

# NORMALITY CALIBRATION IN SEMI-SUPERVISED GRAPH ANOMALY DETECTION

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

Graph anomaly detection (GAD) has attracted growing interest for its crucial ability to uncover irregular patterns in broad applications. Semi-supervised GAD, which assumes a subset of annotated normal nodes available during training, is among the most widely explored application settings. However, the normality learned by existing semi-supervised GAD methods is limited to the labeled normal nodes, often inclining to overfitting the given patterns. These can lead to high detection errors, such as high false positives. To overcome this limitation, we propose **GraphNC**, a graph normality calibration framework that leverages both labeled and unlabeled data to calibrate the normality from a teacher model (a pre-trained semi-supervised GAD model) jointly in anomaly score and node representation spaces. GraphNC includes two main components, anomaly score distribution alignment (**ScoreDA**) and perturbation-based normality regularization (**NormReg**). ScoreDA optimizes the anomaly scores of our model by aligning them with the score distribution yielded by the teacher model. Due to accurate scores in most of the normal nodes and part of the anomaly nodes in the teacher model, the score alignment effectively pulls the anomaly scores of the normal and abnormal classes toward the two ends, resulting in more separable anomaly scores. Nevertheless, there are inaccurate scores from the teacher model. To mitigate the misleading by these scores, NormReg is designed to regularize the graph normality in the representation space, making the representations of normal nodes more compact by minimizing a perturbation-guided consistency loss solely on the labeled nodes. Through comprehensive experiments on six benchmark datasets, we show that, by jointly optimizing these two components, GraphNC can 1) consistently and substantially enhance the GAD performance of teacher models from different types of GAD methods and 2) achieve new state-of-the-art semi-supervised GAD performance.

## 1 INTRODUCTION

Graph anomaly detection (GAD), aiming to identify irregular nodes in a graph, has gained increasing attention due to its critical role in real applications such as detection of spams in social networks and frauds in financial networks (Pang et al., 2021a; Ma et al., 2021; Qiao et al., 2025). Semi-supervised GAD, which leverages a subset of labeled normal nodes to learn normal patterns, has gained significant attention in real-world applications (Qiao et al., 2024). This GAD setting is practical in that normal nodes often account for the majority in a graph, and thus, it is significantly less costly to obtain compared to abnormal samples that are scarce and their occurrence typically involves high financial/reputation loss. However, compared to unsupervised GAD that is more popular, less studies have been done on semi-supervised GAD. Furthermore, existing semi-supervised GAD methods are primarily built on the limited labeled normal nodes, preventing them from learning the exact normal pattern, thus inclining to overfit the annotated normality (Ding et al., 2019; Wang et al., 2021; Ding et al., 2021; Fan et al., 2020; Qiao et al., 2024). As a result, these methods suffer from high detection errors—*e.g.*, normal nodes that are dissimilar to the labeled normal nodes are detected as anomalies (false positives) and vice versa (false negatives), especially the former one—due to the large overlapped anomaly scores for the normal and abnormal nodes. These issues can be observed in the results of a recent state-of-the-art (SOTA) method GGAD (Qiao et al., 2024) in Figs. 1 (a)

and (b). The same phenomenon can also be observed in other semi-supervised methods like data reconstruction and one-class classification-based methods (see App. C).

To address these issues, we propose **GraphNC**, a novel graph normality calibration framework that exploits both labeled and unlabeled data to calibrate the normality learned from a teacher model (*i.e.*, a pre-trained semi-supervised GAD model) jointly in anomaly score and representation spaces.

GraphNC includes two main components, namely anomaly score distribution alignment (**ScoreDA**) and perturbation-based normality regularization (**NormReg**). ScoreDA optimizes the anomaly scores of our model by aligning them with the score distribution yielded by the teacher model. Due to accurate scores in most of the normal nodes and part of the anomaly nodes in the teacher model, the score alignment effectively pulls the anomaly scores of the normal and abnormal classes toward the two ends, resulting in more separable anomaly scores, as shown in Fig.1 (d). This helps reduce false positive and false negative rates.

However, due to their inherent limitation mentioned above, the teacher models inevitably produce some inaccurate anomaly scores, which may mislead the score optimization in ScoreDA. To mitigate this issue, NormReg is devised to regularize the graph normality in the representation space, enabling the student model to refine itself by minimizing a perturbation-guided consistency loss solely on the labeled nodes. This normality calibration allows our model to learn more compact normal representations, which helps pull the anomaly scores of many normal nodes closer to each other and toward the lower end of the anomaly score distribution, as shown in Fig. 1 (e).

By jointly optimizing these two components, GraphNC learns more generalized normality representations and more discriminative anomaly scores. Besides, GraphNC is a generic framework where different pre-trained teacher models can be plugged into, and it can obtain better GAD performance if the teacher model is stronger. In summary, our contributions are as follows

- We introduce GraphNC, a novel graph normality calibration framework that leverages both labeled normal nodes and unlabeled data to calibrate normality learning from the teacher model in both the score and feature spaces for semi-supervised GAD .
- In GraphNC, we introduce two new components: anomaly score distribution alignment (ScoreDA) and perturbation-based normality regularization (NormReg). ScoreDA aligns the scores of our model with those of the pre-trained teacher model, which helps push the anomaly scores of the two classes toward opposite ends. Meanwhile, NormReg mitigates the impact of inaccurate teacher scores by enforcing representation compactness among normal nodes during alignment.
- GraphNC is a flexible framework where different teacher models can be plugged and played, achieving consistently enhanced GAD improvement across three types of teacher models, including data reconstruction-based, one-class-based, and anomaly-generation-based approaches. Further, GraphNC gets better performance when the teacher model is stronger. This is verified by our comprehensive experiments on six benchmark datasets.

## 2 RELATED WORK

**Graph Anomaly Detection.** Existing GAD methods can be broadly categorized into unsupervised, semi-supervised, and fully supervised approaches (Ma et al., 2021; Qiao et al., 2025). Unsupervised methods typically assume that no labeled nodes are known, and they learn normal patterns through

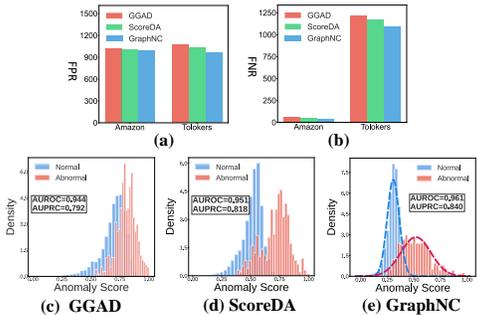


Figure 1: (a) False positive rate and (b) false negative rate results on Amazon (Dou et al., 2020) and Tokers (McAuley et al., 2015). (c), (d), and (e) show the score distributions of normal and abnormal nodes for GGAD, ScoreDA, and ScoreDA+NormReg (*i.e.*, GraphNC) on Amazon, where GGAD is used as a teacher model in both ScoreDA and GraphNC.

some proxy task designs such as reconstruction, one-class classification, and adversarial learning (Ding et al., 2019; Wang et al., 2021; Ding et al., 2021). Although this setting is widely used in anomaly detection, they overlook the abundance of normal samples, where normal nodes are less costly to obtain than the abnormal nodes. The fully supervised methods, on the other hand, rely on labeled normal and abnormal nodes and formulate the task as an imbalanced classification problem (Liu et al., 2021a; Tang et al., 2022; Gao et al., 2023a;b; Chen et al., 2024; Liu et al., 2025). These supervised methods rely on large labeled data, limiting their applicability in real-world scenarios, especially those where anomalies are very infrequent and very costly to collect; they may overfit labeled anomalies, failing to generalize to unseen anomalies (Wang et al., 2025). Semi-supervised methods are generally more practical for real-world applications, as only a subset of labeled normal nodes is required (Qiao et al., 2024; 2025). Only limited work has been done in this line. GGAD (Qiao et al., 2024) is a recent work that utilizes the labeled normal nodes to generate synthetic outliers that mimic real anomalies, and then trains a binary classifier on this data. It also shows that popular unsupervised methods, such as reconstruction, one-class classification, or adversarial learning-based methods, can be adapted to the semi-supervised settings by minimizing their loss on the normal data only.

**Semi-supervised Learning.** Semi-supervised learning aim to leverage a small amount of labeled data together with a large amount of unlabeled data to train models, aiming to achieve high performance while reducing the cost of data annotation (Yang et al., 2022; Chen et al., 2022b). Representative semi-supervised learning approaches include pseudo-labeling and consistency-based methods. The pseudo-labeling with confidence thresholding is highly successful and widely-adopted, which is typically built on the smoothness assumption that neighboring data points in the feature space tend to exhibit similar labels (Grandvalet & Bengio, 2004; Chen et al., 2023). However, such a mechanism requires a quantity–quality trade-off, which undermines the learning process, since incorporating more unlabeled data helps increase data coverage but often introduces noisy or unreliable pseudo-labels (Pham et al., 2021; Kage et al., 2024). The consistency-based model enforces prediction consistency under stochastic noise or data augmentation (Laine & Aila, 2016). Recent studies such as RankMatch (Mai et al., 2024), InterLUDE (Huang et al., 2024), and SCHOOL (Mo et al., 2024), have been proposed in different data modalities, including visual images, text, and graphs, aiming to encourage stable predictions across views, thereby enhancing generalization and reducing reliance on labeled data (Gui et al., 2024). However, they are mainly focused on classification tasks, different from our anomaly detection task, which has only one-class labels.

Many semi-supervised anomaly detection models that are trained on normal data have been proposed for anomaly detection tasks, but they are focused on visual data (Wu et al., 2024; Cao et al., 2024), failing to capture the complex structural information in the graph. There are also methods that aim to leverage small labeled anomaly data and large unlabeled data, such as DevNet (Pang et al., 2019), Deep SAD (Ruff et al., 2019), DPLAN (Pang et al., 2021b), PReNet (Pang et al., 2023), and RoSAS (Xu et al., 2023), which explore a different problem setting from ours.

