electronics

MDPI

Article

Multiplex Graph Contrastive Learning with Soft Negatives

Zhenhao Zhao [1,†], Minhong Zhu [2,†], Chen Wang [1], Sijia Wang [1], Jiqiang Zhang [3], Li Chen [4,*] and Weiran Cai [1,*]

[1] School of Computer Science and Technology, Soochow University, Suzhou 215006, China
[2] Weill Graduate School of Medical Science, Cornell University, New York, NY 10065, USA
[3] School of Physics, Ningxia University, Yinchuan 750021, China
[4] School of Physics and Information Technology, Shaanxi Normal University, Xi'an 710119, China
* Correspondence: chenl@snnu.edu.cn (L.C.); wrcai@suda.edu.cn (W.C.)
† These authors contributed equally to this work.

Abstract: Graph Contrastive Learning (GCL) seeks to learn nodal or graph representations that contain maximal consistent information from graph-structured data. While node-level contrasting modes are dominating, some efforts have commenced to explore consistency across different scales. Yet, they tend to lose consistent information and be contaminated by disturbing features. We propose MUX-GCL, a novel cross-scale contrastive learning framework that addresses these key challenges in GCL by leveraging multiplex representations as effective patches to enhance information consistency. Our method introduces a soft-negative contrasting strategy based on positional affinities to reduce false negatives, thereby minimizing information loss during multi-scale contrasts. While this learning mode minimizes contaminating noises, a commensurate contrasting strategy using positional affinities further avoids information loss by correcting false negative pairs across scales. Extensive downstream experiments demonstrate that MUX-GCL yields multiple state-of-the-art results on public datasets. Our theoretical analysis further guarantees the new objective function as a stricter lower bound of mutual information of raw input features and output embeddings, which rationalizes this paradigm.

Keywords: graph contrastive learning; cross-scale contrast; information consistency; soft negatives

Received: 7 December 2024
Revised: 5 January 2025
Accepted: 15 January 2025
Published: 20 January 2025

Citation: Zhao, Z.; Zhu, M.; Wang, C.; Wang, S.; Zhang, J.; Chen, L.; Cai, W. Multiplex Graph Contrastive Learning with Soft Negatives. *Electronics* **2025**, *14*, 396. https://doi.org/10.3390/electronics14020396

1. Introduction

The rapid growth of graph-structured data across diverse domains such as social networks, biological systems, and recommendation engines has made graph representation learning (GRL) a critical area of research. Traditional supervised learning methods [1–5] have achieved considerable success in extracting meaningful patterns from graph data. However, these approaches rely heavily on labeled data, which are often scarce or expensive to obtain in real-world scenarios. This limitation has driven a surge of interest in self-supervised learning (SSL) techniques [6], particularly Graph Contrastive Learning (GCL), which seeks to leverage the inherent structure of graphs to learn useful representations without relying on labels.

In essence, GCL aims to learn nodal or graph representations by maximizing the information consistency between augmented views of the graph. Most of the established methods share the spirit of operating same-scale contrast between nodal representations through positive and negative pairs [7–9]. For graph-structured data, however, feature consistency can be well conveyed in structures of different scales [10]. Some efforts have thus expanded the scope to cross-scale modes, including the *patch-global* contrast of

nodal and graph representations [10–12], and *context-global* contrast between contextual subgraph and graph levels [13,14]. The contrasts of patches at diverse scales prove to be highly beneficial.

Yet, with the gain of richer information, cross-scale contrasting modes tend to suffer from contamination by inconsistent features [15]. The expansion to larger-scale patches tends to join out-of-class nodes and hence more feature inconsistency. The following is thus an intriguing question: *How to enable contrasts that capture more consistent features across scales while restrict contamination from inconsistency?*

This raises a request for a contrasting paradigm that exploits information maximally and selectively. One has to note that information loss is inherent in GCL. On one hand, an encoding process is not guaranteed to be information conservative. The inclination for oversmoothing is intrinsic to message-passing-based methods. On the other hand, pairing negatives between intra-class nodes leads to a loss of consistent features. This has been spotted in the same-scale contrast. Regarding this, some work excludes neighboring nodes to avoid false negatives [16,17] or weighs them as positives based on their saliency [18]. However, these approaches are not applicable to topological compositions in cross-scale scenarios.

We introduce MUX-GCL, a novel cross-scale contrastive learning paradigm that for the first time utilizes multiplex encoded information with the soft negatives of input graphs. The core of this paradigm lies in the contrasts of "effective patches" constructed from all layers of representations of the encoder. Higher-layer nodal embeddings, treated as representations of nodal patches, are contrasted with lower-layer embeddings, where features are less contaminated. To be commensurate with such patch contrasts, an efficient soft-negative contrasting strategy is proposed to minimize information loss from false negative pairs. In this manner, this GCL paradigm can maximally exploit consistent information from the entire encoder.

Our contributions are summarized as follows:

- We propose a novel cross-scale GCL paradigm, MUX-GCL, utilizing multiplex representations of the entire encoder, which maximally extracts consistent information while mitigates disturbing features.
- We introduce a patch contrasting strategy based on topological affinities to alleviate false negative pairs in cross-scale contrasts.
- Our theoretical justification guarantees the objective function of MUX-GCL as a stricter lower bound of mutual information between raw features and learned representations of augmented views, providing the rationale behind the method.
- Extensive experiments on both classification and clustering tasks demonstrate salient improvements, outperforming multiple state-of-the-art GCL models on public datasets.

2. Related Work

GCL methods have recently witnessed rapid development as an important branch of GRL. The core of GCL is to learn as much consistent information from the graph as possible. To achieve this goal, there are currently two main paradigms of GCL, namely, same-scale contrast (node-to-node/graph-to-graph) and cross-scale contrast (node-to-patch, patch-to-graph) [19].

