InfoNCE is a Free Lunch for Semantically guided Graph Contrastive Learning

Zixu Wang
State Key Laboratory of AI Safety,
Institute of Computing Technology,
Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
wangzixu22s@ict.ac.cn

Bingbing Xu*
State Key Laboratory of AI Safety,
Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
xubingbing@ict.ac.cn

Yige Yuan
State Key Laboratory of AI Safety,
Institute of Computing Technology,
Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
yuanyige20z@ict.ac.cn

Huawei Shen
State Key Laboratory of AI Safety,
Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
shenhuawei@ict.ac.cn

Xueqi Cheng
State Key Laboratory of AI Safety,
Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
cxq@ict.ac.cn

Abstract

As an important graph pre-training method, Graph Contrastive Learning (GCL) continues to play a crucial role in the ongoing surge of research on graph foundation models or LLM as enhancer for graphs. Traditional GCL optimizes InfoNCE by using augmentations to define self-supervised tasks, treating augmented pairs as positive samples and others as negative. However, this leads to semantically similar pairs being classified as negative, causing significant sampling bias and limiting performance. In this paper, we argue that GCL is essentially a Positive-Unlabeled (PU) learning problem, where the definition of self-supervised tasks should be semantically guided, i.e., augmented samples with similar semantics are considered positive, while others, with unknown semantics, are treated as unlabeled. From this perspective, the key lies in how to extract semantic information. To achieve this, we propose IFL-GCL, using InfoNCE as a "free lunch" to extract semantic information. Specifically, We first prove that under InfoNCE, the representation similarity of node pairs aligns with the probability that the corresponding contrastive sample is positive. Then we redefine the maximum likelihood objective based on the corrected samples, leading to a new InfoNCE loss function. Extensive experiments on both the graph pretraining framework and LLM as an enhancer show significantly improvements of IFL-GCL in both IID and OOD scenarios, achieving up to a 9.05% improvement, validating the effectiveness of semantically guided. Code for IFL-GCL is publicly available at: https://github.com/Camel-Prince/IFL-GCL.

*Corresponding author: Bingbing Xu

SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730007

CCS Concepts

• **Computing methodologies** → **Machine learning**.

Keywords

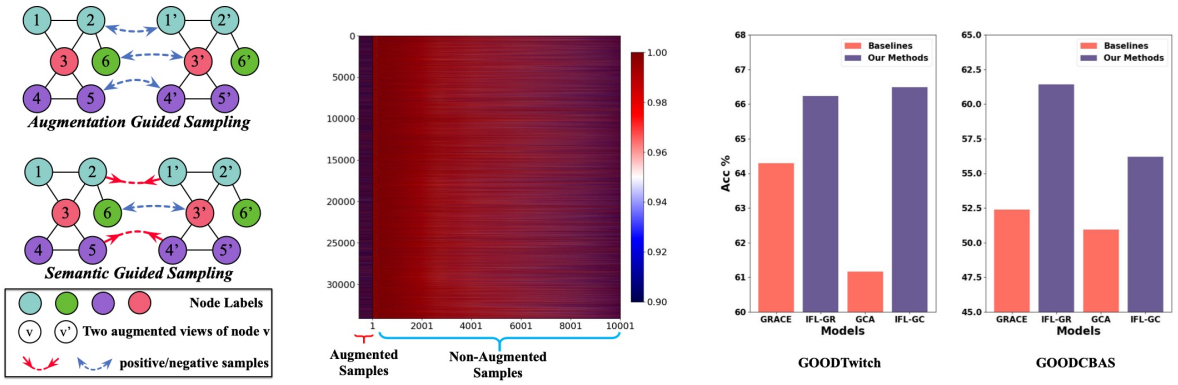
Graph Representation Learning; Graph Contrastive Learning; Positive Unlabeled Learning

ACM Reference Format:
Zixu Wang, Bingbing Xu, Yige Yuan, Huawei Shen, and Xueqi Cheng. 2025. InfoNCE is a Free Lunch for Semantically guided Graph Contrastive Learning. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3726302.3730007

1 Introduction

As a flexible and powerful data structure, graphs play a critical role in modeling domains such as social networks[2, 22, 32], chemical molecules[41], and transportation systems[28]. As a prominent paradigm for pre-training graph models, Graph Contrastive Learning(GCL) [9, 27, 34, 46, 48, 49] learns representations that effectively capture both feature and structural information in a self-supervised manner, which continues to play an indispensable role in the wave of graph foundation model. For example, graph contrastive learning is a commonly used self-supervised pre-training method in the paradigm where LLM serves as an enhancer and graph model as the core[5, 37].

Generally, traditional GCL methods optimize InfoNCE loss [25] to define self-supervised tasks [48, 49]. Specifically, they leverage graph data augmentation[19, 30] to generate different views of the graph and sample augmented pairs as positive samples and non-augmented pairs as negative samples. Returning to the core objective of GCL, it aims to bring samples with similar semantics closer together and push samples with dissimilar semantics farther apart[39]. The aforementioned approach essentially uses augmentations as a



(a) Illustration of Sampling Bias **(b) Validation of Sampling Bias** **(c) Harmfulness of Sampling Bias**

Figure 1: (a) illustrates the sampling bias via a case example; (b) validates sampling bias by comparing the nodes' representation similarity between augmented samples and top-20 non-augmented ones by a supervised graph encoder on GOODTwitch; (c) shows the harmfulness of sampling bias by comparing downstream task performances of traditional GCL baselines and our bias-corrected methods.

proxy for semantics, assuming that augmented samples share similar semantic information and can be treated as positive samples, while non-augmented samples are considered negative. However, detailed analysis and experimental validation reveal that traditional GCL methods suffer from sampling bias, limiting their performance [20, 47]. As shown in Fig.1 (a), in augmentation-guided sampling, non-augmented node pairs like $(node1, node2)$ and $(node4, node5)$ are treated as negative samples and pushed apart, despite their high semantic similarity (in terms of position, features, and structure). This prevents the model from correctly capturing semantic information in the original graph, leading to sampling bias and affecting downstream task performance. To validate this, we compare the node representation similarity between augmented and non-augmented samples in the real dataset GOODTwitch, as shown in Fig.1 (b), where some non-augmented samples exhibit even higher similarity than augmented ones. Furthermore, as shown in Fig.1 (c), this sampling bias harms GCL performance on downstream tasks.

To combat the above challenges, we argue that GCL is essentially a Positive-Unlabeled (PU) learning problem, where the training data consists of positive samples and unlabeled samples, with no explicit negative sample labels [1, 8, 12, 18, 38]. It aligns with the actual situation in GCL i.e., augmented samples with similar semantics are considered as labeled positive samples, while others with unknown semantics are treated as unlabeled. From this perspective, the key of GCL lies in extracting semantic information and leveraging it to resample contrastive samples for the self-supervised task.

To achieve the above objective, we propose IFL-GCL where "IFL" means using **I**nfoNCE as a "**F**ree-**L**unch" to extract semantic information and resample contrastive samples. Specifically, we discover that the representation similarity trained with InfoNCE [25] shares the same order with the probability of the contrastive sample being positive through theoretical analysis. In other words, InfoNCE brings "free-lunch" enabling us to extract semantic information and resample contrastive samples via representation similarity. Based on these, we redefine the maximum likelihood objective of InfoNCE instead of heuristically modifying the samples in InfoNCE, and naturally derive a new InfoNCE loss which demonstrates theoretically superior bias correction capabilities. Building on the above, IFL-GCL frames GCL within the PU learning framework, enabling semantically guided GCL, thereby mitigating sampling bias and enhancing GCL effectiveness.

To demonstrate the effectiveness of IFL-GCL, we validated its performance under two popular frameworks. In the graph pre-training framework, IFL-GCL show significantly improvements in both IID and OOD scenarios with an accuracy improvement up to 9.05% as shown in Fig.1 (c). When using LLM as an enhancer to porcess graph, IFL-GCL also brings consistent improvements showcasing its potential in graph foundation model research. In summary, our contributions are as follows:

- **Promising Way:** We first introduce a GCL framework based on PU learning to achieve semantic guidance rather than augmentation-based guidance.
- **Innovative Method:** We propose IFL-GCL to use InfoNCE as a "free-lunch" to extract semantic information for resampling contrastive samples based on theoretical analysis, then redefine the maximum likelihood objective of InfoNCE and naturally derive a new InfoNCE loss function which exhibits a stronger bias correction capability.
- **Extensive Experiments:** Extensive experimental results on both graph pre-training and LLM as enhancers show significantly improvements of IFL-GCL under both IID and OOD scenarios, achieving up to a 9.05% improvement, validating its effectiveness with semantical guidance.