### 3 PROBLEM STATEMENT

**Notations.** Given an attributed graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ , where  $\mathcal{V}$  denotes the node set with  $v_i \in \mathcal{V}$  and  $|\mathcal{V}| = N$  representing the total number of the node,  $\mathcal{E}$  denotes the edge set, and  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbb{R}^{N \times M}$  is a set of node attributes. Each node  $v_i$  has a  $M$ -dimensional attribute  $\mathbf{x}_i \in \mathbb{R}^M$ . The topological structure of  $\mathcal{G}$  is represented by an adjacency matrix  $\mathbf{A} \in \mathbb{R}^{N \times N}$ .

**Semi-supervised GAD.** Let  $\mathcal{V}_a, \mathcal{V}_n$  be two disjoint subsets of  $\mathcal{V}$ , where  $\mathcal{V}_a$  represents abnormal node set and  $\mathcal{V}_n$  represents normal node set, and typically the number of normal nodes is significantly greater than the abnormal nodes, *i.e.*,  $|\mathcal{V}_n| \gg |\mathcal{V}_a|$ , then the goal of semi-supervised GAD is to learn the mapping function  $f \rightarrow \mathbb{R}$ , such that  $f(v) < f(v')$ , where  $\forall v \in \mathcal{V}_n, v' \in \mathcal{V}_a$ , given a set of labeled normal nodes  $\mathcal{V}_l \subset \mathcal{V}_n$ , with  $|\mathcal{V}_l| = R$ , and no access to any labels of the abnormal nodes.  $\mathcal{V}_u = \mathcal{V}/\mathcal{V}_l$  is the set of the unlabeled nodes and used as test data.

**Graph Neural Networks for Representation Learning.** Graph Neural Networks (GNNs) are typically employed to learn the representation of each node due to their strong representation ability. This can be formulated as follows

$$\mathbf{H}^{(\ell)} = \text{GNN}(\mathbf{A}, \mathbf{H}^{(\ell-1)}; \mathbf{W}^{(\ell)}) \quad (1)$$

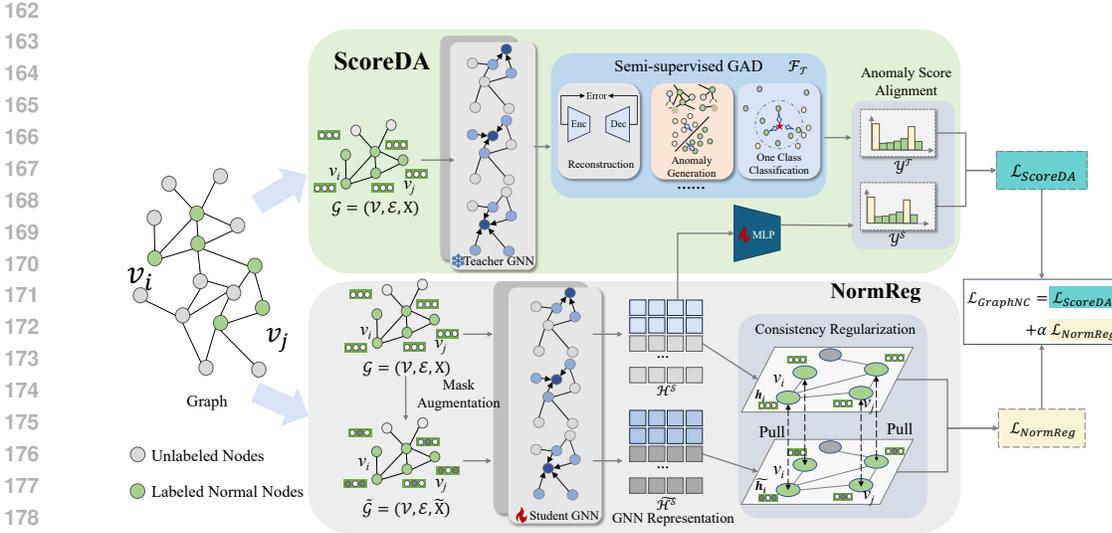


Figure 2: Overview of GraphNC. The input graph consists of a small labeled normal node set and a large unlabeled node set. GraphNC is based on a teacher-student network framework. ScoreDA aims to align the anomaly scores of the student network with the anomaly scores yielded by an existing semi-supervised GAD method to calibrate the normality in the score space. NormReg, on the other hand, is introduced to utilize a consistency regularization loss function in representation space to learn more compact representations of normal nodes, thereby reducing the negative impact of inaccurate anomaly scores that may be produced by the teacher model. The teacher network is pre-trained first, and it is frozen afterward, with only the student being trained.

where  $\mathbf{W}^{(\ell)}$  are the parameters, which are updated during training, and  $\mathbf{H}^{(\ell)} = \{\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_N\}$  denote the  $l$ -th (final) layer embeddings, which are the representation of all nodes at layer  $l$ . In this paper, our student model adopts a 2-layer GNN as the backbone for representation learning, while the teacher model uses the recommended GNN architecture in its original work.

## 4 METHODOLOGY

### 4.1 OVERVIEW OF THE PROPOSED GRAPHNC

The overview of the proposed GraphNC is shown in the Fig. 2. Given a pre-trained semi-supervised GAD model, it includes two steps during training. (1) Given a graph with only a set of labeled normal nodes, ScoreDA aligns the scores of our student model with the anomaly score distribution produced by the pre-trained detector as the teacher model, thereby achieving normality calibration in the score space. (2) NormReg subsequently complements score alignment by mitigating noise in the teacher model’s anomaly scores through a consistency constraint in the representation space on the labeled normal nodes, encouraging them to learn more compact normality representations. During inference, the output scores from the student model are used to derive the anomaly scores.

### 4.2 ANOMALY SCORE DISTRIBUTION ALIGNMENT (SCOREDA)

The simplistic normal pattern extraction in existing semi-supervised GAD models, trained on only a limited set of labeled normal samples, often is prone to overfit the patterns in the labeled data, resulting in high detection errors, such as elevated false positive rates, *i.e.*, many normal samples are misclassified as anomalies. We leverage ScoreDA to align the output scores of our model with the score distribution produced by the teacher model. As the anomaly score distribution of the pre-trained teacher model can correctly characterize most nodes, including most normal nodes and part of the abnormal nodes in the score space, by aligning with this teacher scores, the student model is

enforced to push scores for the normal and abnormal classes toward the two opposite ends, resulting in more separable anomaly scores. This effectively reduces the misclassification of normal samples, leading to a lower false positive rate, while also yielding a decrease in the false negative rate.

Formally, let  $\mathcal{F}_T$  be the pre-trained teacher model,  $\mathcal{Y}^T = \mathcal{F}_T(\mathbf{X}, \mathcal{G}; \Theta)$  be the output scores for the entire node set, where  $\Theta$  are the trainable parameters, and  $\mathcal{Y}^T = \{y_1^T, y_2^T, \dots, y_N^T\}$  be the anomaly score distribution of samples, then we leverage a student model  $\mathcal{F}_S$  with parameter  $\Phi$  which is implemented as the combination of GNN and MLP layers (*i.e.*,  $\mathcal{F}_{GNN}$  with parameters  $\Omega$  and  $\mathcal{F}_{MLP}$  with parameters  $\phi$  respectively) to align the score set  $\mathcal{Y}^T$ . For the student model, let  $\mathcal{H}^S$  be the output of the GNN student model, for a node  $v_i$ , the representation is then denoted by  $\mathbf{h}_i^S = \mathcal{F}_{GNN}(\mathbf{x}_i, \mathcal{G}; \Omega)$ . On top of that, we apply a MLP layer to obtain the predicted score,  $y_i^S = \mathcal{F}_{MLP}(\mathbf{h}_i^S; \phi)$ . Finally, we employ the MSE loss to minimize the discrepancy between the anomaly score from the student model  $\mathcal{Y}^S$  and  $\mathcal{Y}^T$  yielded by the teacher model, which is formulated as follows:

$$\mathcal{L}_{ScoreDA} = \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \|y_i^S - y_i^T\|_2^2, \quad (2)$$

where  $|\mathcal{V}|$  is the number of all nodes, including both labeled and unlabeled nodes. By minimizing the score discrepancy between the teacher and the student, the accurate scores in  $\mathcal{Y}^T$  helps enforce the clusters of the anomaly scores in the normal and abnormal classes in the two opposite ends, resulting in more separable anomaly scores.

### 4.3 PERTURBATION-GUIDED NORMALITY REGULARIZATION (NORMREG)

Due to the inherent limitations in normality characterization and the reliance on a limited set of normal samples, teacher models inevitably produce some inaccurate scores. Solely applying score alignment can enforce the fitting of the student model to these inaccurate scores, thereby misleading the optimization of the student model. To address this challenge, we further introduce NormReg to calibrate the learned normality in the representation space based on solely the labeled normal nodes, enabling the student to not only learn from the teacher but also self-refine.

To this end, we utilize an node attribute-based masking mechanism that randomly masks a proportion  $\omega$  of the attributes on labeled normal nodes to learn consistent representations for these nodes. This masking creates an augmented graph  $\tilde{\mathcal{G}}$ , which contains many variations of the labeled normal nodes, simulating diverse normal patterns that may be different from the ones derived directly from the labeled nodes. Thus, enforcing consistency over the representations of these augmented nodes and the labeled normal nodes helps cluster the normality from both the labeled nodes and their augmented versions closer. Formally, let  $\mathcal{H}_i^S$  and  $\tilde{\mathcal{H}}_i^S$  be the original and augmented node representations in the student model respectively, NormReg then minimizes the discrepancy between the representations of the original node and its augmented counterpart by optimizing the following loss function  $\mathcal{L}_{NormReg}$ :

$$\mathcal{L}_{NormReg} = \frac{1}{|\mathcal{V}_l|} \sum_{v_i \in \mathcal{V}_l} \|\mathbf{h}_i^S - \tilde{\mathbf{h}}_i^S\|_2^2, \quad (3)$$

where  $\mathcal{V}_l$  are the labeled normal nodes, and  $\mathbf{h}_i^S \in \mathcal{H}_i^S$  and  $\tilde{\mathbf{h}}_i^S \in \tilde{\mathcal{H}}_i^S$  are the original and augmented representations for node  $v_i$  respectively. As shown in Figs. 3 (a) and (b), by adding  $\mathcal{L}_{NormReg}$ , the representation distributions of the normal nodes obtained by GraphNC are more compact compared to GraphNC without using  $\mathcal{L}_{NormReg}$ . This demonstrates that the regularization effectively mitigates the negative impact of inaccurate anomaly scores from the teacher model. To provide more evidence, we also compute average deviation of normal nodes to the normal class prototype on four datasets, and it is clear that the deviation is significantly reduced after applying NormReg, as illustrated in Fig. 3 (c). Beyond the empirical demonstrations, our theoretical analysis also show that NormReg can well complement ScoreDA in lowering the detection errors.