2.1. Same-Scale GCL

The most common method for obtaining consistent information in GCL is same-scale contrast. They work by bringing representations of positive pairs from different views closer together and pushing negative pairs (if any) farther apart. GRACE [7], GCA [9], and

ProGCL [16] leverage the InfoNCE loss for nodal representation learning by considering the same node under different views as positives and other nodes as negatives. BGRL [20], G-BT [21], CCA-SSG [22], and HomoGCL [18] draw inspiration from BYOL [23], where only positive samples are considered. There are also works such as GraphCL [8] and JOAO [24] leveraging same-scale contrast across multiple views to learn graph representations. Despite making progress in many scenarios, same-scale contrast overlooks consistent information that exists across different scales and cannot mitigate the loss of consistent information caused by message passing.

2.2. Cross-Scale GCL

Unlike same-scale GCL, which is limited to obtaining consistent information at a single scale, another group of works manage to obtain consistent information from different scales on the graph. DGI [10] contrasts patch representations with the graph representation generated from a readout function to capture global consistent information. InfoGraph [25] further improves this idea by replacing graph representations with that of other larger-scale substructures. More recently, MVGRL [11] extends patch-to-graph contrast to multi-scale contrast by applying diffuse augmentation to one view. However, while it is possible for existing methods to obtain consistent information at different scales, according to the homophily assumption, representations of larger-scale substructures also introduce more inconsistent information, leading to contamination. This suggests that we should redesign a cross-scale contrast paradigm that can avoid such contamination.

2.3. Contrasting Strategies with Negative Mining

The proper mining and identification of false negatives is another important strategy for GCL methods to reduce the loss of consistent information. ProGCL [16] seeks to measure negative samples by fitting a Beta Mixture Model to estimate the probability of being true negatives. AUGCL [17] further utilizes an uncertainty-based modeling of collective affinities to learn a more precise measure. Different from its predecessors, HomoGCL directly treats neighbor nodes as positives and leverages the clustering-based method to evaluate the confidence. Beyond the node level, CuCo [26] attempts to select proper negatives samples on graph learning. Although these methods have achieved some success, they do not directly utilize the topological characteristics of the graph. However, they are all limited to same-scale contrast scenarios. A commensurate way of negative mining is in requests in the context of cross-scale contrastive learning.

3. Methods

3.1. Preliminaries and Notations

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a graph, where $\mathcal{V} = \{v_i\}_{i=1}^{N}$, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ are the node and edge sets, respectively. We let $X \in \mathcal{R}^{N \times F}$ and $A \in \{0,1\}^{N \times N}$ be the feature matrix and adjacency matrix. As a form of SSL, the purpose of our model is to learn a reliable representation $f(X, A) \in \mathcal{R}^{N \times F}$ of the input data with no labels through a GCN encoder. It is essential to support downstream tasks, such as node classification and clustering. Hence, the learned representations will be commonly input to a minimal prediction head for tests.

For a standard GCL paradigm, an augmentation method is applied to transform the original input graph $\mathcal{G}$ into two different views $\mathcal{G}_U(X_U, A_U)$ and $\mathcal{G}_V(X_V, A_V)$. By training the GCN encoders, the final nodal representations $U = f(\mathcal{G}_U)$ and $V = f(\mathcal{G}_V)$ are to maximize an objective function that contrasts them corresponding to the two views. A quintessential objective function as an instantiation of InfoNCE proposed in GRACE [7] is widely adopted as a reference, which is defined as the sum of pairwise functions as

$$\mathcal{L}_g(u_i, v_i) = \log \frac{e^{\theta(u_i, v_i)/\tau}}{e^{\theta(u_i, v_i)/\tau} + \sum_{j \neq i} e^{\theta(u_i, v_j)/\tau} + \sum_{j \neq i} e^{\theta(u_i, u_j)/\tau}} \quad (1)$$

which encompasses the representations of the same anchor node as a positive pair and those of all possible combinations of different nodes as negative pairs, where the metric $\theta(u, v)$ is a predefined similarity function (we use the dot product here). Note that negative pairs can be constructed from the same or different views.

3.2. Motivation

We seek to establish a cross-scale contrastive learning method that gains richer consistent information. Two aspects are of our concern: the construction of multi-scale patches and the contrasting strategy, which address cross-scale constrasts and negative mining, respectively.

The key issue with conventional ways of constructing patch representations through a readout function (such as an extra pooling layer) is the information loss caused by involving inconsistent features. Instead, we consider using the entire ensemble of latent and final representations of an encoder for building patches. From the perspective of message passing, we form an "effective patch" by regarding a k-th layer embedding of an anchor node as a representation of a k-hop ego-net centering on it. While a conventional patch refers to a set of original nodes, an effective patch is a representation of such a nodal set. This way, it treats the encoder as a multiplex network, which introduces no extra information contamination.

Cross-scale contrasts may thus be established between pairs of such patch representations. The roles of effective patches in contrasting can be justified by observing the similarity between cross-layer embeddings as demonstrated for GRACE. Here, a pair of positive effective patches are around the same central node in different views, whereas a pair of negative patches are around different central nodes in either different views or the same view. As shown in Figure 1, all positive patch pairs, regardless of layers, are far more similar than negative pairs, as suggested by the well-separated distribution of similarities. This strongly indicates that all representations across layers deserve to be involved in graph contrastive learning. This insight led to the proposal of Multiplex Patch Contrast.

Yet, to systematically pairing cross-scale patches, we need a contrasting strategy that maximally preserves consistent features. This aims essentially to avoid the brutal erasure of exploitable information by pairing false negatives, which are more likely to occur due to patch overlaps. In the absence of class labels, we evaluate the likelihood of false negatives on the topological affinities of patches as priors. The use of affinities thus builds up "soft negatives" in contrast to the commonly used hard ones in GCL, through the proposed patch affinity estimation module. While hard negative pairs are expected to be dissociated with each other, soft negative pairs still have some similarities and their representations should not be fully distanced.