2 Preliminaries

In this section, we present the formal process of graph contrastive learning (GCL) and validate the sampling bias in traditional GCL.

2.1 Graph Contrastive Learning

Due to the scarcity of labeled data, label-free self-supervised learning paradigms such as [11, 13, 14, 21, 26, 29, 40, 43, 45] have become

mainstream in the field of graph machine learning, with graph contrastive learning being one of the most widely studied branches [9, 15, 27, 34, 36, 48, 49].

Problem Formulation. Let $\mathbf{G} = (\mathbf{X}, \mathbf{A})$ denote a graph, where $\mathbf{X} \in \mathbb{R}^{N \times F}$ denotes the nodes' feature map, and $\mathbf{x}_i$ is the feature of i-th node n_i. $\mathbf{A} \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix, where $\mathbf{A}_{ij} = 1$ if and only if there is an edge from n_i to n_j. GCL aims at training a GNN encoder $f_\theta(\mathbf{G})$ that maps graph $\mathbf{G}$ into the node representations $\mathbf{H} \in \mathbb{R}^{N \times d}$ in a low-dimensional space, which captures the essential intrinsic information from both features and structure.

General Paradigm. First, two augmented views, $\mathbf{G}^{aug_1} = aug_1(\mathbf{G})$ and $\mathbf{G}^{aug_2} = aug_2(\mathbf{G})$, are obtained from the original input graph through two sampled graph data augmentation techniques, such as feature masking, edge dropping, and others [19, 30]. Next, we sample the contrastive samples from these two augmented graphs. In this paper, we take the node-node level contrast [36, 48, 49] as an example:

$$\begin{aligned} D^{aug+} &= \{(u_i, v_i), (v_i, u_i)\}_{i=1}^{N} \\ D^{aug-} &= \{(u_i, v_j), (v_i, u_j)\}_{i \neq j, i, j=1}^{N} \end{aligned} \tag{1}$$

where D^{aug+} means the set of augmented pairs which are sampled as positive samples and D^{aug-} means the set of other non-augmented pairs which are sampled as negative ones; u_i, v_j means the i-th node in the $\mathbf{G}^{aug_1}$ and j-th node in $\mathbf{G}^{aug_2}$ correspondingly. It's worth noting that (u_i, v_j) and (v_i, u_j) are two different contrastive samples which have different augmented graph combination order for the i-th and j-th nodes in the original graph. Finally, the contrastive loss function such as [4, 25, 33] is used to optimize the model, ensuring that the similarity between positive samples is high while the similarity between negative samples is low thereby learning feature representations with high discriminability that can quickly adapt to various downstream tasks. Take InfoNCE [25] loss for example:

$$l_{u_i, v_i} = -\log \frac{s_\theta(u_i, v_i)}{\sum_{j \neq i, j=1}^{N} s_\theta(u_i, u_j) + \sum_{j=1}^{N} s_\theta(u_i, v_j)} \tag{2}$$

$$L = \frac{1}{2N} \sum_{i=1}^{N} l_{u_i, v_i} + l_{v_i, u_i} \tag{3}$$

where $s_\theta(u_i, v_i)$ measures the representation similarity between node u_i and node v_j as follows:

$$s_\theta(u_i, v_j) = \exp(cos(\mathbf{U}_i, \mathbf{V}_j)/\tau) \tag{4}$$

where $\mathbf{U} = f_\theta(\mathbf{G}^{aug_1})$, $\mathbf{V} = f_\theta(\mathbf{G}^{aug_2})$ and $\mathbf{U}_i$ means the i-th node's representation in $\mathbf{G}^{aug_1}$, $\mathbf{V}_j$ means the j-th node's representation in $\mathbf{G}^{aug_2}$; $cos(\cdot, \cdot)$ is the cosine similarity function and τ is the temperature which controls the diameter of the representation space. l_{u_i, v_i} is the local loss produced by contrastive sample (u_i, v_i), L is the the final global loss which is the mean local loss of all samples of D^{aug+} shown in Eq.1.

2.2 Validation of Sampling Bias in GCL

In this section, we consider the node representation similarity learned under a large amount of supervision as an approximation of semantic similarity, and use it to verify that there are semantic similar samples among non-augmented ones which are classified as negative and cause sampling bias.

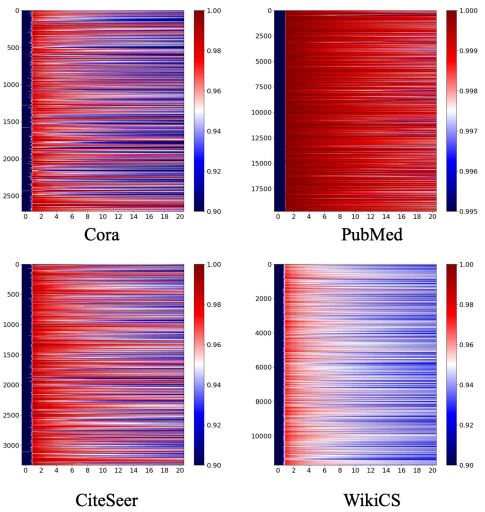


Figure 2: Semantics similarity matrix of $\mathbf{G}^{aug_1}$ and $\mathbf{G}^{aug_2}$ after supervised-learning which is rearranged with the D^{aug+} at the first column and decently sorted D^{aug-} as follows.

Specifically, we train a graph encoder $f_\theta(\cdot)$ using a supervised learning paradigm to capture the semantic information to the best extent:

$$\theta^* = \arg\min_{\theta} L_{sup}(f_\theta(\mathbf{G}), Y) \tag{5}$$

where $f_\theta(\cdot)$ is the graph neural network encoder: GAT[35], L_{sup} is the supervised loss function for node classification, $\mathbf{G}$ is the graph of real-word dataset: Cora, PubMed, CiteSeer or WikiCS [6, 23, 24, 31] and Y is the nodes' label set. Due to the extensive use of supervisory signals, the trained encoder can model node semantic information effectively. It is worth noting that although the encoder used here is trained for a specific task, the method of capturing semantic information through supervisory signals is applicable to any task.

Then we use $f_{\theta^*}(\cdot)$ to capture the semantic information of the augmented graph in GCL:

$$\mathbf{U}^* = f_{\theta^*}(\mathbf{G}^{aug_1}) \quad \mathbf{V}^* = f_{\theta^*}(\mathbf{G}^{aug_2})$$

We calculate the cosine similarity matrix $\mathbf{S}$ of $\mathbf{U}^*$ and $\mathbf{V}^*$:

$$\mathbf{S} = \frac{\mathbf{U}^* \cdot \mathbf{V}^{*T}}{\|\mathbf{U}^*\| \cdot \|\mathbf{V}^*\|^T}$$

where $\|\mathbf{U}^*\|$ is an $N \times 1$ vector containing the $L2$ norm of each row of matrix $\mathbf{U}^*$; $\mathbf{S}_{ij}$ means the cosine similarity between $\mathbf{U}^*{}_i$ and $\mathbf{V}^*{}_j$. For the convenience of observation, we rearranges $\mathbf{S}$ to get $\mathbf{S}'$ by extracting its main diagonal as the first column and sorting the remaining columns in descending order row by row:

$$\mathbf{S}' = \begin{bmatrix} \mathbf{S}_{11} & sorted(\mathbf{S}_{12}, \mathbf{S}_{13}, \cdots, \mathbf{S}_{1N}) \\ \mathbf{S}_{22} & sorted(\mathbf{S}_{21}, \mathbf{S}_{23}, \cdots, \mathbf{S}_{2N}) \\ \vdots & \vdots \\ \mathbf{S}_{NN} & sorted(\mathbf{S}_{N1}, \mathbf{S}_{N2}, \cdots, \mathbf{S}_{NN-1}) \end{bmatrix} \tag{6}$$

The rearranged similarity matrix with first column and top-20 remaining columns $\mathbf{S}'_{:,0:20}$ is extracted and shown in Fig.2. The

first column in S' shows the cosine similarity between nodes in augmented samples D^{aug+} and others corresponds to the top-20 most semantically similar ones in non-augmented samples D^{aug-}. From Fig.2, we observe that the first columns of the matrix exhibit a dark blue color, while the subsequent columns transition from red to blue. From such a distribution, it can be inferred that some node pairs in D^{aug-} exhibit semantic similarity either higher than (the red ones) or close to (the blue ones) that of the samples in D^{aug+}, proving our perspective that GCL with augmentation-guided sampling approach suffers from sampling bias because of the misclassified semantic similar non-augmented samples.