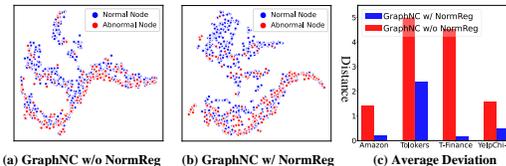


Figure 3: (a) and (b) provide t-SNE visualization of the node representations for GraphNC with/without using NormReg. (c) The average deviation of the normal class on Tolokers.

**Theorem 1.** *Let  $\sigma_T^2$  and  $\sigma_S^2$  denote the variance of the anomaly scores yielded by the teacher model and the student model in the normal class, respectively, then under our GraphNC framework, the student model achieves shrinking score variance (i.e.,  $\sigma_S^2 < \sigma_T^2$ ) by minimizing  $\mathcal{L}_{NormReg}$  while applying  $\mathcal{L}_{ScoreDA}$ , thereby reducing False Positive Rate (FPR) and False Negative Rate (FNR).*

The proof can be found in App. D. According to Theorem 1, minimizing  $\mathcal{L}_{NormReg}$  complements  $\mathcal{L}_{ScoreDA}$  in reducing the intra-class score variance of normal classes in the student model, which can also be observed when comparing the scores of the normal class in Fig. 1(e) to that in Fig. 1(d), lowering both FPR and FNR, especially the FPR.

#### 4.4 TRAINING AND INFERENCE

**Training.** During training, the student model is optimized by aligning its score distribution with that of the teacher model, while also minimizing the consistency regularization. To be specific, the total loss function  $\mathcal{L}_{GraphNC}$  is formulated as a combination of the  $\mathcal{L}_{ScoreDA}$  and  $\mathcal{L}_{NormReg}$ :

$$\mathcal{L}_{GraphNC} = \mathcal{L}_{ScoreDA} + \alpha \mathcal{L}_{NormReg}, \quad (4)$$

where  $\alpha$  is a hyper-parameter that adjusts the influence of normality consistency regularization in optimization.

**Inference.** During inference, since the student model  $\mathcal{F}_S$  calibrates the normality of the teacher and yields more accurate scores, its output scores can be directly employed as an anomaly score, which is defined as:

$$S(v_i) = \mathcal{F}_S(v_i, \mathbf{x}_i, \mathcal{G}, \Phi^*), \quad (5)$$

where  $\Phi^* = \{\Omega^*, \phi^*\}$  is the optimized parameter set of the student model.

## 5 EXPERIMENTS

**Datasets.** We evaluate the effectiveness of the GraphNC across six real-world GAD datasets drawn from diverse domains, including social networks: Reddit (Kumar et al., 2019)), online shopping co-review networks: Amazon (Dou et al., 2020) and YelpChi (Dou et al., 2020), a co-purchase network: Photo (Shchur et al., 2018), a collaboration network: Tolokers (McAuley et al., 2015), and financial networks: T-Finance (Tang et al., 2022). Additional description and statistical results are provided in App. A. Note that some unsupervised methods, such as CoLA (Liu et al., 2021b), GRADATE (Duan et al., 2023), and HUGE (Pan et al., 2025), cannot be adapted to semi-supervised setting due to their design more relying on fully unlabeled data, which are excluded from our comparison.

**Competing Methods.** The competing methods include reconstruction-based methods: DOMINANT (Ding et al., 2019), GADNR (Roy et al., 2024), ADA-GAD (He et al., 2024) and Anomaly-DAE (Fan et al., 2020), one-class classification: OCGNN (Wang et al., 2021), adversarial learning based methods: AEGIS (Ding et al., 2021) and GAAN (Chen et al., 2020), affinity maximization-based methods: TAM (Qiao & Pang, 2023), and anomaly generation-based method: GGAD (Qiao et al., 2024). A more detailed description of competing methods can be found in the App. B. Except GGAD that is specifically designed for the semi-supervised setting, the competing methods are primarily unsupervised methods, which are adapted to our semi-supervised setting following (Qiao et al., 2024): reconstruction is performed using only the normal nodes, the one-class center calculation is based on only the normal nodes, the adversarial discriminator is trained with the normal nodes, and the affinity maximization is also applied on the normal nodes solely.

**Evaluation Metric.** Following (Liu et al., 2021b; Qiao et al., 2024), two widely-used metrics, AUROC and AUPRC, are used to evaluate the performance of all methods. For both metrics, a higher value denotes a better performance. Moreover, for each method, we report the average performance after 5 independent runs with different random seeds.

**Implementation Details.** Our GraphNC model is implemented in Pytorch 1.10.0 with Python 3.8, and all the experiments are performed using GeForce RTX 3090 (24GB). GraphNC is optimized using the Adam optimizer (Kinga et al., 2015) with a learning rate of  $5e - 3$  for Photo and Reddit that have relatively high attribute dimensions, and  $5e - 4$  for Amazon, T-Finance, YelpChi, and Tolokers that have relatively low attribute dimensions, to avoid overfitting. The pre-trained teacher is set to GGAD (Qiao et al., 2024) by default, the experimental results of other teacher models can

Table 1: AUROC and AUPRC results on six real-world GAD datasets. For each dataset, the best performance per column within each metric is boldfaced, with the second-best underlined. ‘‘Avg’’ denotes the average performance of each method.

Metric	Method	Dataset						Avg.
		Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	
AUROC	DOMINANT	0.8867	0.6167	0.5194	<u>0.6314</u>	0.5121	0.5314	0.6162
	AnomalyDAE	0.9171	0.6027	0.5280	<u>0.6161</u>	0.6047	0.5272	0.6326
	OCGNN	0.8810	0.5742	0.5622	0.6155	0.4803	0.6461	0.6265
	AEGIS	0.7593	0.6728	0.5605	0.6022	0.4451	0.5936	0.6055
	GAAN	0.6531	0.3636	0.5349	0.5605	0.3785	0.4355	0.4876
	TAM	0.8405	0.5923	0.5829	0.6075	0.4847	0.6013	0.6182
	GADNR	0.6436	0.5943	0.5978	0.4554	<u>0.6406</u>	0.6158	0.5913
	ADA-GAD	0.2034	0.4416	0.5765	0.4400	0.5369	0.4067	0.4342
	GGAD	<u>0.9443</u>	<u>0.8228</u>	<u>0.6354</u>	0.6180	0.5340	<u>0.6476</u>	<u>0.7003</u>
	GraphNC	<b>0.9613</b>	<b>0.8340</b>	<b>0.6420</b>	<b>0.6630</b>	<b>0.6505</b>	<b>0.7693</b>	<b>0.7533</b>
AUPRC	DOMINANT	0.7289	0.0542	0.0414	0.2237	0.2217	0.1283	0.2330
	AnomalyDAE	0.7748	0.0538	0.0362	0.2064	0.2697	0.1177	0.2431
	OCGNN	0.6895	0.0492	0.0400	0.1937	0.2138	<u>0.1501</u>	0.2227
	AEGIS	0.2616	0.0685	0.0441	0.2192	0.1943	0.1110	0.1497
	GAAN	0.0856	0.0324	0.0362	0.1724	0.1693	0.0768	0.0945
	TAM	0.5183	0.0551	0.0446	0.2184	0.2178	0.1087	0.1938
	GADNR	0.0866	0.0523	0.0456	0.1324	<b>0.3511</b>	0.1430	0.1352
	ADA-GAD	0.0388	0.0375	0.0417	0.1295	0.2566	0.0723	0.0961
	GGAD	<u>0.7922</u>	<u>0.1825</u>	<u>0.0520</u>	<u>0.2261</u>	0.2502	0.1442	<u>0.2745</u>
	GraphNC	<b>0.8403</b>	<b>0.3667</b>	<b>0.0560</b>	<b>0.2389</b>	<u>0.3082</u>	<b>0.3561</b>	<b>0.3610</b>

be found in Sec. 5.2.  $\alpha$  that controls the importance of NormReg is set to 0.01 and the mask ratio for the augment mechanism is set 0.30 by default.

## 5.1 MAIN EXPERIMENTS

The main comparison results are shown in Table 11, where all models use 15% labeled normal nodes during training. From the results, we find that GGAD surpasses reconstruction-based, one-class classification, and adversarial learning approaches, suggesting that the classifier based on anomaly generation is particularly effective for learning the normality in semi-supervised GAD. We observe that GraphNC consistently outperforms all the semi-supervised methods on the six datasets, having an average 7.1% AUROC and 31.5% AUPRC improvement over the best competing method GGAD. These experimental results show that despite building upon GGAD, GraphNC consistently outperforms GGAD, indicating that the two components in GraphNC can effectively calibrate the normality learned from GGAD by aligning its scores with the output of GGAD, while at the same time refining itself through representation regularization to mitigate inaccurate anomaly scores from GGAD. Besides, although the reconstruction-based, one-class classification-based, and affinity maximized methods achieve competitive performance on Amazon, T-Finance, and YelpChi, they still largely underperform GraphNC. This is primarily because their modeling of normal patterns is less effective than that of the anomaly-generation-based method GGAD.