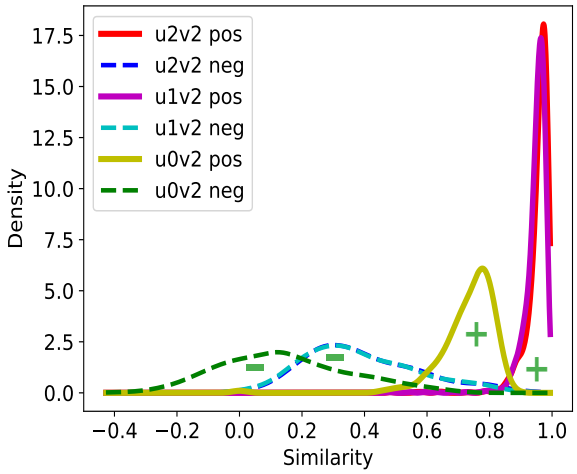


Figure 1. Similarity distributions of cross-layer embeddings between two augmented views (for GRACE). u and v denote two augmented views derived from the original graph. The vertical axis represents the node similarity calculated using the dot product. The vertical axis represents the probability density function values estimated using a Gaussian kernel. All positive pairs are substantially more similar than negative pairs, labeled as $u_m v_n$ pos/neg with m and n numbering the layers.

3.3. Framework

From the rationale above, we establish "effective patches" using all representations of the encoder for contrastive learning. Each nodal embedding $U^{(k)}$ ($V^{(k)}$ in the other view) on the k-th layer of the GCN now serves as an effective representation of a k-hop ego-net centered at the anchor node. Specifically, the definition of the patch representation takes the standard form $U^{(k)} = \sigma(\tilde{A}U^{(k-1)}W^{(k)}) \in \mathcal{R}^{N\times F}$ with the initial input $U^{(0)} = X$, where $\tilde{A}$ is the normalized adjacency matrix, and $W^{(k)}$ a set of trainable parameters.

With this premise, we now introduce the cross-scale contrastive learning paradigm MUX-GCL (Figure 2). We deploy two modules, i.e., **Multiplex Patch Contrast** and **patch affinity estimation**.

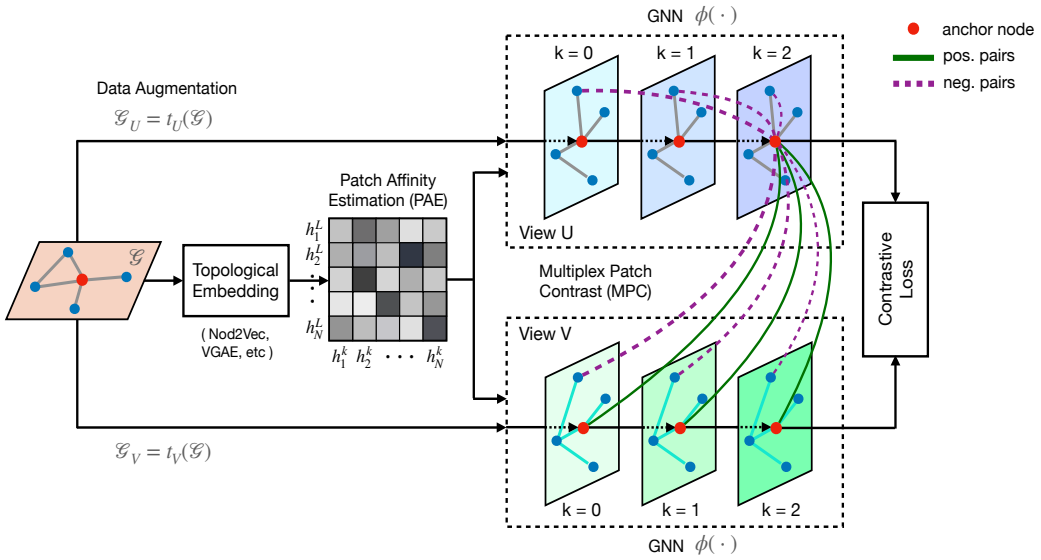


Figure 2. Overall architecture of MUX-GCL. Contrasts are executed between "effective patches" constructed from all representations of the multiplex encoder as illustrated by the links. The pairwise affinities of topological embedding estimate the likelihood of being false negatives. Augmentations are implemented as in GRACE. Positive and negative pairs are labeled in the figure.

Multiplex Patch Contrast (MPC). To contrast effective patches across scales, we extend the commonly used InfoNCE loss from same-scale contrast to a multiplex setting. Since final representations of the encoder are ultimately desired, we conduct cross-scale contrasts between final and all intermediate layers representations. The multiplex objective function is given as follows:

$$\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) = \log \frac{e^{\theta(u_i^{(L)}, v_i^{(k)})/\tau}}{e^{\theta(u_i^{(L)}, v_i^{(k)})/\tau} + \sum_{j \neq i} \omega_{ij}^{Lk} e^{\theta(u_i^{(L)}, v_j^{(k)})/\tau} + \sum_{j \neq i} \omega_{ij}^{Lk} e^{\theta(u_i^{(L)}, u_j^{(k)})/\tau}} \tag{2}$$

where $\theta(\cdot, \cdot)$ is the similarity function. The metric ω_{ij}^{Lk} represents a measure of the likelihood of being false negatives. The embeddings from intermediate layers are transformed using feed-forward layers to ensure dimensional compatibility. To treat the contrasts in a balanced way, we average the objective function across different scales as expressed by the pairwise objective function

$$\mathcal{L}_c(u_i, v_i) = \sum_{k=0}^{L} \lambda_k \mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) \tag{3}$$

where λ_k is the weight for contrasting the final L-th layer and the intermediate k-th layer, with $\sum_{k=0}^{L} \lambda_k = 1$.