3 Methodology

In this section we propose IFL-GCL which uses **I**nfoNCE as a "**F**ree-**L**unch" to extract semantic information for resampling the contrastive samples, in order to correct the sampling bias in traditional GCL. We elaborate on the following topics: the first topic explains our motivation that GCL is a Positive-Unlabeled learning problem and clarify that the root cause of the sampling bias is semantically similar non-augmented samples. Then we focus on extracting semantic information and updating the training objective to correct the sampling bias. The third part discusses the theoretical superiority of our proposed method IFL-GCL.

3.1 Motivation

We point out that GCL is essentially a Positive-Unlabeled (PU) learning problem. We define two flag signs: semantic label $y = \{-1, +1\}$ and labeling status $o = \{-1, +1\}$, where $y(\mathbf{x}) = +1$ means $\mathbf{x}$ belongs to positive class and $y(\mathbf{x}) = -1$ means it's negative; $o(\mathbf{x}) = +1$ means $\mathbf{x}$ is labeled and vice versa. It is worth noting that y reflects the confirmed semantic properties of the data, while o just reflects samples' labeling status with unknown semantic information.

Traditional GCL methods substitute positive/negative samples with augmented/non-augmented ones:

$$\begin{aligned} D^{+} &= D^{aug+} \\ D^{-} &= D^{aug-} \end{aligned} \tag{7}$$

where $D^{+} = \{\mathbf{x}|y = +1\}$, $D^{-} = \{\mathbf{x}|y = -1\}$. In fact, augmented samples have confirmed semantically similarity and the semantic information of non-augmented samples remains unknown, therefore they should be viewed as follows:

$$\begin{aligned} D^{aug+} &= D_L^{+} \\ D^{aug-} &= D_U = D_U^{+} \cup D_U^{-} \end{aligned} \tag{8}$$

where $D_L^{+} = \{\mathbf{x}|y = +1, o = +1\}$,$D_U = \{\mathbf{x}|o = -1\}$ and the unlabeled samples can further be divided into unlabeled positive samples $D_U^{+} = \{\mathbf{x}|y = +1, o = -1\}$ and unlabeled negative samples $D_U^{-} = \{\mathbf{x}|y = -1, o = -1\}$ according to their semantics. PU learning focus on the tasks where data are split into positive labeled samples and unlabeled ones which is more aligned with the contrastive samples in traditional GCL. Therefore we treat GCL as a PU learning problem and reconsider the data composition as the following equations:

$$\begin{aligned} D^{+} &= D_L^{+} \cup D_U^{+} \\ D^{-} &= D_U - D_U^{-} \end{aligned} \tag{9}$$

Viewing GCL as a PU learning problem, it's evident that the semantic similar non-augmented samples D_U^{+} are misclassified as negative samples in traditional GCL which leads to the sampling bias.

3.2 Method

In this section, we propose IFL-GCL which uses infoNCE as a free-lunch to extract semantic information for resampling and update the training objective based on the resampling results.

3.2.1 ***Extracting Semantic Information to Resample.*** Firstly, we demonstrate the Invariance of Order (IOD) assumption and its corollary, introducing the key probability density ratio function $r(x)$. Next, we proof that InfoNCE in GCL exactly models $r(x)$. Finally, through $r(x)$ and the corollary of the Invariance of Order assumption, we obtain a classifier that allows us to extract D_U^{+} for correcting the sampling bias.

Invariance of Order Assumption and Corollary. Kato et al. [17] proposed a relaxed and universal assumption known as the **I**nvariance of **OrD**er (**IOD**) as follows:

$$\begin{aligned} &\forall \mathbf{x}, \hat{\mathbf{x}} \in D : \\ &p(y = +1|\mathbf{x}) \leq p(y = +1|\hat{\mathbf{x}}) \Leftrightarrow p(o = +1|\mathbf{x}) \leq p(o = +1|\hat{\mathbf{x}}) \end{aligned} \tag{10}$$

which means that the higher probability of a contrastive sample belongs to the positive class, the greater the likelihood that it will be labeled as positive, and vice versa. Kato et al. also derived the following significant corollary under the IOD assumption:

$$\begin{aligned} &\forall \mathbf{x}, \hat{\mathbf{x}} \in D : \\ &p(y = +1|\mathbf{x}) \leq p(y = +1|\hat{\mathbf{x}}) \Leftrightarrow r(\mathbf{x}) \leq r(\hat{\mathbf{x}}) \end{aligned} \tag{11}$$

where $r(\mathbf{x})$ is the distribution density ratio as following:

$$r(\mathbf{x}) := \frac{p(\mathbf{x}|y = +1, o = +1)}{p(\mathbf{x})} \tag{12}$$

According to Eq.11 and Eq.12, the of density ratio between data $D_L^{+} = \{\mathbf{x}|y = +1, o = +1\}$ and $D = \{\mathbf{x}\}$ reflects the order of the positive class posterior probability $p(y = +1|\mathbf{x})$. Therefore, we can derive a classifier $h(\cdot)$ which can be used to find semantically similar samples after modeling the $r(\cdot)$ and given a threshold t_r as follows:

$$y(\mathbf{x}) = h(\mathbf{x}) := sign(r(\mathbf{x}) - t_r) \tag{13}$$

where $y(\mathbf{x})$ is the class label y of contrastive sample $\mathbf{x}$.

Free-Lunch Provided by InfoNCE. Instead of employing traditional density ratio estimation methods such as LSIFs[10, 16, 44] to model $r(\mathbf{x})$, we astutely note that the InfoNCE loss function used in Eq.2 inherently provides a "free-lunch": the representations similarity trained under InfoNCE is proportional to the density ratio $r(\mathbf{x})$. We will now proof this "free-lunch".

Firstly, we notice that the initial design of InfoNCE in [25] was motivated by the aim to model a density ratio as follows:

$$f_k(x_{t+k}|c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \tag{14}$$

where $\propto$ denotes "proportional to", x_{t+k} is the future observations, c_t is the anchored sample and $f_k(\cdot, \cdot)$ is the model which targets at preserves the mutual information between x_{t+k} and c_t. Under the context of GCL, the $f_k(\cdot, \cdot)$ corresponds to $s_\theta(\cdot, \cdot)$ in Eq.4. Generally speaking, given a node pair (n, n') where n serve as the anchor, the

$s_\theta(\cdot,\cdot)$ learned by GCL under InfoNCE essentially models a nodes' probability density ratio as follows:

$$s_\theta(n,n') \propto \frac{q(n|n')}{q(n)} = \frac{q(n,n')/q(n')}{q(n)} = \frac{q(n,n')}{q(n)q(n')} \tag{15}$$

where $q(\cdot)$ is the probability density of node.

Next, we connect the contrastive samples' distributions and the nodes' distributions. Considering the labeled positive contrastive samples $\mathbf{x} = (n,n')$ with $y=+1, o=+1$ which is used to train $s_\theta(\cdot,\cdot)$, they are ***non-independent*** pairs of nodes connected together by data augmentation. Thus the distribution of labeled positive contrastive samples can be represented as the joint distribution of the node pair:

$$p(\mathbf{x}=(n,n')|y=+1,o=+1) = q(n,n') \tag{16}$$

Conversely, if a contrastive sample $\mathbf{x}=(n,n')$ is constructed without any information regarding y and o, it is obtained by ***independently*** sampling two nodes n and n', which can be expressed as:

$$p(\mathbf{x}=(n,n')) = q(n)q(n') \tag{17}$$

By combining Eq.17, Eq.16 and Eq.12, we obtain:

$$s_\theta(n,n') \propto \frac{q(n,n')}{q(n)q(n')} = \frac{p(\mathbf{x}|y=+1,o=+1)}{p(\mathbf{x})} = r(\mathbf{x}) \tag{18}$$

which means that $s_\theta(n,n')$ is proportional to $r(\mathbf{x})$. Finally, combining the conclusion derived from Eq.11, we complete the proof of infoNCE's free-lunch as shown below:

$$\begin{aligned} &\forall \mathbf{x}=(n,n'), \hat{\mathbf{x}}=(\hat{n},\hat{n}') \in D: \\ &p(y=+1|\mathbf{x}) \le p(y=+1|\hat{\mathbf{x}}) \Leftrightarrow s_\theta(n,n') \le s_\theta(\hat{n},\hat{n}') \end{aligned} \tag{19}$$

which means that training with infoNCE not only optimizes the contrastive loss but also provides a "free-lunch" to model the positive class posterior probability $p(y=+1|\mathbf{x})$.