## 5.2 GRAPHNC ENABLED THREE TEACHER MODELS

To demonstrate that GraphNC is flexible and can be plugged into various teacher models, in addition to the default teacher model GGAD, we further evaluate the performance GraphNC when plugging the reconstruction-based method DOMINANT and the one-class classification-based method OCGNN as a pre-trained teacher model into our model. The AUROC and AUPRC results are shown in Table 3. It is clear from the results that, similar to the results using GGAD as the teacher model, GraphNC can substantially and consistently improve the performance of DOMINANT and OCGNN in terms of both AUROC and AUPRC. In particular, on Tolokers and Photo, both teacher models struggle to deliver strong performance, whereas applying GraphNC leads to substantial gains in both AUROC and AUPRC, resulting in a 36.2% and 26.3% AUROC improvement on Tolokers for DOMINANT and OCGNN, respectively. The main reason is that aligning the score distribution from the

Table 2: Results of GraphNC using different teacher models. (\*\*\*) denotes the teacher model used.

Metric	Method	Dataset						
		Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	Avg.
AUROC	DOMINANT	0.8867	0.6167	0.5194	0.6314	0.5121	0.5314	0.6162
	GraphNC (DOMINANT)	<b>0.8936</b>	<b>0.7985</b>	<b>0.5260</b>	<b>0.6543</b>	<b>0.6977</b>	<b>0.6829</b>	<b>0.7088</b>
	OCGNN	0.8810	0.5742	0.5622	0.6155	0.4803	0.6461	0.6265
	GraphNC (OCGNN)	<b>0.9255</b>	<b>0.6819</b>	<b>0.6093</b>	<b>0.6447</b>	<b>0.6069</b>	<b>0.7068</b>	<b>0.6956</b>
	GGAD	0.9443	0.8228	0.6354	0.6180	0.5340	0.6476	0.7003
	GraphNC (GGAD)	<b>0.9613</b>	<b>0.8340</b>	<b>0.6420</b>	<b>0.6630</b>	<b>0.6505</b>	<b>0.7693</b>	<b>0.7533</b>
AUPRC	DOMINANT	0.7289	0.0542	<b>0.0414</b>	0.2237	0.2217	0.1283	0.2330
	GraphNC (DOMINANT)	<b>0.7290</b>	<b>0.1703</b>	<b>0.0414</b>	<b>0.2375</b>	<b>0.3465</b>	<b>0.1526</b>	<b>0.2795</b>
	OCGNN	0.6895	0.0492	0.0400	0.1937	0.2138	0.1501	0.2227
	GraphNC (OCGNN)	<b>0.7372</b>	<b>0.0748</b>	<b>0.0512</b>	<b>0.2116</b>	<b>0.2814</b>	<b>0.1823</b>	<b>0.2564</b>
	GGAD	0.7922	0.1825	0.0520	0.2261	0.2502	0.1442	0.2745
	GraphNC (GGAD)	<b>0.8403</b>	<b>0.3667</b>	<b>0.0560</b>	<b>0.2389</b>	<b>0.3082</b>	<b>0.3561</b>	<b>0.3610</b>

Table 3: AUROC and AUPRC results comparison of GraphNC and its five variants.

Metric	Method	Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	Avg.
AUROC	OT	0.9443	0.8228	0.6354	0.6180	0.5340	0.6476	0.7003
	OT+NormReg	0.8956	<b>0.8455</b>	0.6317	0.6050	0.5843	0.6731	0.7058
	OT+NormReg-Finetune	0.9060	0.8008	0.6089	0.5729	0.5396	0.6307	0.6764
	OT+ScoreDA	<u>0.9511</u>	0.8300	<u>0.6364</u>	0.6263	0.5811	0.7397	0.7274
	OT+ScoreDA+NormReg*	0.9452	0.8278	0.6212	<u>0.6271</u>	<u>0.6492</u>	<u>0.7433</u>	<u>0.7356</u>
	OT+ScoreDA+NormReg	<b>0.9613</b>	<u>0.8340</u>	<b>0.6420</b>	<b>0.6630</b>	<b>0.6505</b>	<b>0.7693</b>	<b>0.7533</b>
AUPRC	OT	0.7922	0.1825	0.0520	<u>0.2261</u>	0.2502	0.1442	0.2745
	OT+NormReg	0.6568	0.2413	0.0520	0.2108	0.2688	0.1649	0.2657
	OT+NormReg-Finetune	0.7008	0.1406	0.0454	0.1820	0.2465	0.1310	0.2410
	OT+ScoreDA	<u>0.8189</u>	<u>0.2566</u>	<u>0.0530</u>	0.2111	0.2567	<u>0.2969</u>	<u>0.3155</u>
	OT+ScoreDA+NormReg*	0.8180	0.1971	0.0523	0.1866	<u>0.3034</u>	0.2697	0.3045
	OT+ScoreDA+NormReg	<b>0.8403</b>	<b>0.3667</b>	<b>0.0560</b>	<b>0.2389</b>	<b>0.3082</b>	<b>0.3561</b>	<b>0.3610</b>

pre-trained teacher model in GraphNC enables the student to calibrate the coarse normality learned by the teacher. This is because that GraphNC can not only inherit the strengths of the teacher but also correct its inaccuracies and refines normality on its own, resulting in consistent performance improvements. Importantly, given a better teacher model, GraphNC can obtain better GAD performance, *e.g.*, GraphNC (GGAD) vs. GraphNC (DOMINANT) and GraphNC (OCGNN).

### 5.3 ABLATION STUDY

In this section, we perform an ablation study to evaluate the contribution of each component in GraphNC. To this end, several variants of GraphNC are introduced. (1) Only Teacher (OT) directly uses the pre-trained semi-supervised teacher model to derive the anomaly score. (2) **OT+NormReg** incorporates NormReg into the pre-training of the teacher model. (3) **OT+NormReg+Finetune** leverages NormReg to fine-tune the pre-trained teacher. (4) **OT+ScoreDA** includes the student model and adds ScoreDA on top of OT (*i.e.*, optimizing GraphNC using ScoreDA only). (5) **OT+ScoreDA+NormReg\*** applies NormReg on top of OT+ScoreDA but it applies the NormReg loss on all nodes instead of the labeled normal nodes only. **OT+ScoreDA+NormReg** is the default full model using both ScoreDA and NormReg. The experimental results are shown in Table 3. We observe that OT+ScoreDA enhances the performance of OT, achieving the second-best results on four datasets in terms of AUPRC. This demonstrates that the ScoreDA component can effectively calibrate the normality learned from the teacher, ensuring better separation between normal and abnormal classes. However, OT+ScoreDA underperforms the full model (the last row) on all datasets, highlighting that the negative impact of inaccurate anomaly scores from the teacher model can be effectively mitigated by the NormReg component. On the other hand, applying NormReg directly to the teacher via either joint optimization (OT+NormReg) or finetuning (OT+NormReg+Finetune) can improve OT to some extent. This justifies the effectiveness of NormReg from a different perspective, but NormReg works best when working in the student model and combining with ScoreDA due to

its joint effects in minimizing the detection errors discussed in our theoretical analysis. Besides, we also conduct the ablation study on other teacher models to further analyze the effectiveness of each module in GraphNC. The results can be found in the App. C.2

### 5.4 HYPERPARAMETER SENSITIVITY ANALYSIS

We evaluate the sensitivity of GraphNC w.r.t. the loss moderator  $\alpha$ , mask ratio  $\omega$ , and training size  $R$ . A detailed analysis on time complexity can be found App. E.1.

**Performance w.r.t. the loss moderator  $\alpha$ .** As shown in Figs. 4 (a-b), our model GraphNC remains generally stable as the consistency regularization weight  $\alpha$  varies. However, on Tolokers and YelpChi, the performance slightly declines as  $\alpha$  increases. This is mainly because excessive emphasis on the consistency may lead to an overly compressed representation space, which may in turn weaken the representation discriminability.

**Performance w.r.t. the mask ratio  $\omega$ .** We evaluate GraphNC under different masking ratios. As shown in Figs. 4 (c-d), we observe that increasing the masking ratio  $\omega$  generally enhances the performance of GraphNC on Tolokers and T-Finance, while leading to performance degradation on Photo and YelpChi. This suggests that different datasets benefit from different levels of augmentation, depending on the variation in the underlying normality. For consistency, we set  $\omega$  to 0.3 across all datasets.

**Performance w.r.t. the training size  $R$ .** To explore the impact of training size  $R$ , we compare GraphNC with four semi-supervised GAD methods using varying numbers of training normal nodes, as shown in Figs.4 (e-f). Similar with other competing methods, with the increase of training size  $R$ , our model GraphNC generally performs better and it maintains the superiority over other competing methods, demonstrating the effectiveness of GraphNC using different scales of labeled training data.

## 6 CONCLUSION

In this paper, we propose GraphNC, a generic graph normality calibration framework designed to enhance existing semi-supervised GAD methods, achieving substantial and consistent performance improvements across different types of existing approaches. GraphNC is composed of two key components: ScoreDA and NormReg. The ScoreDA module aligns the anomaly scores with the anomaly score distribution produced by the teacher model, effectively pushing the anomaly scores for the normal and abnormal classes toward opposite ends. This alignment ensures a clearer separation between normal and abnormal nodes in the score space, thereby leading to a lower FPR and FNR. The NormReg module is introduced to reduce the impact of inaccurate scores in the anomaly score distribution served by the teacher, enabling GraphNC to learn more compact normal representations. GraphNC is comprehensively verified by empirical results on six GAD datasets, which show that GraphNC consistently enhances performance of three different types of teacher models, and it obtains better performance if the teacher model is stronger.

**Limitation and Future work.** A potential limitation of GraphNC is that it relies on an assumption that the pre-trained teacher model performs fairly well to provide informative supervision signals for the ScoreDA component. Future work will explore other sources, such as some labeled anomaly data or expert knowledge, as supplementary supervision to reduce its reliance on the teacher model.