Finally, to ensure symmetry between the two views, the overall objective function is defined as

$$\mathcal{L}_{MUX} = \frac{1}{2N} \sum_{i=1}^{N} [\mathcal{L}_c(u_i, v_i) + \mathcal{L}_c(v_i, u_i)]. \tag{4}$$

Patch Affinity Estimation (PAE). The affinity estimation function assigns weights to negative pairs to alleviate the problem of false negatives. Notably, in the cross-scale contrast, patches are more likely to share information due to their positional affinity, where overlaps are significantly more incident. A higher affinity score thus indicates a higher likelihood of being false negatives. This weighting scheme is thus to reduce the loss of consistent information in negative pairs.

For this scenario, we propose an affinity estimation strategy using topological positions as a decent prior. Concretely, we employ a graph embedding algorithm to obtain nodal representations that contain solely topological information. The topological representation of a patch is then simply obtained by pooling the encompassed nodes

$$H^{(0)} = T(A, X) \qquad h_i^{(k)} = Pool_{j \in G_i^{(k)}}(h_j^{(0)}) \tag{5}$$

where $T(\cdot)$ represents a learning algorithm that maps nodes to a topological embedding space. $G_i^{(k)}$ represents the k-hop ego-network centered on node i. *Pool* denotes the pooling function aggregating nodal embeddings within the patch.

Here, we consider two learning algorithms to obtain the topological embeddings: Node2Vec [27] and VGAE (Variational Graph Auto-Encoder) [28]. We remark that the decoder of VGAE is to recover the adjacency matrix of the input graph and hence learns topological features only.

To obtain the inter-patch affinities, we compute the similarities of these topological representations. Based on the affinity score for a negative instance pair, we compute the weight ω as the estimated likelihood of being false negatives

$$\omega_{ij}^{Lk} = 1 - \eta(h_i^{(L)}, h_j^{(k)}) \tag{6}$$

where $k \in \{0, 1, \ldots, L\}$; $\eta(\cdot, \cdot)$ is the affinity function that measures the positional similarity. Here, we take the form of normalized inner product $\eta(h_i^{(L)}, h_j^{(k)}) = \langle h_i^{(L)}, h_j^{(k)} \rangle$.

3.4. Theoretical Justification

The InfoMax principle claims that a mapping function for contrastive learning should be learned to maximize the mutual information between the input node features and learned node representations. Based on this, we provide a theoretical justification for our multiplex contrastive objective, demonstrating its rationale through the lens of maximizing mutual information.

Proposition 1. *The multiplex contrastive objective in Equation (2) is a lower bound of mutual information (MI) between raw input features* $\mathbf{X}$ *and output node embeddings* $\mathbf{U}$ *and* $\mathbf{V}$ *in two augmented views. Moreover, with a high statistical significance, the objective is also a stricter lower bound compared with the contrastive objective* $\mathcal{L}_{GR}$ *in Equation (1) proposed by GRACE. Formally,*

$$\mathcal{L}_{GR} < \mathcal{L}_{MUX} < I(\mathbf{X}; \mathbf{U}, \mathbf{V}) \tag{7}$$

Proof. See Appendix A.2. □

We can hence conclude that maximizing $\mathcal{L}_{MUX}$ is equivalent to maximizing a lower bound of the mutual information between raw features and learned node representations, which is yet stricter than the commonly used contrastive objective. This guarantees model convergence and provides a theoretical base for the performance boost [18].

3.5. Time Complexity Analysis

The time cost of the multiplex contrast mechanism is limited compared to the prevailing GCL methods. Concretely, we choose GRACE for comparison. Given a graph with N nodes and E edges, and assuming a GCN encoder with L layers and d hidden dimensions, the time complexity of encoding and loss function of GRACE are $O(L(Nd^2 + Ed))$ and $O(N^2 d)$, respectively. For the encoding stage, MUX-GCL takes extra $O(LNd^2)$ to acquire intermediate embeddings through linear layers, which does not increase the time complexity significantly, as L is typically very small ($L = 2$ for most cases). For the loss function, the time complexity of MUX-GCL is $O((L + 1)N^2 d)$, which is on the same order of magnitude as that of GRACE, noting that the InfoNCE loss in GRACE is a special case of Equation (3) when $\lambda_L = 1$. Furthermore, the time complexities of Node2Vec and VGAE used in the PAE module are $O(N)$ and $O(Nd^2 + Ed)$. This does not add to the overall complexity since PAE can be implemented as pre-processing and computed only once in the training phase.

4. Experiments

We conduct experiments on various tasks and datasets to demonstrate the effectiveness of MUX-GCL. First, we outline the experimental setup. Then, we compare the performance of MUX-GCL with state-of-the-art GCL methods on widely used benchmarks. We also explore the effects of the proposed blocks and hyper-parameters through ablation experiments and sensitivity analysis. Finally, we provide a runtime analysis of training MUX-GCL.

4.1. Experimental Setup

Datasets. We evaluate our method on five real-world benchmark datasets that have been widely used for previous GCL methods: three citation networks, Cora, Citeseer, and PubMed [29], and two co-purchase networks, Amazon-Photo and Amazon-Computers [30]. All datasets are randomly divided into 10%, 10%, and 80% proportions for training, validation, and testing. We do not use the public split of Cora, Citeseer, and PubMed, as they contain only

a partial portion of the whole dataset. However, for completeness, we also provide results on the public split in the appendix. Table 1 presents informative statistics for these datasets. Further detailed descriptions are provided below:

- Cora, Citeseer, and Pubmed [29] are three citation network datasets. Their node labels are related research area of publications. Each publication in Cora or Citeseer is described by a 0/1-valued word vector, indicating the absence/presence of the corresponding word from the dictionary. In Pubmed, each publication is described by a TF/IDF weighted word vector from the dictionary.
- Amazon-Photo, Amazon-Computers [30] are based on Amazon's co-purchase data. Nodes represent products, while edges show how frequently they are purchased together. Each product is described using a Bag-of-Words representation based on the reviews (node features).

Table 1. Statistics of datasets used in our experiments.