This free-lunch allows us to identify the D_U^+ samples like Eq.13 with a new threshold t_s corresponds to $s_\theta(\cdot,\cdot)$ as hyper-parameter:

$$y(\mathbf{x}) = h_s(\mathbf{x}=(n,n');\theta) := sign(s_\theta(n,n') - t_s) \tag{20}$$

By this point, we can extract semantically similar non-augmented samples D_U+ from D_U for resampling based on Eq.9:

$$\begin{aligned} D_U^+ &= \{(n,n')\}_{sign(s_\theta(n,n')-t_s)=+1}^{(n,n')\in D_U} \\ D^+ &= D_L^+ \cup D_U^+ \\ D^- &= D_U - D_U^+ \end{aligned} \tag{21}$$

which means that the semantically similar non-augmented samples D_U+ is treated as positive samples instead of negative ones.

3.2.2 ***Updating Training Objective.*** After resampling, we introduce D_U^+ into the GCL training process, ensuring the correction of sampling bias. We point out that the Eq.2 can be understood as the form of negative-log-likelihood probability as shown below:

$$J_{u_i,v_i} := -\log \frac{s_\theta(u_i,v_i)}{\sum_{j\neq i,j=1}^N s_\theta(u_i,u_j) + \sum_{j=1}^N s_\theta(u_i,v_j)} = -\log \mathbf{P}_{u_i,v_i} \tag{22}$$

where $\mathbf{P}_{u_i,v_i}$ is the normalized probability of numerator $s_\theta(u_i,v_i)$ over all possible v_j given u_i as an anchor node. Furthermore, we point out that $\mathbf{P}_{n,n'}$ can be regarded as the approximation of likelihood probability of a general node pair (n,n') being positive contrastive sample when n is the anchor sample. According to Eq.19, $s_\theta(n,n')$ satisfies the order-invariant relationship with the probability of contrastive sample (n,n') being positive. Note that when n is given as the anchor node, the denominator of the normalization for $s_\theta(n,n')$ to obtain $\mathbf{P}(n,n')$ is consistent. For example, given u_i as anchor node, for any v_j the denominator of $\mathbf{P}_{u_i,v_j}$ is always $\sum_{j\neq i,j=1}^N s_\theta(u_i,u_j) + \sum_{j=1}^N s_\theta(u_i,v_j)$. Therefore, based on Eq.19, the order of $\mathbf{P}_{n,n'}$ and $p(y=+1|(n,n'))$ is consistent when node n serves as the anchor:

$$\begin{aligned} &\forall n', n'' \in \mathcal{N}: \\ &\mathbf{P}_{n,n'} \le \mathbf{P}_{n,n''} \Leftrightarrow p(y=+1|(n,n')) \le p(y=+1|(n,n'')) \end{aligned} \tag{23}$$

where $\mathcal{N}$ is the set of nodes. Thus, the normalized $\mathbf{P}_{n,n'}$ can be seen as an approximation of $p(y=+1|(n,n'))$.

Under this understanding, L in Eq.3 can be naturally interpreted as the expectation of negative-log-likelihood probabilities of all labeled-positive samples:

$$\begin{aligned} L :=& \frac{1}{N}\sum_{i=1}^N \frac{J_{u_i,v_i} + J_{v_i,u_i}}{2} \\ =& \mathbb{E} J_{n,n'} \\ =& \mathbb{E} -\log \mathbf{P}_{n,n'} \end{aligned} \tag{24}$$

where $(n,n') \in D_L^+ = D^{aug+} = \{(u_i,v_i),(v_i,u_i)\}_{i=1}^N$. Therefore, minimizing the InfoNCE loss function is equivalent to maximizing the likelihood probability of all positive samples, i.e., D_L^+ in traditional GCL from the perspective of PU learning.

After resampling in Eq.21, the scope of positive samples is expanded to $D_L^+ \cup D_U^+$, therefore, the likelihood probability of D_U^+ must also be incorporated into the optimization objective:

$$L_{n,n'}^{corrected} := -\log(\mathbf{P}_{n,n'} \prod_{(n,n'')\in D_U+} (\mathbf{P}_{n,n''})^{\beta \hat{s}_\theta(n,n'')}) \tag{25}$$

As shown in the following Eq.27, considering that our method has different confidence for different D_U^+ samples, we assign exponential weights $\hat{s}_\theta(n,n'')$ to the negative-log-likelihood probabilities of each D_U^+ sample (n,n''). The weight is the globally normalized similarity score as follows:

$$\hat{s}_\theta(n,n') = \frac{s_\theta(n,n') - \min\{s_\theta(n_i,n'_j)\}_{i,j=1}^N}{\max\{s_\theta(n_i,n'_j)\}_{i,j=1}^N} \tag{26}$$

The more similar of (n,n'), the higher weight of $\mathbf{P}_{n,n'}$. Moreover, considering the differences between D_L^+ and D_U^+, we further assign an exponential weight β to the negative-log-likelihood probabilities of all D_U^+ samples comparing to the weight 1 for D_L^+ samples. After expanding the loss based on a certain D_L^+ sample, taking the expectation will yield the final loss:

$$\begin{aligned} L^{corrected} :=& \mathop{\mathbb{E}}_{(n,n')\in D_L^+} L_{n,n'}^{corrected} \\ =& \mathop{\mathbb{E}}_{\substack{(n,n')\in D_L^+ \\ (n,n'')\in D_U^+}} -log(\mathbf{P}_{n,n'} \prod (\mathbf{P}_{n,n''})^{\beta \hat{s}_\theta(n,n'')}) \end{aligned} \tag{27}$$

To mitigate the cumulative negative impact of individual erroneous D_U^+ instances, we employ a dynamic updating method for D_U^+. Specifically, we first train the model for M epochs using Eq.3. Then, every K training epochs, we use the representation similarity from the model obtained in the previous stage to resample the

D_U^+, and update the loss function for training the graph encoder. This process repeats until the best checkpoint of our graph encoder $\theta^*_{corrected}$. The pseudo-code of the whole pre-training process is shown in Algorithm 1.

Algorithm 1 The whole pre-training process of our semantic-guided sampling approach for GCL

Input: origin Graph $\mathbf{G}$, augmentation functions $aug_1(\cdot)$ and $aug_2(\cdot)$, randomly initialized graph encoder *theta*, warm-up epochs M, update interval epochs K, threshold t_s, max updating objective times T, learning rate α

Output: optimized graph encoder $\theta^*_{corrected}$

```
G^{aug_1} ← aug_1(G), G^{aug_2} ← aug_2(G)
D^+ ← D^{aug+}, D^- ← D^{aug-}                          ▷ Eq.(1)
θ^(0) ← θ
L^(0) ← L(θ^(0))                                        ▷ Eq.(3)
for i = 1 to M do
    θ^(0) ← θ^(0) − α · ∇_θ L^(0)                       ▷ Warming Up θ^(0)
end for
θ^(1) ← θ^(0)
for t = 1, ..., T do
    h^(t)(n, n'; θ^(t)) ← sign(s_{θ^(t)}(n, n') − t_s)  ▷ Free-Lunch
    D_U^{+,(t)} ← {(n, n')}_{h^(t)(n,n';θ^(t))=1}
    D^+ ← D^{aug+} ∪ D_U^{+,(t)}
    D^- ← D^{aug-} − D_U^{+,(t)}                        ▷ Resample
    L^(t) ← L^{corrected}(θ^(t))                        ▷ Eq.(27)
    for i = 1 to K do
        θ^(t) ← θ^(t) − α · ∇_θ L^(t)                   ▷ Update Training Objective
    end for
end for
Return θ*_{corrected} ← θ^(T)
```

3.3 Discussion

This section discusses the theoretical advantages of our method compared to other approaches such as [1, 8, 18, 38]. In summary, our advantages are twofold: First, the assumptions we use to identify D_U^+ through the free lunch of InfoNCE are more aligned with the actual conditions in GCL. Second, by updating the training loss based on the maximum likelihood objective of InfoNCE, we impose a stronger constraint on D_U^+, resulting in more thorough bias correction.