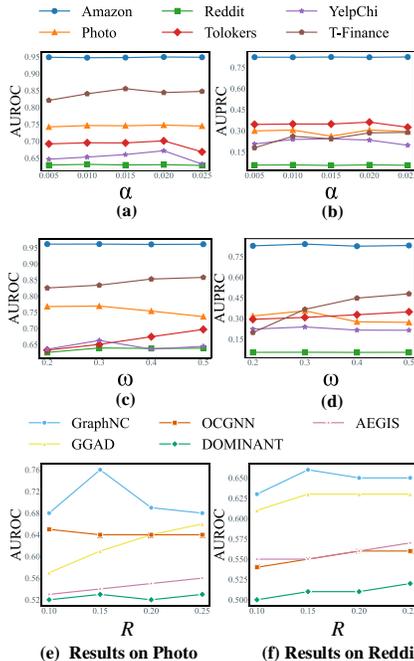


Figure 4: (a-d) AUROC and AURPC results w.r.t  $\alpha$  and  $\omega$ . (e-f) AUROC results w.r.t  $R$  on Photo and Reddit.

486 REPRODUCIBILITY STATEMENT  
487

488 We have taken several steps to ensure the reproducibility of our work. The datasets used in our exper-  
489 iments are all publicly available, and the construction and analysis of the data are presented in detail  
490 in App. A. The implementation details, including model configurations and training hyperparame-  
491 ters, are thoroughly documented in Sec. 5. The source code has been submitted as supplementary  
492 materials.

493  
494 REFERENCES  
495

- 496 Yunkang Cao, Xiaohao Xu, Jiangning Zhang, Yuqi Cheng, Xiaonan Huang, Guansong Pang, and  
497 Weiming Shen. A survey on visual anomaly detection: Challenge, approach, and prospect. *arXiv*  
498 *preprint arXiv:2401.16402*, 2024.
- 499 Bo Chen, Jing Zhang, Xiaokang Zhang, Yuxiao Dong, Jian Song, Peng Zhang, Kaibo Xu, Evgeny  
500 Kharlamov, and Jie Tang. Gccad: Graph contrastive coding for anomaly detection. *IEEE Trans-*  
501 *actions on Knowledge and Data Engineering*, 35(8):8037–8051, 2022a.
- 502  
503 Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj,  
504 and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised  
505 learning. *arXiv preprint arXiv:2301.10921*, 2023.
- 506  
507 Nan Chen, Zemin Liu, Bryan Hooi, Bingsheng He, Rizal Fathony, Jun Hu, and Jia Chen. Consis-  
508 tency training with learnable data augmentation for graph anomaly detection with limited super-  
509 vision. In *The twelfth international conference on learning representations*, 2024.
- 510  
511 Yanbei Chen, Massimiliano Mancini, Xiatian Zhu, and Zeynep Akata. Semi-supervised and un-  
512 supervised deep visual learning: A survey. *IEEE transactions on pattern analysis and machine*  
*intelligence*, 46(3):1327–1347, 2022b.
- 513  
514 Zhenxing Chen, Bo Liu, Meiqing Wang, Peng Dai, Jun Lv, and Liefeng Bo. Generative adversarial  
515 attributed network anomaly detection. In *Proceedings of the 29th ACM International Conference*  
516 *on Information & Knowledge Management*, pp. 1989–1992, 2020.
- 517  
518 Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed  
519 networks. In *Proceedings of the 2019 SIAM international conference on data mining*, pp. 594–  
602. SIAM, 2019.
- 520  
521 Kaize Ding, Jundong Li, Nitin Agarwal, and Huan Liu. Inductive anomaly detection on attributed  
522 networks. In *Proceedings of the twenty-ninth international conference on international joint*  
523 *conferences on artificial intelligence*, pp. 1288–1294, 2021.
- 524  
525 Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. Enhancing graph  
526 neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th*  
*ACM international conference on information & knowledge management*, pp. 315–324, 2020.
- 527  
528 Jingcan Duan, Siwei Wang, Pei Zhang, En Zhu, Jingtao Hu, Hu Jin, Yue Liu, and Zhibin Dong.  
529 Graph anomaly detection via multi-scale contrastive learning networks with augmented view. In  
530 *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 7459–7467, 2023.
- 531  
532 Haoyi Fan, Fengbin Zhang, and Zuoyong Li. Anomalydae: Dual autoencoder for anomaly detec-  
533 tion on attributed networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics,*  
*Speech and Signal Processing (ICASSP)*, pp. 5685–5689. IEEE, 2020.
- 534  
535 Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. Ad-  
536 dressing heterophily in graph anomaly detection: A perspective of graph spectrum. In *Proceed-*  
537 *ings of the ACM Web Conference 2023*, pp. 1528–1538, 2023a.
- 538  
539 Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. Al-  
leviating structural distribution shift in graph anomaly detection. In *Proceedings of the sixteenth*  
*ACM international conference on web search and data mining*, pp. 357–365, 2023b.

- 540 Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances*  
541 *in neural information processing systems*, 17, 2004.
- 542
- 543 Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. A survey  
544 on self-supervised learning: Algorithms, applications, and future trends. *IEEE Transactions on*  
545 *Pattern Analysis and Machine Intelligence*, 46(12):9052–9071, 2024.
- 546 Junwei He, Qianqian Xu, Yangbangyan Jiang, Zitai Wang, and Qingming Huang. Ada-gad:  
547 Anomaly-denoised autoencoders for graph anomaly detection. In *Proceedings of the AAAI Con-*  
548 *ference on Artificial Intelligence*, volume 38, pp. 8481–8489, 2024.
- 549
- 550 Zhe Huang, Xiaowei Yu, Dajiang Zhu, and Michael C Hughes. Interlude: Interactions between la-  
551 beled and unlabeled data to enhance semi-supervised learning. *arXiv preprint arXiv:2403.10658*,  
552 2024.
- 553 Patrick Kage, Jay C Rothenberger, Pavlos Andreadis, and Dimitrios I Diochnos. A review of pseudo-  
554 labeling for computer vision. *arXiv preprint arXiv:2408.07221*, 2024.
- 555
- 556 D Kinga, Jimmy Ba Adam, et al. A method for stochastic optimization. In *International conference*  
557 *on learning representations (ICLR)*, volume 5, pp. 6. San Diego, California;, 2015.
- 558 Srijan Kumar, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in tem-  
559 poral interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference*  
560 *on knowledge discovery & data mining*, pp. 1269–1278, 2019.
- 561 Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint*  
562 *arXiv:1610.02242*, 2016.
- 563
- 564 Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Pick and  
565 choose: a gnn-based imbalanced learning approach for fraud detection. In *Proceedings of the*  
566 *web conference 2021*, pp. 3168–3177, 2021a.
- 567
- 568 Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection  
569 on attributed networks via contrastive self-supervised learning. *IEEE transactions on neural*  
570 *networks and learning systems*, 33(6):2378–2392, 2021b.
- 571 Zemin Liu, Yuan Li, Nan Chen, Qian Wang, Bryan Hooi, and Bingsheng He. A survey of imbal-  
572 anced learning on graphs: Problems, techniques, and future directions. *IEEE Transactions on*  
573 *Knowledge and Data Engineering*, 2025.
- 574
- 575 Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z Sheng, Hui Xiong, and Leman  
576 Akoglu. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Trans-*  
577 *actions on Knowledge and Data Engineering*, 35(12):12012–12038, 2021.
- 578 Huayu Mai, Rui Sun, Tianzhu Zhang, and Feng Wu. Rankmatch: Exploring the better consistency  
579 regularization for semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF con-*  
580 *ference on computer vision and pattern recognition*, pp. 3391–3401, 2024.
- 581 Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based rec-  
582 ommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR*  
583 *conference on research and development in information retrieval*, pp. 43–52, 2015.
- 584
- 585 Yujie Mo, Zhihe Lu, Runpeng Yu, Xiaofeng Zhu, and Xinchao Wang. Revisiting self-supervised  
586 heterogeneous graph learning from spectral clustering perspective. *Advances in Neural Informa-*  
587 *tion Processing Systems*, 37:43133–43163, 2024.
- 588 Junjun Pan, Yixin Liu, Xin Zheng, Yizhen Zheng, Alan Wee-Chung Liew, Fuyi Li, and Shirui Pan.  
589 A label-free heterophily-guided approach for unsupervised graph fraud detection. In *Proceedings*  
590 *of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12443–12451, 2025.
- 591
- 592 Guansong Pang, Chunhua Shen, and Anton Van Den Hengel. Deep anomaly detection with deviation  
593 networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge*  
*discovery & data mining*, pp. 353–362, 2019.