Dataset	#Nodes	#Edges	#Features	#Classes
Photo	7650	238,163	745	8
Computers	13,752	491,722	767	10
Cora	2708	10,556	1433	7
Citeseer	3327	9228	3703	6
Pubmed	19,717	44,338	500	3

Baselines. We compare MUX-GCL with multiple baselines, including traditional graph self-supervised learning methods such as Node2Vec [27] and DeepWalk [31], autoencoder-based models like GAE and VGAE [28], DMoN [32], a graph clustering method build upon GNN, and contrastive-based graph self-supervised learning methods like DGI [10], GRACE [7], MVGRL [11], GCA [9], SUGRL [33], BGRL [20], ProGCL [16], G-BT [21], COSTA [34], SFA [35], HomoGCL [18], and MA-GCL [36]. For all baselines, we reproduce the experiments using the code provided by the original papers, and all results are obtained from the hyper-parameters specified in the original papers. For models that are reproducible, we use the results reported in the existing literature.

Evaluation protocol . To adhere to the evaluation framework utilized by prior work [7,9,10], we initially train each model in an unsupervised manner using the entire graph along with node features. Subsequently, we feed the raw features into a standard trained two-layer GCN encoder, yielding embeddings for utilization in downstream tasks. For the node classification task, we employ an ℓ_2-regularized logistic regression classifier from the Scikit-Learn library [37], utilizing the embeddings acquired in the preceding step. For the node clustering task, we employ KMeans as clustering method and measure the clustering performance in terms of two prevalent metrics: Normalized Mutual Information (NMI) score and Adjusted Rand Index (ARI). NMI $= 2I(Y;C)/[H(Y) + H(C)]$, with Y and C being the class labels and predicted cluster indexes, respectively, $I(\cdot)$ being the mutual information, and $H(\cdot)$ being the entropy. ARI $= (RI - E[RI])/(max(RI) - E(RI))$, with RI being the Rand Index [18,38].

4.2. Node Classification

We first validate the effectiveness of MUX-GCL via node classification tasks on five public datasets. As summarized in Table 2, MUX-GCL significantly outperforms all the baseline models across five public datasets. The superiority of MUX-GCL can be attributed to its special efforts in mitigating the loss of consistency information. By contrasting output embeddings with patch representations obtained from intermediate layers, the multiplex contrast paradigm helps mitigate the accumulation of inconsistent information from the expanding computation subgraph. This allows MUX-GCL to surpass advanced same-scale

GCL methods (e.g., ProGCL and HomoGCL), where the contrasting process occurs only between output embeddings. Meanwhile, MUX-GCL assigns lower weights to potential intra-class nodes in the contrastive loss to avoid mistakenly treating them as negatives, thereby reducing the potential loss of consistency information. This further ensures that MUX-GCL can surpass other same-scale GCL methods (e.g., GRACE, GCA, BGRL, COSTA) that lack the capability to discern false negatives. Further more, different from those previous cross-scale GCL methods (e.g., DGI, MVGRL) where output embeddings are contrasted with embeddings pooled from a larger scale, MUX-GCL choose to contrast with intermediate embeddings summarizing information of a smaller subgraph that contains less inconsistency information. Such avoidance of potential inconsistency information contamination helps MUX-GCL perform better than those cross-scale GCL methods.

Table 2. Node classification results (Acc (%) ± std for 5 seeds) of running on five commonly used datasets. X and A denote the feature matrix and adjacency matrix, respectively. The highest performance is highlighted in boldface.

Model	Training Data	Cora	Citeseer	Pubmed	Photo	Computers
raw feat.	X	64.8 ± 0.1	64.6 ± 0.1	84.8 ± 0.0	78.5 ± 0.0	73.8 ± 0.0
node2vec	A	74.8 ± 0.0	52.3 ± 0.1	80.3 ± 0.1	89.7 ± 0.1	84.4 ± 0.1
DeepWalk	A	75.7 ± 0.1	50.5 ± 0.1	80.5 ± 0.2	89.4 ± 0.1	85.7 ± 0.1
GAE	X, A	76.9 ± 0.0	60.6 ± 0.2	82.9 ± 0.1	91.6 ± 0.1	85.3 ± 0.2
VGAE	X, A	78.9 ± 0.1	61.2 ± 0.0	83.0 ± 0.1	92.2 ± 0.1	86.4 ± 0.2
DGI	X, A	82.6 ± 0.4	68.8 ± 0.7	86.0 ± 0.1	91.6 ± 0.2	84.0 ± 0.5
GRACE	X, A	83.3 ± 0.4	72.1 ± 0.5	86.3 ± 0.1	92.5 ± 0.2	87.8 ± 0.2
MVGRL	X, A	83.8 ± 0.3	73.1 ± 0.5	86.3 ± 0.2	91.7 ± 0.1	87.5 ± 0.1
GCA	X, A	82.8 ± 0.3	71.5 ± 0.3	86.0 ± 0.2	92.2 ± 0.2	87.5 ± 0.5
SUGRL	X, A	83.4 ± 0.5	73.0 ± 0.4	84.9 ± 0.3	93.2 ± 0.4	88.8 ± 0.2
BGRL	X, A	83.7 ± 0.5	73.0 ± 0.1	84.6 ± 0.3	91.5 ± 0.4	87.3 ± 0.4
G-BT	X, A	83.6 ± 0.4	72.9 ± 0.1	84.5 ± 0.1	92.6 ± 0.5	86.8 ± 0.3
ProGCL	X, A	84.2 ± 0.5	72.2 ± 0.2	86.4 ± 0.2	93.2 ± 0.1	88.7 ± 0.1
COSTA	X, A	84.3 ± 0.2	72.9 ± 0.3	86.0 ± 0.2	92.6 ± 0.5	88.3 ± 0.1
SFA	X, A	84.1 ± 0.1	73.7 ± 0.2	85.6 ± 0.1	92.8 ± 0.1	88.1 ± 0.1
HomoGCL	X, A	84.9 ± 0.2	71.7 ± 0.3	85.8 ± 0.1	93.0 ± 0.2	89.0 ± 0.1
MA-GCL	X, A	83.9 ± 0.1	72.1 ± 0.4	85.6 ± 0.4	93.4 ± 0.1	89.0 ± 0.1
MUX-GCL	X, A	**85.5 ± 0.3**	**73.8 ± 0.2**	**86.9 ± 0.2**	**93.9 ± 0.1**	**90.7 ± 0.1**