3.3.1 ***General Assumption for GCL.*** Generally, PU learning methods adopt the **S**elect **C**ompletely **A**t **R**andom (**SCAR**) assumption [12] which means the labeled positive samples are selected completely at random from the positive distribution independent from the sample's attributes. Under the SCAR assumption, both D_L^+ and D_U^+ are considered to be independently and identically distributed as the overall positive data $D^+ = D_L^+ \cup D_U^+$. Consequently, it is tempting to conclude that D_L^+ and D_U^+ share the same distribution. However, the SCAR assumption is too strong to hold because D_L^+ in GCL is constructed artificially via data augmentation while D_U^+ are the ones sharing similar position, features and structure information that exits inherently in the original graph. Unlike the SCAR assumption, the IOD assumption used in IFL-GCL is more relaxed and aligns more closely with the practical scenarios of GCL. For any contrastive sample $\mathbf{x} = (n, n')$, if node n and n' is semantically similar, then the probability of $\mathbf{x}$ being observed as a positive one is higher and vice versa.

3.3.2 ***Theoretically Solid Corrected Objective.*** Focus on the incorporating of newly discovered samples D_U^+, we find that some other works such as [38, 42] .etc just heuristically modify the InfoNCE loss function without understanding the original objective from the perspective of maximum likelihood. Specifically, to put the newly found D_U^+ in model's training, these works can be approximated as employing a linear combination of the likelihoods of D_L^+ and D_U^+ as shown below:

$$\begin{aligned} l^{corrected}_{u_i,v_i} &= -\log \frac{s_\theta(u_i,v_i) + \sum_{i,j=1}^{N} \alpha_{ij} s_\theta(u_i,v_j)}{\sum_{j\neq i,j=1}^{N} s_\theta(u_i,u_j) + \sum_{j=1}^{N} s_\theta(u_i,v_j)} \\ &= -log(\mathbf{P}_{u_i,v_i} + \sum_{i,j=1}^{N} \alpha_{ij} \mathbf{P}_{u_i,v_j}) \end{aligned} \tag{28}$$

where $\mathbf{P}_{u_i,v_i}$ and $\mathbf{P}_{u_i,v_j}$ corresponds to the likelihood of D_L^+ and D_U^+. Optimizing this loss is equivalent to maximizing a linearly combined likelihood of D_L^+ and D_U^+. It only requires a sufficiently large $\mathbf{P}u_i, v_i$ for D_L^+, imposing no strong constraints on the likelihood of D_U^+. In contrast, our proposed loss in Eq.27 adopts an exponential weighted product of the likelihoods of both D_U^+ and D_L^+. Our loss is more affected and sensitive to the likelihood of D_U^+, which results in a more rigorous constraint on D_U^+, leading to stronger bias correction.

4 Experiments

In this section, we perform thorough experiments to evaluate the effectiveness of our proposed method IFL-GCL. Under both paradigms of graph pre-training and LLMs as feature enhancement, we demonstrate the performance improvement of IFL-GCL over 7 traditional GCL baselines on 9 datasets for downstream node classification task. Additionally, we have conducted extensive empirical experiments, including: analyzing the semantics of D_U^+ found by IFL-GCL, and examining the impact of three important hyper-parameters.

4.1 Experimental Settings

Dataset Description. We evaluate our IFL-GCL on node classification task across 9 datasets under independent and identically distributed (IID) and out-of-distribution (OOD) scenarios. We employ five commonly used graph datasets Cora, PubMed, CiteSeer, WikiCS, Computers, and Photo [6, 23, 24, 31] for IID scenario where the train set and the test set exhibit similar distributions. We conduct experiments on three datasets from the GOOD benchmark [7]: GOODTwitch, GOODCora, and GOODCBAS, which are designed for OOD scenarios.

Baselines. We use 7 generally used GCL methods as baselines, among which MVGRL, GBT, BGRL, COSTA, DGI[3, 9, 34, 36, 46] are not directly related to our proposed method IFL-GCL, and GRACE and GCA [48, 49] are directly related to our proposed method IFL-GCL as they both utilize the InfoNCE loss. The directly-related baselines are used as warm-up. We denote our method based on

Table 1: Comparing our methods with 7 baselines across 9 datasets. We show the node classification accuracy, along with the corresponding standard deviation in parentheses. The best performance achieved on each dataset is highlighted in bold underlined, and the second-best performance is indicated as *bold italic*. We also show the improvement of our method compared to the directly related methods in the last two rows where Δ_{GR} represents the increase of IFL-GR compared to GRACE, and Δ_{GC} denotes the increase of IFL-GC compared to GCA.

	Cora	PubMed	CiteSeer	WikiCS	Computers	Photo	GOODTwitch	GOODCora	GOODCBAS
DGI	82.99$_{(1.38)}$	84.89$_{(0.05)}$	69.79$_{(0.40)}$	78.54$_{(0.85)}$	86.97$_{(0.07)}$	91.73$_{(0.11)}$	59.67$_{(2.26)}$	44.73$_{(0.79)}$	57.62$_{(1.78)}$
COSTA	84.61$_{(0.28)}$	86.01$_{(0.28)}$	71.31$_{(0.62)}$	78.93$_{(0.92)}$	88.56$_{(0.40)}$	92.30$_{(0.20)}$	62.12$_{(0.99)}$	49.31$_{(1.08)}$	41.90$_{(3.75)}$
BGRL	78.48$_{(0.54)}$	84.41$_{(0.08)}$	63.62$_{(1.28)}$	77.65$_{(0.67)}$	85.94$_{(0.51)}$	91.70$_{(0.55)}$	56.85$_{(4.31)}$	43.17$_{(0.93)}$	52.38$_{(4.42)}$
MVGRL	83.60$_{(0.65)}$	84.29$_{(0.16)}$	69.95$_{(0.34)}$	76.03$_{(0.52)}$	84.64$_{(0.38)}$	89.67$_{(0.14)}$	57.75$_{(2.21)}$	28.01$_{(0.60)}$	47.62$_{(1.78)}$
GBT	83.51$_{(0.62)}$	85.90$_{(0.03)}$	70.07$_{(1.15)}$	78.91$_{(0.52)}$	**89.34**$_{(0.09)}$	92.93$_{(0.10)}$	59.37$_{(2.31)}$	47.22$_{(0.40)}$	52.38$_{(4.42)}$
GRACE	83.97$_{(0.38)}$	86.34$_{(0.06)}$	70.75$_{(0.96)}$	78.90$_{(0.56)}$	88.51$_{(0.36)}$	92.33$_{(0.53)}$	64.29$_{(1.25)}$	50.30$_{(1.19)}$	52.38$_{(5.99)}$
GCA	84.79$_{(0.37)}$	***87.13***$_{(0.16)}$	70.57$_{(1.31)}$	77.99$_{(0.60)}$	88.63$_{(0.42)}$	***92.81***$_{(0.12)}$	61.17$_{(0.87)}$	51.66$_{(1.44)}$	50.95$_{(1.78)}$
IFL-GR	**85.40**$_{(0.42)}$	86.54$_{(0.09)}$	**71.83**$_{(0.52)}$	**79.36**$_{(0.51)}$	89.00$_{(0.08)}$	92.63$_{(0.21)}$	***66.23***$_{(0.71)}$	***51.84***$_{(0.87)}$	**61.43**$_{(4.04)}$
IFL-GC	***85.35***$_{(1.42)}$	**87.37**$_{(0.08)}$	***71.35***$_{(0.81)}$	***79.22***$_{(0.56)}$	***89.33***$_{(0.21)}$	**93.13**$_{(0.25)}$	**66.49**$_{(1.09)}$	**52.55**$_{(0.11)}$	***56.19***$_{(1.78)}$
Δ_{GR}	+1.43%	+0.20%	+1.08%	+0.46%	+0.49%	+0.30%	+1.94%	+1.54%	+9.05%
Δ_{GC}	+0.57%	+0.24%	+0.78%	+1.23%	+0.71%	+0.32%	+5.32%	+0.89%	+5.24%

GRACE as IFL-GR and the one based on GCA as IFL-GC. For all the baselines, we carefully tuned the hyper-parameters to achieve the best performances.