- 594 Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for  
595 anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021a.  
596
- 597 Guansong Pang, Anton Van Den Hengel, Chunhua Shen, and Longbing Cao. Toward deep su-  
598 pervised anomaly detection: Reinforcement learning from partially labeled anomaly data. In  
599 *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp.  
600 1298–1308, 2021b.
- 601 Guansong Pang, Chunhua Shen, Huidong Jin, and Anton Van Den Hengel. Deep weakly-supervised  
602 anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discov-  
603 ery and Data Mining*, pp. 1795–1807, 2023.  
604
- 605 Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *Proceedings of the  
606 IEEE/CVF conference on computer vision and pattern recognition*, pp. 11557–11568, 2021.
- 607 Hezhe Qiao and Guansong Pang. Truncated affinity maximization: One-class homophily modeling  
608 for graph anomaly detection. In *Advances in Neural Information Processing Systems*, 2023.  
609
- 610 Hezhe Qiao, Qingsong Wen, Xiaoli Li, Ee-Peng Lim, and Guansong Pang. Generative semi-  
611 supervised graph anomaly detection. *Advances in neural information processing systems*, 37:  
612 4660–4688, 2024.
- 613 Hezhe Qiao, Hanghang Tong, Bo An, Irwin King, Charu Aggarwal, and Guansong Pang. Deep  
614 graph anomaly detection: A survey and new perspectives. *IEEE Transactions on Knowledge and  
615 Data Engineering*, 2025.
- 616 Amit Roy, Juan Shu, Jia Li, Carl Yang, Olivier Elshocht, Jeroen Smeets, and Pan Li. Gad-nr:  
617 Graph anomaly detection via neighborhood reconstruction. In *Proceedings of the 17th ACM  
618 international conference on web search and data mining*, pp. 576–585, 2024.  
619
- 620 Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-  
621 Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. *arXiv preprint  
622 arXiv:1906.02694*, 2019.
- 623 Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls  
624 of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.  
625
- 626 Jianheng Tang, Jiabin Li, Ziqi Gao, and Jia Li. Rethinking graph neural networks for anomaly  
627 detection. In *International Conference on Machine Learning*, pp. 21076–21089. PMLR, 2022.
- 628 Qizhou Wang, Guansong Pang, Mahsa Salehi, Wray Buntine, and Christopher Leckie. Open-set  
629 graph anomaly detection via normal structure regularisation. In *ICLR*, 2025.  
630
- 631 Xuhong Wang, Baihong Jin, Ying Du, Ping Cui, Yingshui Tan, and Yupu Yang. One-class graph  
632 neural networks for anomaly detection in attributed networks. *Neural computing and applications*,  
633 33:12073–12085, 2021.
- 634 Peng Wu, Chengyu Pan, Yuting Yan, Guansong Pang, Peng Wang, and Yanning Zhang. Deep  
635 learning for video anomaly detection: A review. *arXiv preprint arXiv:2409.05383*, 2024.  
636
- 637 Hongzuo Xu, Yijie Wang, Guansong Pang, Songlei Jian, Ning Liu, and Yongjun Wang. Rosas:  
638 Deep semi-supervised anomaly detection with contamination-resilient continuous supervision.  
639 *Information Processing & Management*, 60(5):103459, 2023.
- 640 Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning.  
641 *IEEE transactions on knowledge and data engineering*, 35(9):8934–8954, 2022.  
642  
643  
644  
645  
646  
647

## 648 A DETAILED DESCRIPTION OF DATASETS

649  
650 The key statistics of the datasets are presented in Table 4. A detailed introduction of these datasets  
651 is given as follows.

- 652 • Amazon (Dou et al., 2020): It is a co-review network obtained from the Musical Instrument  
653 category on Amazon.com. There are also three relations: U-P-U (users reviewing at least  
654 one same product), U-S-U (users having at least one same star rating within one week), and  
655 U-V-U (users with top-5% mutual review similarities).  
656
- 657 • T-Finance (Tang et al., 2022): It is a financial transaction network where the node repre-  
658 sents an anonymous account and the edge represents two accounts that have transaction  
659 records. Some attributes of logging like registration days, logging activities, and interac-  
660 tion frequency, etc, are used as the features of each account. Users are labeled as anomalies  
661 if they fall into categories such as fraud, money laundering, or online gambling
- 662 • Reddit (Kumar et al., 2019): It is a user-subreddit graph which captures one month’s worth  
663 of posts shared across various subreddits at Reddit. The node represents the users, and  
664 the text of each post is transformed into a feature vector and the features of the user and  
665 subreddits are the feature summation of the post they have posted. The anomalies are the  
666 used who have been banned by the platform
- 667 • YelpChi (Dou et al., 2020): Similar to Amazon-all, YelpChi-all incorporates three types  
668 of edges: R-U-R (reviews written by the same user), R-S-R (reviews of the same product  
669 with identical star ratings), and R-T-R (reviews of the same product posted within the same  
670 month). YelpChi-all is then constructed by merging these different relations into a single  
671 unified relation (Chen et al., 2022a; Qiao & Pang, 2023).
- 672 • Tolokers (McAuley et al., 2015): It is obtained from the Toloka crowdsourcing platform,  
673 where the node represents the person who has participated in the selected project, and the  
674 edge represents two workers work on the same task. The attributes of the node are the  
675 profile and task performance statistics of workers
- 676 • Photo (Shchur et al., 2018): It is an Amazon co-purchase network where nodes represent  
677 products and edges indicate co-purchase relationships. Each node is described by a bag-of-  
678 words representation derived from user reviews

## 679 B DESCRIPTION OF BASELINES

680 A more detailed introduction of the nine competing GAD models is given as follows.

- 681 • DOMINANT (Ding et al., 2019): It uses a three-layer GCN to reconstruct both structure  
682 and attributes, and the resulting reconstruction error on both structure and attribute is taken  
683 as the anomaly score.
- 684 • AnomalyDAE (Fan et al., 2020): It incorporates graph attention into the GNN-based struc-  
685 ture and attribute encoders, where the anomaly score is derived from the reconstruction  
686 error.
- 687 • OCGNN (Wang et al., 2021): It applies a one-class approach on the GNN, aiming to com-  
688 bine the representational power of the GNN with the anomaly detection capability of one-  
689 class classification. The anomaly score is computed by measuring the distance of each  
690 point from the center.
- 691 • AEGIS (Ding et al., 2021): It leverages a generative adversarial network (GAN), where the  
692 generator is designed to produce pseudo anomalies, while the discriminator distinguishes  
693 between genuine normal nodes and the generated anomalies.
- 694 • GAAN (Chen et al., 2020): It leverages a generative adversarial network where fake graph  
695 nodes are generated. Then the covariance matrix for real nodes and fake nodes are computer  
696 to enhance the node. Finally, a discriminator is trained to recognize whether two connected  
697 nodes are from a real or fake node.
- 698 • TAM (Qiao & Pang, 2023): It first reveals the ‘one class homophily’ and introduces the  
699 new anomaly measure score ‘local node affinity’. As the local node affinity can not be  
700  
701

Table 4: Key statistics of GAD datasets.

Datasets	Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo
#Nodes	11,944	39,357	10,984	45,941	11,758	7,484
#Edges	4,398,392	21,222,543	168,016	3,846,979	519,000	119,043
#Attributes	25	10	64	32	10	745
Anomaly	9.5%	4.6%	3.3%	14.52%	21.8%	4.9%

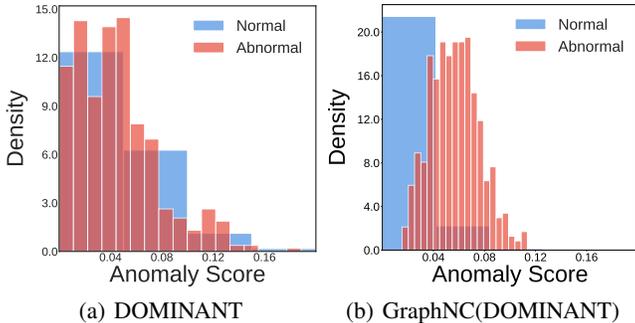


Figure 5: The score distribution of DOMINANT along the corresponding NomrDR enabled DOMINANT on Amazon (Dou et al., 2020)

directly obtained based on the embedding, it is optimized on truncated graphs where non-homophily edges are removed iteratively. The local node affinity is calculated on the learned representations on the truncated graphs.

- **GADNR (Roy et al., 2024):** It employs a GNN-based autoencoder to reconstruct each node’s attributes, degree, and Gaussian-approximated neighbor feature distribution, and uses the weighted sum of these reconstruction losses as the anomaly score.
- **ADA-GAD (He et al., 2024):** It adopts a two-stage anomaly-denoised autoencoder framework that first pretrains on anomaly-reduced augmented graphs and then retrains decoders on the original graph, taking the final reconstruction error with a distribution regularization term as the anomaly score.
- **GGAD (Qiao et al., 2024):** It employs two priors related to anomalies, asymmetric local affinity and egocentric closeness, to generate pseudo anomaly nodes that can well simulate the real abnormal nodes. Then, a binary one-class classifier is trained based on the existing normal nodes and generated outliers for semi-supervised GAD.

## C ADDITIONAL EXPERIMENTAL RESULTS

### C.1 SCORE DISTRIBUTION VISUALIZATION ON OTHER TEACHER MODEL

In this section, we visualized the score distributions of the reconstruction-based method DOMINANT and the one-class classification method OCGNN, along with their corresponding GraphNC-enabled models on the Amazon. As shown in the Fig. 5 and 6, we observe that the GraphNC-enabled model achieves better separation between normal and abnormal classes, further demonstrating that GraphNC can substantially and consistently enhance performance.

### C.2 ABLATION STUDY ON OTHER TEACHER MODELS

We also conduct the ablation study on the GraphNC-enabled other two teacher models to demonstrate the effectiveness of each module in the GraphNC. We employ the reconstruction-based method, DOMINANT, and one-class classification-based method, OCGNN as the teacher model and make a comparison with the variant of GraphNC. The experimental results are shown in the

756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809

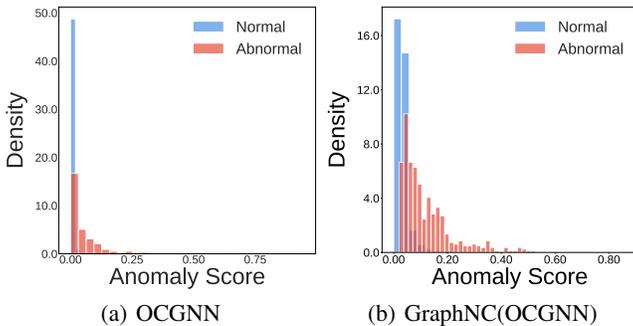


Figure 6: The score distribution of OCGNN along the corresponding GraphNC enabled OCGNN on Amazon (Dou et al., 2020)

Table. 5 We observe that GraphNC outperforms the corresponding variant on most of the datasets for both DOMINANT and OCGNN. This further demonstrates the flexibility and effectiveness of GraphNC in enhancing existing semi-supervised methods.

Table 5: AUROC and AUPRC results comparison of the proposed method GraphNC and its variants.