4.3. Node Clustering

Performance on node classification tasks indicates that our MUX-GCL can obtain node embeddings better than those previous GCL methods. However, further experiments are needed to demonstrate that the embeddings learned by MUX-GCL contain more consistent information, thereby achieving higher quality. Here, we choose to perform node clustering on the Photo and Computers datasets. Specifically, we opt for NMI and ARI as metrics to assess the concordance between our clustering results and the true data distribution. Higher concordance between the clustering results and true labels means that there is more consistency information among the intra-class nodes, making them easier to be recognized as belonging to the same class.

As shown in Table 3, MUX-GCL mostly outperforms other methods by a large margin on both metrics for the two datasets. We credit the performance enhancement to two aspects. For one thing, the PAE module assigns higher affinity scores to intra-class nodes, thus pushing inter-class nodes away from intra-class ones. For another, the MPC module helps make the clusters more compact by filtering out the inconsistent information among intra-class nodes. As a result, the boundaries between different clusters within the embedding

space become more defined and clear, indicating that more consistent information is preserved within the clusters.

Table 3. Node clustering results in terms of NMI and ARI on Photo and Computer datasets. $\triangle_x = 0.01x$ is used to denote the standard deviation on 5 seeds. "-" means missing values from the literature. The highest performance is highlighted in boldface.

Model	**Photo**		**Computers**	
Metric	**NMI**	**ARI**	**NMI**	**ARI**
GAE	$0.616 \pm \triangle_1$	$0.494 \pm \triangle_1$	$0.441 \pm \triangle_0$	$0.258 \pm \triangle_0$
VGAE	$0.530 \pm \triangle_4$	$0.373 \pm \triangle_4$	$0.423 \pm \triangle_0$	$0.238 \pm \triangle_0$
DGI	$0.376 \pm \triangle_3$	$0.264 \pm \triangle_3$	$0.318 \pm \triangle_2$	$0.165 \pm \triangle_2$
MVGRL	$0.344 \pm \triangle_4$	$0.239 \pm \triangle_4$	$0.244 \pm \triangle_0$	$0.141 \pm \triangle_0$
BGRL	$0.668 \pm \triangle_3$	$0.547 \pm \triangle_4$	$0.484 \pm \triangle_0$	$0.295 \pm \triangle_0$
GCA	$0.614 \pm \triangle_0$	$0.494 \pm \triangle_0$	$0.426 \pm \triangle_0$	$0.246 \pm \triangle_0$
DMoN	$0.633 \pm \triangle_0$	-	$0.493 \pm \triangle_0$	-
HomoGCL	$0.671 \pm \triangle_2$	$0.587 \pm \triangle_2$	$0.534 \pm \triangle_0$	**0.396 ±** $\triangle_0$
MUX-GCL	**0.712** $\pm \triangle_1$	**0.609** $\pm \triangle_1$	**0.552** $\pm \triangle_0$	$0.388 \pm \triangle_1$

4.4. Ablation Study

In this section, we verify the effectiveness of the proposed multiplex contrast mechanism and patch affinity estimation. We conducted ablation experiments to test the performance of the following model variants on three datasets:

(1) **PAE**: only conducting same-scale contrast between the output embeddings, without engaging in cross-scale contrast.
(2) **MPC**: performing a complete cross-scale contrast but refraining from utilizing patch affinity estimation to identify false negatives.
(3) **MUX-GCL (PAE + MPC)**: the full version of our model.

As illustrated in Table 4, compared with the the latest baselines, both PAE and MPC contribute to performance enhancement, with the optimal outcome attained when the two are integrated. This demonstrates that contrasting patch representations at different scales and effectively identifying false negatives both play crucial roles in preserving consistency information.

Table 4. Ablation study (Accuracy (%) ± std for 5 seeds) on two multiplex blocks. The highest performance is highlighted in boldface.

Model\Dataset	**Cora**	**Citeseer**	**Photo**
PAE	85.08 ± 0.26	73.29 ± 0.20	93.34 ± 0.09
MPC	84.78 ± 0.36	73.44 ± 0.20	93.78 ± 0.07
PAE+MPC	**85.43 ± 0.21**	**73.77 ± 0.17**	**93.89 ± 0.10**

4.5. Variants of PAE-Based Models

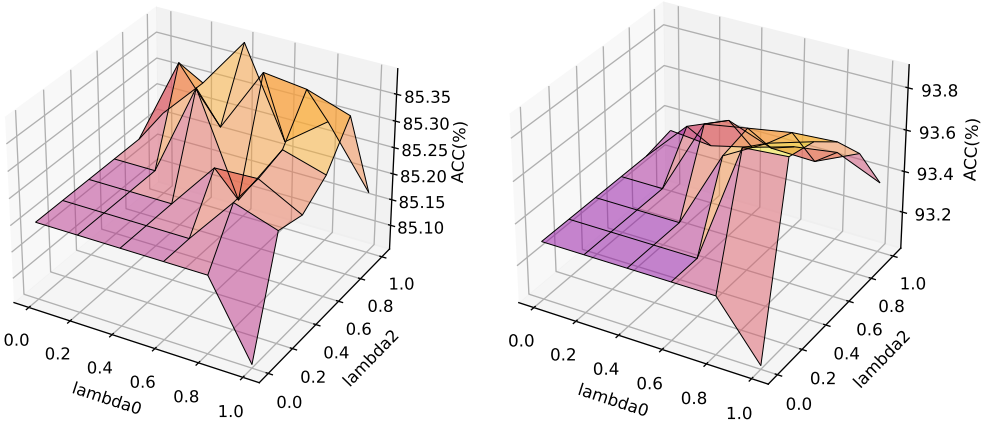
We additionally assess the influence of selecting base models for patch affinity estimation. We choose from Node2Vec and VGAE, as they all prioritize preserving graph topology in the learned embeddings. As is shown in Table 5, regardless of which model is used as the base model for patch affinity estimation, the results obtained surpass those of existing state-of-the-art models. When comparing the two variants, mixed results emerge across different datasets. While VGAE excels on Cora and Photo, it falls short compared to Node2Vec on Pubmed. This indicates that choosing different base models for patch affinity estimation will be more targeted for different datasets. It also reveals that a better patch affinity estimation base model can improve the performance of MUX-GCL.