Evaluation and Metric. We focus on the node classification as the downstream task to show the quality of GCL's pre-trained representations. After pre-training via GCLs, we fine-tune a task-specific classifier with supervision. Specifically, we divide the dataset into train, valid, and test sets in a 1:1:8 ratio for supervised fine-tuning. Each time, we pre-train the graph model once and repeat the supervised-fine-tuning of linear node classifier 3 times, then we calculate the average and standard deviation of the node classification accuracy on test set.

4.2 IFL-GCL on the Framework of Graph Pre-training

As shown in table 1, we employ 7 commonly used graph contrastive learning methods as baselines, among which GRACE and GCA, which use InfoNCE as the loss function, were selected as directly relevant baselines. Additionally, the 6 datasets on the left side of the table are used for testing in IID scenarios, while the 3 datasets on the right side are used for testing in OOD scenarios. The main conclusions that can be drawn from this table are as follows.

(1) Our methods outperform the baselines. Our methods IFL-GR and IFL-GC occupy the best performances across all datasets apart from Computers. And our models rank second-best on all but PubMed and Photo. It is worth noting that our method is on par with the optimal performance on Computers, as well as the suboptimal performance on PubMed and Photo.

(2) Our methods consistently enhance the capability of directly related models. As indicated in the last two rows of Table 1, the improvement results on all datasets are positive, with the highest increase up to 9.05%. This demonstrates that after using IFL-GCL to correct sampling bias, the performance of GRACE and GCA is stably enhanced across all datasets.

(3) Our method shows more significant improvement under OOD scenarios. Under OOD scenarios, our methods occupy the top and second-best results across all datasets, with greater Δ_{GC} and Δ_{GR} than IID scenarios. This may be due to the reason that our method bridges the gap between different distributions by treating non-augmented samples under different distributions as positive ones, enabling the model to capture transferable knowledge across different distributions. As a result, IFL-GCL demonstrates good OOD generalization capability.

4.3 IFL-GCL on the Framework of LLM as Graph Enhancers

LLM plays a significant role in the study of graph foundation models where using LLMs as feature enhancers for Text-Attributed-Graphs (TAGs) is a common paradigm. In this section, we select 2 TAGs Cora and CiteSeer and use 4 pre-trained LLMs Llama3.2-1B, Llama3.2-3B, Qwen2.5-0.5B and Qwen2.5-1.5B to encode the their nodes' raw text features into dense representations. Then we use the LLM-encoded-representations as input features for baselines GRACE and GCA, and our methods IFL-GR and IFL-GC, thereby testing the effectiveness of our method under the current trends in graph foundation model research. Based on the analysis of Table 2, we can draw the following conclusions:

(1) Our method still outperforms directly-related baselines. The positive values in both the Δ_{GR} and Δ_{GC} rows indicate that, our method consistently improves the performance of the directly related GCL baselines on downstream task across all four LLMs enhancers and two TAG datasets. Since the node features encoded by LLM can capture semantic information better than traditional shallow text features, the semantic-guided sampling approach proposed in this paper can more accurately identify similar non-augmented samples, resulting in superior performance.

(2) Our method is sensitive to the choice of LLM and has the potential to conform to scaling laws. Under the same other

Table 2: Node classification results with 4 pre-trained LLMs as feature enhancers across 2 datasets. We show the accuracy, standard deviation, best and second best on each column, IFL's improvement compared to baselines the same way in Table 1.

	Llama3.2-1B		Llama3.2-3B		Qwen2.5-0.5B		QWen2.5-1.5B	
	Cora	CiteSeer	Cora	CiteSeer	Cora	CiteSeer	Cora	CiteSeer
GRACE	$84.34_{(0.61)}$	$75.48_{(0.95)}$	$84.64_{(0.64)}$	$76.75_{(0.30)}$	$84.07_{(1.08)}$	$75.83_{(0.27)}$	$84.09_{(0.49)}$	$75.52_{(0.51)}$
GCA	$84.71_{(0.39)}$	$\mathbf{76.10}_{(0.95)}$	$84.93_{(0.69)}$	$76.63_{(0.23)}$	$84.08_{(1.35)}$	$76.04_{(0.56)}$	$\mathbf{84.47}_{(0.35)}$	$76.31_{(0.76)}$
IFL-GR	$\mathbf{84.79}_{(0.68)}$	$76.02_{(0.74)}$	$\mathbf{85.03}_{(0.23)}$	$\mathbf{77.65}_{(0.40)}$	$\mathbf{84.12}_{(0.99)}$	$\mathbf{76.29}_{(0.51)}$	$84.34_{(0.58)}$	$\mathbf{76.42}_{(0.41)}$
IFL-GC	$\underline{85.38}_{(0.37)}$	$\underline{76.42}_{(0.82)}$	$\underline{85.43}_{(0.99)}$	$\underline{77.72}_{(0.46)}$	$\underline{85.38}_{(0.68)}$	$\underline{77.13}_{(0.34)}$	$\underline{85.37}_{(0.83)}$	$\underline{77.63}_{(0.19)}$
Δ_{GR}	+0.45%	+0.54%	+0.39%	+0.90%	+0.05%	+0.46%	+0.25%	+0.90%
Δ_{GC}	+0.67%	+0.32%	+0.5%	+1.09%	+1.30%	+1.09%	+0.90%	+1.32%

conditions, different LLMs yield varying results when used as enhancers. Specifically, Llama3.2-3B significantly outperforms Llama3.2-1B in all IFL-GR and IFL-GC results, with advantages of 1.3% and 1.6% on the CiteSeer dataset, respectively; Qwen2.5-1.5B is slightly better than Qwen2.5-0.5B. This phenomenon is due to our method's reliance on the modeling capability of representations for semantics; the larger the LLM, the stronger the enhanced features' ability to model semantics, which in turn facilitates our method's ability to find the correct D_U^+. Therefore, based on theoretical analysis and preliminary experimental results, our method has the potential to exhibit stronger bias correction capabilities as the scale of the LLM used as an enhancer increases.

Overall, our semantic-guided sampling approach for GCL holds great potential for the current research trend where LLMs are commonly involved in constructing graph foundation models.

4.4 Analysis of the Discovered Positive Samples and hyper-parameters

We also conduct extensive empirical analysis and evaluation on the proposed semantic-guided sampling approach, including: the analysis of discovered semantically similar non-augmented samples, and the analysis of hyper-parameters.

4.4.1 ***Analysis of the Discovered Positive Samples.*** This section will analyze how the semantic similarity of the D_U^+ samples identified by our method changes during training as it dynamically updates as shown in the following Fig.3. The x-axis of each subplot represents the number of completed training epochs including warming-up. And the interval on x-axis is set to the optimal update interval K for the corresponding setting. The y-axis shows the values of the two metrics measuring the average semantic similarity of node pairs in D_U^+ found in the corresponding epoch. Besides, we mark the epoch of optimal checkpoint with gray dashed line to research the relationship between model performance and D_U^+ semantic similarity during training process. To consider both IID and OOD scenarios, we selected the Cora and GOODTwitch as datasets. We statistically measure the semantic similarity of the D_U^+ samples found at each stage of the training process of IFL-GR and IFL-GC, respectively. We propose two metrics to reflect the similarity of elements in comparative samples: the *SameClassRatio* which is the proportion of samples with the same class label of

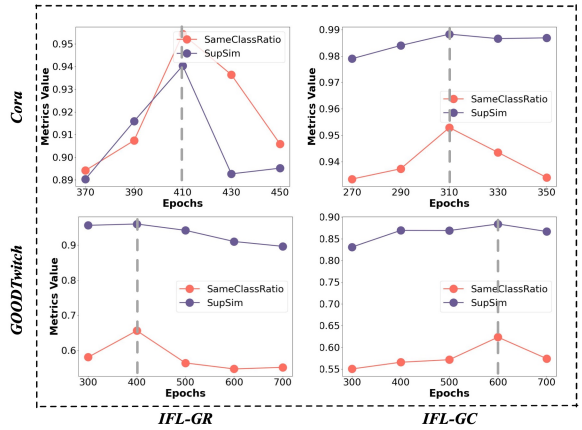


Figure 3: Analysis of semantic similarity of D_U^+ during training. The gray dashed line indicates the training epoch of optimal checkpoint.

