2019. Adversarial examples on graph data: Deep insights into attack and defense. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

Wu, Z.; Chen, Z.; Du, S.; Huang, S.; and Wang, S. 2024. Graph Convolutional Network with elastic topology. *Pattern Recognition*, 151: 110364.

Wu, Z.; Lin, X.; Lin, Z.; Chen, Z.; Bai, Y.; and Wang, S. 2023. Interpretable graph convolutional network for multi-view semi-supervised learning. *IEEE Transactions on Multimedia*, 25: 8593–8606.

Wu, Z.; Zhang, Z.; and Fan, J. 2024. Graph Convolutional Kernel Machine versus Graph Convolutional Networks. In *Advances in Neural Information Processing Systems*.

Xu, K.; Chen, H.; Liu, S.; Chen, P.-Y.; Weng, T.-W.; Hong, M.; and Lin, X. 2019. Topology attack and defense for graph neural networks: An optimization perspective. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

Yu, J.; Wu, Z.; Cai, J.; Jia, A. L.; and Fan, J. 2024. Kernel Readout for Graph Neural Networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2505–2514.

Zhang, X.; and Zitnik, M. 2020. Gnnguard: Defending graph neural networks against adversarial attacks. In *Advances in Neural Information Processing Systems*, 9263–9275.

Zhang, Y.; Sun, Y.; Cai, J.; and Fan, J. 2024. Deep Orthogonal Hypersphere Compression for Anomaly Detection. In *Proceedings of the International Conference on Learning Representations*.

Zheng, Y.; Pan, S.; Lee, V.; Zheng, Y.; and Yu, P. S. 2022a. Rethinking and scaling up graph contrastive learning: An extremely efficient approach with group discrimination. In *Advances in Neural Information Processing Systems*, volume 35, 10809–10820.

Zheng, Y.; Zhang, H.; Lee, V.; Zheng, Y.; Wang, X.; and Pan, S. 2023. Finding the missing-half: Graph complementary learning for homophily-prone and heterophily-prone graphs. In *Proceedings of the International Conference on Machine Learning*, 42492–42505.

Zheng, Y.; Zheng, Y.; Zhou, X.; Gong, C.; Lee, V. C.; and Pan, S. 2022b. Unifying graph contrastive learning with flexible contextual scopes. In *Proceedings of the IEEE International Conference on Data Mining*, 793–802.

Zhu, D.; Zhang, Z.; Cui, P.; and Zhu, W. 2019. Robust graph convolutional networks against adversarial attacks. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1399–1407.

Zhu, Y.; Lai, Y.; Zhao, K.; Luo, X.; Yuan, M.; Ren, J.; and Zhou, K. 2022. Binarizedattack: Structural poisoning attacks to graph-based anomaly detection. In *Proceedings of the IEEE International Conference on Data Engineering*, 14–26.

Zhu, Y.; Tong, L.; Li, G.; Luo, X.; and Zhou, K. 2023. FocusedCleaner: Sanitizing Poisoned Graphs for Robust GNN-based Node Classification. *IEEE Transactions on Knowledge and Data Engineering*.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2020. Deep graph contrastive representation learning. *arXiv*.

Zhuang, S.; Huang, S.; Huang, W.; Chen, Y.; Wu, Z.; and Liu, X. 2024. Enhancing Multi-view Graph Neural Network with Cross-view Confluent Message Passing. In *Proceedings of the ACM International Conference on Multimedia*, 10065–10074.

Zügner, D.; Akbarnejad, A.; and Günnemann, S. 2018. Adversarial attacks on neural networks for graph data. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2847–2856.

Zugner, D.; and Gunnemann, S. 2019. Adversarial Attacks on Graph Neural Networks via Meta Learning. In *Proceedings of the International Conference on Learning Representations*.