Metric	Method	Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	Avg.	
AUROC	DOMINANT								
		<b>0.8936</b>	<b>0.7985</b>	<u>0.5260</u>	<b>0.6543</b>	<b>0.6977</b>	<b>0.6829</b>	<b>0.7088</b>	
		0.7103	<u>0.7069</u>	0.4623	0.5125	0.5342	<u>0.6384</u>	0.5941	
		0.6054	0.4920	0.5192	0.6299	<u>0.6258</u>	0.5188	0.5651	
		0.7320	0.5041	<b>0.5968</b>	<u>0.6315</u>	0.5943	0.5128	0.5952	
		<u>0.8867</u>	0.6167	0.5194	0.6314	0.5121	0.5314	<u>0.6162</u>	
		OCGNN							
		<b>0.9255</b>	<b>0.6819</b>	<b>0.6093</b>	<b>0.6447</b>	<u>0.6069</u>	<b>0.7068</b>	<b>0.6956</b>	
		0.9158	0.4101	0.5235	<u>0.6411</u>	<b>0.6134</b>	<u>0.6913</u>	0.6325	
		0.8274	0.5960	0.4939	0.4738	0.5309	0.5588	0.5801	
		0.9095	<u>0.6113</u>	<u>0.5820</u>	0.6248	0.5245	0.6478	<u>0.6499</u>	
	0.8810	0.5742	0.5622	0.6155	0.4803	0.6461	0.6265		
AUPRC	DOMINANT								
		<b>0.7290</b>	<b>0.3703</b>	<u>0.0414</u>	<b>0.2375</b>	<b>0.3465</b>	<b>0.1526</b>	<b>0.2795</b>	
		0.1466	<u>0.1759</u>	0.0312	0.1581	0.2445	0.1274	0.1472	
		0.0750	0.0415	0.0330	0.2292	<u>0.2793</u>	0.1122	0.1283	
		0.1094	0.0424	<b>0.0486</b>	<u>0.2307</u>	0.2445	0.1066	0.1303	
		<u>0.7289</u>	0.0542	0.0414	0.2237	0.2217	<u>0.1283</u>	<u>0.2330</u>	
		OCGNN							
		<b>0.7372</b>	<b>0.0748</b>	<b>0.0512</b>	<b>0.2116</b>	<u>0.2814</u>	<b>0.1823</b>	<b>0.2564</b>	
		0.6298	0.0350	0.0327	<u>0.2112</u>	<b>0.2915</b>	<u>0.1782</u>	0.2297	
		0.4785	0.0554	0.0326	0.1382	0.2433	0.1093	0.1762	
		<b>0.7015</b>	<u>0.0567</u>	<u>0.0444</u>	0.2036	0.2415	0.1525	<u>0.2333</u>	
	0.6895	0.0492	0.0400	0.1937	0.2138	0.1501	<u>0.2227</u>		

### C.3 THE PERFORMANCE OF GRAPHNC WHEN THE TEACHER PERFORMS POORLY

To evaluate the performance of GraphNC when the teacher performs poorly ( $AUROC \leq 0.5$ ), we inject anomalies into the labeled normal nodes to deliberately degrade the teacher’s performance. When the teacher model performs poorly, we still apply score distribution alignment and NormReg to achieve normality calibration. The experimental results using GGAD as teacher model are shown in Table. 6.

Table 6: AUROC and AUPRC results of GraphNC and GGAD.

Metric	Method	Dataset						Avg.
		Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	
AUROC	GGAD	0.6336	0.5132	0.4483	0.5130	0.4504	0.4465	0.5008
	GraphNC	<b>0.7769</b>	<b>0.7450</b>	<b>0.5303</b>	<b>0.5685</b>	<b>0.6291</b>	<b>0.5970</b>	<b>0.6411</b>
AUPRC	GGAD	0.1171	0.0443	0.0276	0.1603	0.1946	0.0856	0.1049
	GraphNC	<b>0.2274</b>	<b>0.0930</b>	<b>0.0375</b>	<b>0.1823</b>	<b>0.2930</b>	<b>0.1145</b>	<b>0.1580</b>

Table 7: The comparison with GraphNC under the joint training with the teacher.

Metric	Method	Dataset						Avg.
		Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	
AUROC	GraphNC-Joint	0.9391	0.7631	0.5273	0.5654	0.4626	0.7441	0.6669
	GraphNC	<b>0.9613</b>	<b>0.8340</b>	<b>0.6420</b>	<b>0.6630</b>	<b>0.6505</b>	<b>0.7693</b>	<b>0.7534</b>

#### C.4 THE COMPARISON WITH GRAPHNC UNDER THE JOINT TRAINING WITH THE TEACHER

We have explicitly articulated that GraphNC is built upon a pre-trained semi-supervised GAD model (as the teacher model), indicating a sequential training of the teacher model and the GraphNC model (the student model). There are three fundamental reasons for this GraphNC design. i) Since GraphNC involves a teacher-student learning component (ScoreDA), it is assumed that we have a pre-trained teacher model so that the teacher can provide useful supervision signals for the student. ii) Motivated by the high false positives and false negatives in the teacher models, GraphNC (the student model) is designed specifically to reduce these two types of errors—validated through both theoretical analysis and empirical results—while leveraging the knowledge learned in the teacher models. iii) The separate training can provide a stable optimization trajectory for optimizing the student model. In contrast, joint training often obscures the optimal direction due to the large shifting in the supervision signals from the teacher model. For completeness, we also provide the results of GraphNC under the joint-training setting for comparison, as shown in Table. 7, which demonstrate a largely degraded performance if the training of the teacher and the student models is done jointly.

#### C.5 ANALYSIS ON TRUE ANOMALOUS INSTANCES RECONGINTION

To offer more insights into the dependency between our model and the teacher model, we provide statistics for the number of anomalous instances which are accurately detected by the teacher (GGAD) versus those detected by GraphNC in Table. 8. We can observe that the number of anomalous instances accurately detected by GraphNC is consistently and substantially higher than the teacher model across different datasets where the teacher model’s performance varies largely. This indicates that GraphNC can go far beyond the supervision from the teacher model to detect anomalies that the teacher model missed. We attribute this capability to the NormReg component and its synergy with the ScoreDA component.

#### C.6 STATISTICS OF THE AVERAGE AUROC IMPROVEMENT OF GRAPHNC W.R.T. DIFFERENT ACCURACY LEVELS

We summarize the average improvement of GraphNC over the teacher model across different teacher score ranges, as presented in Table. 9.

## D THEORETICAL ANALYSIS

**Theorem 1.** Let  $\sigma_T^2$  and  $\sigma_S^2$  denote the variance of the anomaly scores yielded by the teacher model and the student model in the normal class, respectively, then under our GraphNC framework, the student model achieves shrinking score variance (i.e.,  $\sigma_S^2 < \sigma_T^2$ ) by minimizing  $\mathcal{L}_{NormReg}$  while applying  $\mathcal{L}_{ScoreDA}$ , thereby reducing False Positive Rate (FPR) and False Negative Rate (FNR).

Table 8: The number of true anomalous instances that are accurately detected by the teacher (GGAD) versus those detected by GraphNC.

Method	Dataset						Avg.
	Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo	
GGAD	423	707	55	1186	378	150	483
GraphNC	<b>438</b>	<b>757</b>	<b>71</b>	<b>1352</b>	<b>455</b>	<b>214</b>	<b>548</b>

Table 9: Statistics of the average AUROC improvement of GraphNC w.r.t. different accuracy levels of supervision from the teacher model GGAD.

Teacher (GGAD)	0.4-0.5	0.5-0.6	0.6-0.7	0.7-0.9
GraphNC	0.1345	0.1128	0.0764	0.0204

*Proof.* We denote the teacher model’s score on the normal class by the random variable  $T$  (i.e.,  $S_T|y=0 \sim (\mu_0, \sigma_T^2)$ ), which follows a sub-Gaussian distribution, with a mean of  $\mu_0$  and a variance of  $Var(T) = \sigma_T^2$ . The law of total variance is given by

$$Var(T) = Var(\mathbb{E}[T|\mathbf{X}]) + \mathbb{E}(Var(T|\mathbf{X})), \quad (6)$$

where  $\mathbf{X}$  is the input feature. The optimal solution of the student model  $\mathcal{F}_S$  learned through soft score distillation with the input feature is  $\mathcal{F}_S^*(\mathbf{X}) = \mathbb{E}[T|\mathbf{X}]$ , then we have

$$Var(\mathcal{F}_S^*(\mathbf{X})) = Var(\mathbb{E}[T|\mathbf{X}]) \leq Var(T) = \sigma_T^2. \quad (7)$$

By optimizing the  $\mathcal{L}_{ScoreDA}$ , the variance of the predicted scores of the student model on normal nodes is usually not larger than that of the teacher model.