D_U^+, and the *SupSim* which is the average similarity of supervised representations of D_U^+.

Specifically, the *SameClassRatio* refers to:

$$SameClassRatio = \frac{|\{(n,n')\}_{(n,n')\in D_U^+}^{Y_n=Y_{n'}}|}{|\{(n,n')\}_{(n,n')\in D_U^+}|} \tag{29}$$

where Y_n means the class label of node n in node classification task, $|\cdot|$ computes the number of contrastive samples in the set.

The supervised-representation-similarity *SupSim* use the graph encoder f_{θ^*} trained with supervision as shown in section 2 to calculate the average representation similarity of all node pairs in D_U^+ as shown in the following equation:

$$SupSim = \frac{1}{|\{(n,n')\}_{(n,n')\in D_U^+}|} \sum_{(n,n')\in D_U^+} \frac{\mathbf{H}_n \cdot \mathbf{H}_{n'}}{\|\mathbf{H}_n\|\|\mathbf{H}_{n'}\|} \tag{30}$$

where $\mathbf{H}$ is the nodes representation of $\mathbf{G}$ encoded by f_{θ^*}, $\mathbf{H}_n$ is the representation of node n, $\|\cdot\|$ is L_2 norm of the vector. These metrics reflect the semantic proximity of the mined D_U^+ samples from different perspectives; a higher metric indicates a higher probability of D_U^+ being positive samples.

Observing Fig.3, it can be seen that the trends of *SameClassRatio* and *SupSim* are largely consistent, both increase first and then decrease. And the peak value points of two metrics both correspond

to the optimal checkpoint epoch. This indicates that in the early stages, as the model trains to better and better state, the ability of GCL model gradually reaches optimality, leading to an overall increase in the semantic similarity of D_U^+. In the meanwhile, better D_U^+ samples correct the sampling bias in GCL. As model training progresses, overfitting and representation collapse issues may cause the decline in the model's capability, resulting in the decrease of the D_U^+ quality. Thus the semantic similarity of D_U^+ declines after the best epoch. This trend reflects a close relationship between the model's ability and the quality of D_U^+, with the two aspects complementing each other. This indicates that IFL-GCL is capable of accomplishing sampling bias correction based on a warmed-up GCL model and enhancing it's capabilities.

4.4.2 ***Analysis of Hyper-Parameters.*** In this part we study the effect of the following important hyper-parameters: warm-up epochs M, update-interval epochs K and the threshold t_s.

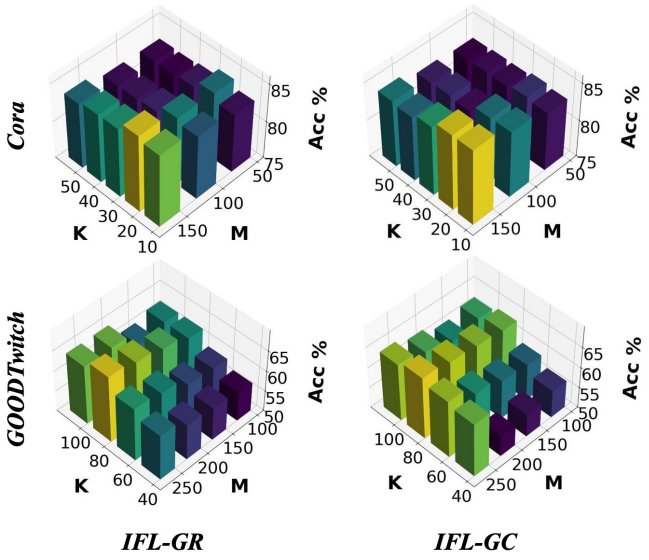


Figure 4: Analysis of Hyper-parameters: number of warm-up epochs M and number of update-interval epochs K.

- **Effect of warm-up and update-interval epochs.** As shown Fig.4, we conducted pre-training with different values of warm-up epochs M and update-interval epochs K on the Cora and GOODTwitch datasets using IFL-GR and IFL-GC, followed by the same supervised fine-tuning process on the same training set and report the node classification accuracy for each set. Observing this 3D bar chart, it can be seen that for both datasets, the performance generally improves as M increases within a certain range. We believe this is because within a certain range, a larger M allows the learned $s_\theta(\cdot,\cdot)$ to model $r(\mathbf{x})$ more adequately, thereby capturing semantic information more effectively and performing better bias correction. Additionally, it can be observed that the update interval K varies for different datasets. We think that it is related to the differences in the number of D_U^+ across different datasets. GOODTwitch have far more nodes in total than Cora, leading to more nodes in D_U^+ thus it needs larger update interval K to learn from these D_U^+ samples for correcting the sampling bias.
- **Effect of threshold.** To study the impact of the threshold t_s, we still conduct experiments on the Cora and GOODTwitch datasets for both IFL-GR and IFL-GC. With other hyper-parameters optimized, we set the threshold range to {0.8, 0.85, 0.90, 0.95, 0.99}, and the results are shown in Fig.5. The experimental results indicate that our method is sensitive to the threshold t_s, and different datasets have different optimal thresholds. For Cora the optimal threshold is 0.9 and for GOODTwitch is 0.95. Besides, we find out that a too low or too high threshold hurts the model's performance on both datasets. Here is our explanation: a threshold that is too low may lead to negative samples with low semantic similarity being incorrectly converted into positive samples which not only fails to correct the sampling bias but actually exacerbates it. This causes the representations learned by GCL can not accurately capture semantic information, leading to poor performance in downstream tasks. While a threshold that is too high may result in a too small number of D_U^+ samples, which play a minimal role in correcting thus the model cannot be optimized to its best state. Therefore, both too high and too low thresholds can affect the bias correction performance of our method.

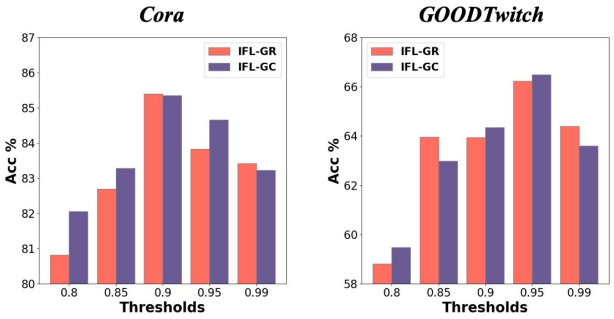


Figure 5: Analysis of Hyper-parameter: threshold t_s

5 Conclusion

This paper addresses the sampling bias issue in traditional graph contrastive learning by treating it as a Positive-Unlabeled learning problem where the definition of self-supervised tasks and contrastive samples should be semantically guided. We propose IFL-GCL, utilizing InfoNCE as a "free-lunch" to extract semantic information for resampling and redefine the maximum likelihood objective based on the corrected samples, resulting in a new InfoNCE loss function. Extensive experiments demonstrate the effectiveness and potential of our method in graph pretraining and graph foundation model research.

6 Acknowledgment

This work was supported by the Strategic Priority Research Program of the CAS under Grants No. XDB0680302 and the National Natural Science Foundation of China (Grant No.U21B2046, No.62202448).

References

[1] Anish Acharya, Sujay Sanghavi, Li Jing, Bhargav Bhushanam, Dhruv Choudhary, Michael Rabbat, and Inderjit Dhillon. 2022. Positive unlabeled contrastive learning. arXiv preprint arXiv:2206.01206 (2022).

[2] Peng Bao, Hua-Wei Shen, Wei Chen, and Xue-Qi Cheng. 2013. Cumulative effect in information diffusion: empirical study on a microblogging network. PloS one 8, 10 (2013), e76027.

[3] Piotr Bielak, Tomasz Kajdanowicz, and Nitesh V Chawla. 2022. Graph barlow twins: A self-supervised representation learning framework for graphs. Knowledge-Based Systems 256 (2022), 109631.

[4] Xingping Dong and Jianbing Shen. 2018. Triplet loss in siamese network for object tracking. In Proceedings of the European conference on computer vision (ECCV). 459–474.

[5] Yi Fang, Dongzhe Fan, Daochen Zha, and Qiaoyu Tan. 2024. Gaugllm: Improving graph contrastive learning for text-attributed graphs with large language models. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 747–758.