Table 5. Variants of PAE-based models. Node classification results (Acc (%) ± std for 5 seeds) of running on Cora, Pubmed and Photo. The highest performance is highlighted in boldface.

PAE Method	Cora	Pubmed	Photo
Node2Vec	85.33 ± 0.37	**86.94 ± 0.24**	93.73 ± 0.04
VGAE	**85.43 ± 0.21**	86.63 ± 0.15	**93.89 ± 0.10**

4.6. Hyper-Parameter Sensitivity Analysis

Here we study the effect of hyper-parameter λ in Equation (3). The optimal value of λ reflects the impact of different intermediate layer representations on the gain of consistency information. For the typical two-layer GCN encoder we use in most datasets, based on Equation (3), there are three hyper-parameters $\lambda_0, \lambda_1, \lambda_2$ that determine the importance of each component. We perform grid search over two hyper-parameters λ_0, λ_2 and set $\lambda_1 = 1 - \lambda_2 - \lambda_0$. The result is shown in Figure 3. When we set the hyper-parameters to $\lambda_0 = \lambda_1 = 0$ and $\lambda_2 = 1$ (same-level contrast), the results on both datasets are at a low level. As λ_0 and λ_1 increase, cross-scale contrast information is introduced, leading to a significant improvement in the results. In general, using cross-scale contrast outperforms using contrast between output embeddings. Moreover, it is interesting that although the optimal parameter combinations may vary across different datasets, contrasting with the 0-th layer embedding, i.e., the projection of original feature to embedding space, is beneficial across all datasets. Given that the 0-th layer embedding contains the purest consistency information, this result indicates that our cross-scale contrast can preserve more of this information.



(**a**) Cora (**b**) Photo

Figure 3. Hyper-parameter λ analysis (Acc (%) for 3 seeds) on Cora and Photo. The combinations of λ that cannot be taken among them are set to the average.

4.7. Runtime Analysis

In this section, we empirically validate that MUC-GCL introduces minimal additional computational cost compared to typical GCL methods. Here, we compare the training time of several advanced GCL methods with that of MUX-GCL. Table 6 presents the running time of different GCL methods for each epoch on the four datasets. For methods such as GRACE that do not rely on extra computation modules (e.g., false negative detections), MUX-GCL only slightly increases the training time but achieves a huge improvement. As for other sophisticated methods, MUX-GCL is comparable to ProGCL, which also employs a one-time false negative detection strategy, and is far cheaper than HomoGCL, which updates the saliency once per epoch. The above experimental results confirm that MUX-GCL can achieve significant performance improvements without explicitly increasing computational costs.

Table 6. Time per epoch for GCL mehtods (on 24GB RTX 3090Ti GPU).

Model	Cora	Citeseer	Photo	Computer
GRACE	0.20 s	0.02 s	0.05 s	0.12 s
ProGCL	0.04 s	0.05 s	0.17 s	0.49 s
HomoGCL	1.09 s	0.48 s	0.50 s	1.32 s
MA-GCL	0.19 s	0.02 s	0.04 s	0.08 s
MUX-GCL	0.04 s	0.05 s	0.16 s	0.42 s

5. Conclusions

In this paper, we introduced MUX-GCL, a novel cross-scale contrastive learning paradigm that leverages multiplex representations to extract richer and more consistent information from graphs while mitigating disturbing features. By introducing a patch contrasting strategy based on topological affinities, MUX-GCL effectively alleviates the issue of false negative pairs in cross-scale contrasts, a common challenge in InfoNCE-based GCL methods. Our theoretical analysis proves that the objective function of MUX-GCL serves as a stricter lower bound of mutual information between raw features and learned representations, providing a solid foundation for its superior performance. Extensive experiments demonstrate that MUX-GCL outperforms state-of-the-art GCL models on both classification and clustering tasks. Despite its strengths, MUX-GCL faces the limitations with message passing, which forbids the use of a large number of hidden layers and thus the maximal scale of effective patches. Such multiplex cross-scale constrastive learning paradigms will be benefited by future work that explores multiple ways of constructing effective patches while maintaining the encoder's expressive power.

Author Contributions: Conceptualization: W.C. and L.C.; formal analysis: W.C. and J.Z.; funding acquisition: W.C.; investigation: Z.Z. and M.Z.; validation: C.W. and S.W.; writing—original draft: Z.Z. and M.Z.; writing—review and editing: L.C. and W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by Excellence Fund of Soochow University NH11800823.