Furthermore, the designed  $\mathcal{L}_{NormReg}$  loss further shrinks the intra-class variance of normal classes on the student model.  $\mu_0$  can be regarded as the potential true center of the normal class, and then the representation of each normal node  $v_i$  in the two views of the student model can be written as  $h_i = \mu_0 + \epsilon_i$ ,  $\tilde{h}_i = \mu_0 + \tilde{\epsilon}_i$ , where  $\epsilon_i$  and  $\tilde{\epsilon}_i$  are the perturbations or noise introduced by two different views. We assume that the perturbations in both views are statistically independent and have zero mean, i.e.,  $\mathbb{E}[\epsilon_i] = \mathbb{E}[\tilde{\epsilon}_i] = 0$ . By substituting  $h_i$  and  $\tilde{h}_i$  into the  $\mathcal{L}_{NormReg}$ , we can redefine it as

$$\mathcal{L}_{NormReg} = \frac{1}{|\mathcal{V}_l|} \sum_{v_i \in \mathcal{V}_l} \|\epsilon_i - \tilde{\epsilon}_i\|_2^2 \quad (8)$$

By unfolding the loss and then computing its expectations, we obtain

$$\begin{aligned} \mathbb{E}[\|\epsilon_i - \tilde{\epsilon}_i\|_2^2] &= \mathbb{E}[\|\epsilon_i\|_2^2 + \|\tilde{\epsilon}_i\|_2^2 - 2\langle \epsilon_i, \tilde{\epsilon}_i \rangle] \\ &= \mathbb{E}[\|\epsilon_i\|_2^2] + \mathbb{E}[\|\tilde{\epsilon}_i\|_2^2] - 2\langle \mathbb{E}[\epsilon_i], \mathbb{E}[\tilde{\epsilon}_i] \rangle \end{aligned} \quad (9)$$

Since  $\epsilon_i$  and  $\tilde{\epsilon}_i$  are independent, and  $\langle \mathbb{E}[\epsilon_i], \mathbb{E}[\tilde{\epsilon}_i] \rangle = 0$ . We have,

$$\mathbb{E}[\|\epsilon_i - \tilde{\epsilon}_i\|_2^2] = \mathbb{E}[\|\epsilon_i\|_2^2] + \mathbb{E}[\|\tilde{\epsilon}_i\|_2^2] = \sigma_\epsilon^2 + \sigma_{\tilde{\epsilon}}^2, \quad (10)$$

where  $\sigma_\epsilon^2$  and  $\sigma_{\tilde{\epsilon}}^2$  are the population variances of the two views in the student model, respectively. By minimizing  $\mathcal{L}_{NormReg}$ , the node embeddings tend to be consistent across the two views (i.e., the consistent variances pattern) and the population variance decreases, making the normal node embeddings more compactly clustered around  $\mu_0$ . It is known that  $Var(\mathcal{F}_S^*(\mathbf{X}))$  corresponds to the variance of the normal class in the student model when considering only the distillation loss  $\mathcal{L}_{ScoreDA}$ . Therefore, when  $\mathcal{L}_{NormReg}$  is taken into account, the variance of the normal class in the student model satisfies  $\sigma_S^2 = Var(\mathcal{F}_S^*(\mathcal{H})) < Var(\mathcal{F}_S^*(\mathbf{X})) \leq \sigma_T^2$ , where  $\mathcal{H}$  is the output of the student model.

Since  $S_T|y=0$  follows a sub-Gaussian distribution, there exists  $\sigma_T$  such that for  $t > 0$  it satisfies

$$\mathbb{P}(S_T - \mu_0 \geq t) \leq \exp\left(-\frac{t^2}{2\sigma_T^2}\right) \quad (11)$$

Given an arbitrary fixed threshold  $\tau$  ( $\mu_0 < \tau$ ), if the predicted score of a sample exceeds this threshold, the sample is classified as abnormal. According to Eq. (11) and taking  $t = \tau - \mu_0$ , we have,

$$\mathbb{P}(S_S \geq \tau|y=0) \leq \exp\left(-\frac{(\tau - \mu_0)^2}{2\sigma_S^2}\right), \quad \mathbb{P}(S_T \geq \tau|y=0) \leq \exp\left(-\frac{(\tau - \mu_0)^2}{2\sigma_T^2}\right) \quad (12)$$

Table 10: Runtimes (in seconds) of training on the six datasets of GGAD and GraphNC.

Method	Dataset					
	Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo
GGAD	658	9345	368	480	850	106
GraphNC	60	3050	900	4200	19	89

Table 11: Runtimes (in seconds) of inference on the six datasets of four methods.

Method	Dataset					
	Amazon	T-Finance	Reddit	YelpChi	Tolokers	Photo
DOMINANT	37.8001	69.3396	1.3239	34.1469	2.7525	1.2359
OCGNN	0.1141	1.1454	0.0944	1.4889	0.1067	0.0502
GGAD	0.0991	1.0896	0.0867	1.4931	0.1023	0.0508
GraphNC	0.0951	1.0929	0.0785	1.5323	0.0934	0.0403

Since  $\sigma_S^2 < \sigma_T^2$ ,  $\mathbb{P}(S_s \geq \tau|y = 0) < \mathbb{P}(S_T \geq \tau|y = 0)$ . It indicates that the probability of the student model wrongly identifying an abnormal node as a normal node is lower than that of the teacher model, thereby reducing FNR.

Similarly, if the predicted score is below this threshold, the node is classified as normal. We compute the probabilities of the student model ( $\mathbb{P}(S_s < \tau|y = 0)$ ) and the teacher model ( $\mathbb{P}(S_T < \tau|y = 0)$ ) predicting it as a normal node as follows,

$$\mathbb{P}(S_s < \tau|y = 0) = 1 - \mathbb{P}(S_s \geq \tau|y = 0), \quad \mathbb{P}(S_T < \tau|y = 0) = 1 - \mathbb{P}(S_T \geq \tau|y = 0). \quad (13)$$

Since  $\mathbb{P}(S_s \geq \tau|y = 0) < \mathbb{P}(S_T \geq \tau|y = 0)$ , we have,

$$1 - \mathbb{P}(S_s \geq \tau|y = 0) \geq 1 - \mathbb{P}(S_T \geq \tau|y = 0). \quad (14)$$

Finally, we derive  $\mathbb{P}(S_s < \tau|y = 0) > \mathbb{P}(S_T < \tau|y = 0)$ , which can equivalently be expressed as  $\mathbb{P}(S_s < \tau|y = 1) < \mathbb{P}(S_T < \tau|y = 1)$ , given that the anomaly detection task uses binary labels ( $y = 0$  for normal and  $y = 1$  for abnormal). It indicates that the probability of misclassifying a normal node as abnormal is lower in the student model than in the teacher model, leading to a reduced FPR.  $\square$

## E COMPUTATIONAL EFFICIENCY ANALYSIS

### E.1 TIME COMPLEXITY ANALYSIS

In this section, we analyze the time complexity of GraphNC. Since it is designed as a plug-in framework that can be integrated with various teacher models to enhance GAD, we focus here only on the complexity of the additional module itself. We employ the combination of GCN and one MLP layer as the student model in GraphNC. The GCN takes  $\mathcal{O}(EM + NMd)$ , where  $E$  is the number of edges in the graph,  $N$  is the number of nodes,  $M$  is the dimension of the attribute, and  $d$  is the dimension of representation. The MLP used for feature transformation in GraphNC is  $\mathcal{O}(NMd)$ . In GraphNC, ScoreDA aims to align the student score with the output score distribution, which will take  $\mathcal{O}(N)$ . The NormReg aiming to minimize the distance between two views, which will also take  $\mathcal{O}(N)$ . The total complexity is  $\mathcal{O}(EM + 2NMd + 2N) + \mathbf{T}$ , where  $\mathbf{T}$  is the time complexity of the corresponding teacher model.

### E.2 RUNTIME RESULTS

We provide a runtime comparison between the standalone teacher model and the model augmented with GraphNC below. The training and inference time comparison are shown in Table. 10 and 11. It is clear that GraphNC is highly efficient for training, as its training process builds directly upon the pretrained teacher model. In addition, GGAD and GraphNC are both very efficient for inference.

## F ALGORITHM

The algorithm of GraphNC is summarized in Algorithm 1

972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025

---

**Algorithm 1** GraphNC
 

---

**Input:** Graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ ; Pre-trained Semi-supervised GAD Model  $\mathcal{F}_T$ ; Student Model  $\mathcal{F}_S$  with parameters  $\Phi$ ; Labeled Normal nodes  $\mathcal{V}_l \subset \mathcal{V}$ ; Number of training epochs  $E$ ; Trade-off weight  $\alpha$ .

**Output:** Anomaly score  $y_i^S$  for each node  $v_i \in \mathcal{V}$ .

- 1: Initialize parameters  $\Theta$  for the student model  $\mathcal{F}_S$ .
  - 2: Obtain the anomaly scores distribution  $\mathcal{Y}^T = \mathcal{F}_T(\mathbf{X}, \mathcal{G}; \Theta) = \{y_1^T, y_2^T, \dots, y_N^T\}$  from the teacher model.
  - 3: Create an augmented feature matrix  $\mathbf{X}'$  by applying RandomMask to features of nodes in  $\mathcal{V}_l$ .
  - 4: **for**  $e = 1$  to  $E$  **do**
  - 5:   **for**  $v_i$  in  $\mathcal{V}$  **do**
  - 6:     Obtain the representation for node  $v_i$ :  $\mathbf{h}_i^S \leftarrow \mathcal{F}_{GNN}(\mathbf{x}_i, \mathcal{G}, \mathbf{X}; \Omega)$ .
  - 7:     Obtain the augmented representation for node  $v_i$ :  $\tilde{\mathbf{h}}_i^S \leftarrow \mathcal{F}_{GNN}(\tilde{\mathbf{x}}_i, \tilde{\mathcal{G}}, \tilde{\mathbf{X}}; \Omega)$ .
  - 8:     Obtain the anomaly score of student model for node  $v_i$ :  $y_i^S = \mathcal{F}_{MLP}(\mathbf{h}_i^S; \phi)$ .
  - 9:   **end for**
  - 10:   // — *Anomaly Score Distribution Alignment (ScoreDA)* —  
 $\mathcal{L}_{ScoreDA} = \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \|y_i^S - y_i^T\|_2^2$ .
  - 11:   // — *Perturbation-based Regularization (NormaReg)* —  
 $\mathcal{L}_{NormReg} = -\frac{1}{|\mathcal{V}_l|} \sum_{v_i \in \mathcal{V}_l} \|\mathbf{h}_i^S - \tilde{\mathbf{h}}_i^S\|_2^2$ .
  - 13:   // — *Total Loss Function* —
  - 14:    $\mathcal{L}_{GraphNC} \leftarrow \mathcal{L}_{ScoreDA} + \alpha \cdot \mathcal{L}_{NormReg}$ .
  - 15:   Minimize  $\mathcal{L}_{GraphNC}$  and update parameters of student model,  $\Phi = \{\Omega, \phi\}$  by gradient descending.
  - 16: **end for**
  - 17: **return** Anomaly scores  $S(v_i) = \mathcal{F}_S(v_i, \mathbf{x}_i, \mathcal{G}, \Phi^*)$ , for each node  $v_i \in \mathcal{V}$
-