[6] C Lee Giles, Kurt D Bollacker, and Steve Lawrence. 1998. CiteSeer: An automatic citation indexing system. In Proceedings of the third ACM conference on Digital libraries. 89–98.

[7] Shurui Gui, Xiner Li, Limei Wang, and Shuiwang Ji. 2022. Good: A graph out-of-distribution benchmark. Advances in Neural Information Processing Systems 35 (2022), 2059–2073.

[8] Zayd Hammoudeh and Daniel Lowd. 2020. Learning from positive and unlabeled data with arbitrary positive shift. Advances in Neural Information Processing Systems 33 (2020), 13088–13099.

[9] Kaveh Hassani and Amir Hosein Khasahmadi. 2020. Contrastive multi-view representation learning on graphs. In International conference on machine learning. PMLR, 4116–4126.

[10] Shohei Hido, Yuta Tsuboi, Hisashi Kashima, Masashi Sugiyama, and Takafumi Kanamori. 2011. Statistical outlier detection using direct density ratio estimation. Knowledge and information systems 26 (2011), 309–336.

[11] Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, and Yizhou Sun. 2020. Gpt-gnn: Generative pre-training of graph neural networks. In Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. 1857–1867.

[12] Kristen Jaskie and Andreas Spanias. 2019. Positive and unlabeled learning algorithms and applications: A survey. In 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA). IEEE, 1–8.

[13] Wei Jin, Tyler Derr, Haochen Liu, Yiqi Wang, Suhang Wang, Zitao Liu, and Jiliang Tang. 2020. Self-supervised learning on graphs: Deep insights and new direction. arXiv preprint arXiv:2006.10141 (2020).

[14] Baoyu Jing, Chanyoung Park, and Hanghang Tong. 2021. Hdmi: High-order deep multiplex infomax. In Proceedings of the Web Conference 2021. 2414–2424.

[15] Wei Ju, Yifan Wang, Yifang Qin, Zhengyang Mao, Zhiping Xiao, Junyu Luo, Junwei Yang, Yiyang Gu, Dongjie Wang, Qingqing Long, et al. 2024. Towards Graph Contrastive Learning: A Survey and Beyond. arXiv preprint arXiv:2405.11868 (2024).

[16] Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. 2009. A least-squares approach to direct importance estimation. The Journal of Machine Learning Research 10 (2009), 1391–1445.

[17] Masahiro Kato, Takeshi Teshima, and Junya Honda. 2019. Learning from positive and unlabeled data with a selection bias. In International conference on learning representations.

[18] Zhiqiang Li, Jie Wang, and Jiye Liang. 2024. Debiased graph contrastive learning based on positive and unlabeled learning. International Journal of Machine Learning and Cybernetics 15, 6 (2024), 2527–2538.

[19] Gang Liu, Tong Zhao, Jiaxin Xu, Tengfei Luo, and Meng Jiang. 2022. Graph rationalization with environment-based augmentations. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 1069–1078.

[20] Mengyue Liu, Yun Lin, Jun Liu, Bohao Liu, Qinghua Zheng, and Jin Song Dong. 2023. B2-sampling: Fusing balanced and biased sampling for graph contrastive learning. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 1489–1500.

[21] Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and S Yu Philip. 2022. Graph self-supervised learning: A survey. IEEE transactions on knowledge and data engineering 35, 6 (2022), 5879–5900.

[22] Yujia Liu, Kang Zeng, Haiyang Wang, Xin Song, and Bin Zhou. 2021. Content matters: A GNN-based model combined with text semantics for social network cascade prediction. In Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, 728–740.

[23] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. 2000. Automating the construction of internet portals with machine learning. Information Retrieval 3 (2000), 127–163.

[24] Péter Mernyei and Cătălina Cangea. 2020. Wiki-cs: A wikipedia-based benchmark for graph neural networks. arXiv preprint arXiv:2007.02901 (2020).

[25] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018).

[26] Zhen Peng, Yixiang Dong, Minnan Luo, Xiao-Ming Wu, and Qinghua Zheng. 2020. Self-supervised graph representation learning via global context prediction. arXiv preprint arXiv:2003.01604 (2020).

[27] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. 2020. Gcc: Graph contrastive coding for graph neural network pre-training. In Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. 1150–1160.

[28] Saeed Rahmani, Asiye Baghbani, Nizar Bouguila, and Zachary Patterson. 2023. Graph neural networks for intelligent transportation systems: A survey. IEEE Transactions on Intelligent Transportation Systems 24, 8 (2023), 8846–8885.

[29] Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang. 2020. Self-supervised graph transformer on large-scale molecular data. Advances in neural information processing systems 33 (2020), 12559–12571.

[30] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2019. Dropedge: Towards deep graph convolutional networks on node classification. arXiv preprint arXiv:1907.10903 (2019).

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

[32] Hua-Wei Shen, Xue-Qi Cheng, and Jia-Feng Guo. 2011. Exploring the structural regularities in networks. Physical Review E—Statistical, Nonlinear, and Soft Matter Physics 84, 5 (2011), 056111.

[33] Kihyuk Sohn. 2016. Improved deep metric learning with multi-class n-pair loss objective. Advances in neural information processing systems 29 (2016).

[34] Shantanu Thakoor, Corentin Tallec, Mohammad Gheshlaghi Azar, Mehdi Azabou, Eva L Dyer, Remi Munos, Petar Veličković, and Michal Valko. 2021. Large-scale representation learning on graphs via bootstrapping. arXiv preprint arXiv:2102.06514 (2021).

[35] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. arXiv preprint arXiv:1710.10903 (2017).

[36] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2018. Deep graph infomax. arXiv preprint arXiv:1809.10341 (2018).

[37] Duo Wang, Yuan Zuo, Fengzhi Li, and Junjie Wu. 2024. LLMs as Zero-shot Graph Learners: Alignment of GNN Representations with LLM Token Embeddings. arXiv preprint arXiv:2408.14512 (2024).

[38] Lu Wang, Chao Du, Pu Zhao, Chuan Luo, Zhangchi Zhu, Bo Qiao, Wei Zhang, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, et al. 2024. Contrastive Learning with Negative Sampling Correction. arXiv preprint arXiv:2401.08690 (2024).

[39] Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In International conference on machine learning. PMLR, 9929–9939.

[40] Xiao Wang, Nian Liu, Hui Han, and Chuan Shi. 2021. Self-supervised heterogeneous graph neural network with co-contrastive learning. In Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining. 1726–1736.

[41] Yuyang Wang, Zijie Li, and Amir Barati Farimani. 2023. Graph neural networks for molecules. In Machine learning in molecular sciences. Springer, 21–66.

[42] Zixu Wang, Bingbing Xu, Yige Yuan, Huawei Shen, and Xueqi Cheng. 2024. Negative as Positive: Enhancing Out-of-distribution Generalization for Graph Contrastive Learning. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2548–2552.

[43] Yaochen Xie, Zhao Xu, Jingtun Zhang, Zhengyang Wang, and Shuiwang Ji. 2022. Self-supervised learning of graph neural networks: A unified review. IEEE transactions on pattern analysis and machine intelligence 45, 2 (2022), 2412–2429.

[44] Makoto Yamada, Taiji Suzuki, Takafumi Kanamori, Hirotaka Hachiya, and Masashi Sugiyama. 2013. Relative density-ratio estimation for robust distribution comparison. Neural computation 25, 5 (2013), 1324–1370.

[45] Yuning You, Tianlong Chen, Yang Shen, and Zhangyang Wang. 2021. Graph contrastive learning automated. In International Conference on Machine Learning. PMLR, 12121–12132.

[46] Yifei Zhang, Hao Zhu, Zixing Song, Piotr Koniusz, and Irwin King. 2022. COSTA: covariance-preserving feature augmentation for graph contrastive learning. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2524–2534.

[47] Han Zhao, Xu Yang, Zhenru Wang, Erkun Yang, and Cheng Deng. 2021. Graph Debiased Contrastive Learning with Joint Representation Clustering.. In IJCAI. 3434–3440.

[48] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2020. Deep graph contrastive representation learning. arXiv preprint arXiv:2006.04131 (2020).

[49] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph contrastive learning with adaptive augmentation. In Proceedings of the Web Conference 2021. 2069–2080.