Data Availability Statement: The datasets and codes can be publicly available at GitHub https://github.com/MUX-GCL/MUX-GCL/ (accessed on 21 May 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

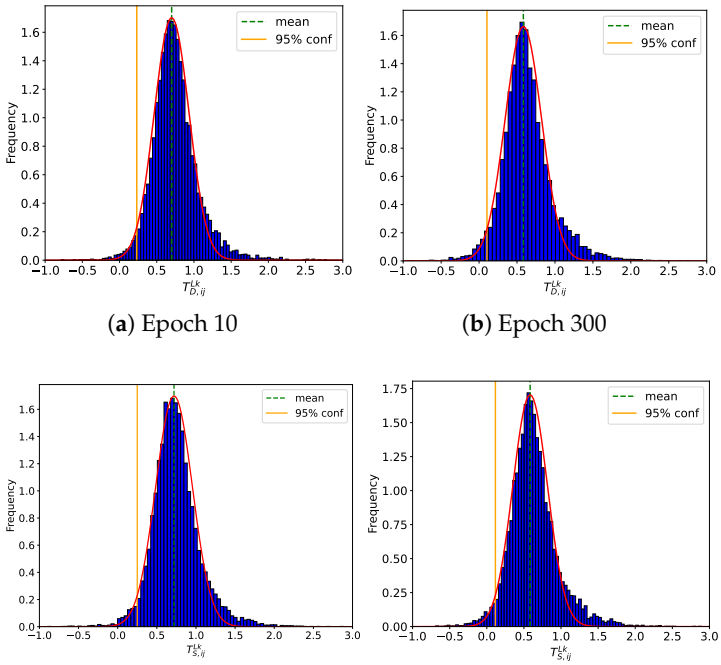
Appendix A

Table A1. Performance of node classification (Acc (%) ± std for 5 seeds). Here we use public splits on Cora, Citeseer, and Pubmed. The highest performance is highlighted in boldface.

Model	Cora	Citeseer	Pubmed
DGI	82.3 ± 0.6	71.8 ± 0.7	76.8 ± 0.6
MVGRL	83.5 ± 0.4	**73.3 ± 0.5**	80.1 ± 0.7
GRACE	81.5 ± 0.3	70.6 ± 0.5	80.2 ± 0.3
GCA	81.4 ± 0.3	70.4 ± 0.4	80.7 ± .5
ProGCL	81.2 ± 0.4	69.8 ± 0.5	79.2 ± 0.2
BGRL	82.7 ± 0.6	71.1 ± 0.8	79.6 ± 0.5
COSTA	82.2 ± 0.2	70.7 ± 0.5	80.4 ± 0.3
CCA-SSG	84.0 ± 0.4	73.1 ± 0.3	81.0 ± 0.4
HomoGCL	84.5 ± 0.5	72.3 ± 0.7	81.1 ± 0.3
MUX-GCL	**84.7 ± 0.2**	72.5 ± 0.1	**82.2 ± 0.2**

Appendix A.1. Tests on Public Splits

In alignment with some previous GCL methods that use public splits on Cora, Citeseer, and PubMed, we also investigate another benchmark setting with public splits on these three datasets and compare it with the most competitive baselines. As shown in Table 2, MUX-GCL consistently outperforms the baseline methods, demonstrating its effectiveness.



(**a**) Epoch 10 (**b**) Epoch 300

(**c**) Epoch 10 (**d**) Epoch 300

Figure A1. $T^{Lk}_{D,ij}$ and $T^{Lk}_{S,ij}$ values distribution during training process of Cora. We take out the embeddings of our encoder trained at epoch 10 and 300, respectively, to calculate $T^{Lk}_{D,ij}$ and $T^{Lk}_{S,ij}$, respectively. Gaussian curves are fitted to the values of $T^{Lk}_{D,ij}$ and $T^{Lk}_{S,ij}$ at epochs 10 and 300, respectively.

Appendix A.2. Proof of Proposition 1

We provide a semi-empirical proof for Proposition 1: *The cross-patch contrastive objective in Equation (2) is a lower bound of the mutual information between the raw input features* $\mathbf{X}$ *and output node embeddings* $\mathbf{U}$ *and* $\mathbf{V}$ *in two augmented views. Formally,* $\mathcal{L}_{MUX} < I(\mathbf{X};\mathbf{U},\mathbf{V})$. *Moreover, with a statistical significance, the objective is also a stricter lower bound comparing with the contrastive objective* $\mathcal{L}_{GR}$ *in Equation (1) proposed by GRACE:* $\mathcal{L}_{MUX} > \mathcal{L}_{GR}$.

Proof. We first prove $\mathcal{L}_{MUX} < I(\mathbf{X};\mathbf{U},\mathbf{V})$. Let $\mathbf{U}^{(k)}, \mathbf{V}^{(k)}$ (for $k = 0, 1, \ldots, L$) be the embeddings generated by the k-th layer of the encoder, where the final output $\mathbf{U} = \mathbf{U}^{(L)}$ and $\mathbf{V} = \mathbf{V}^{(L)}$. Our proposed objective includes $2(L+1)$ cross-scale contrasting pairs

$$\mathcal{L}_{MUX} = \frac{1}{2}\sum_{k=0}^{L}\frac{\lambda_k}{N}\sum_{i=1}^{N}\Big[\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) + \mathcal{L}_c(v_i^{(L)}, u_i^{(k)})\Big]. \tag{A1}$$

where the terms in the brackets are symmetric regarding the two views. We now focus on the former term, which is comprised of positive and negative contrasting terms

$$\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) = \log\frac{Pos}{Pos + Neg}, \tag{A2}$$

with $Pos = \mathrm{e}^{\theta(u_i^{(L)}, v_i^{(k)})}$ and $Neg = \sum_{j\neq i}\omega_{ij}^{Lk}\left[\mathrm{e}^{\theta(u_i^{(L)}, u_j^{(k)})} + \mathrm{e}^{\theta(u_i^{(L)}, v_j^{(k)})}\right]$.

For a sufficiently large N, we have $\omega_{ij}^{Lk} > 1/N$ and hence

$$Neg > \frac{1}{N}\sum_{j\neq i}\left[e^{\theta(u_i^{(L)},u_j^{(k)})} + e^{\theta(u_i^{(L)},v_j^{(k)})}\right] > \frac{1}{N}\sum_{j\neq i} e^{\theta(u_i^{(L)},v_j^{(k)})}. \tag{A3}$$

Substituting this inequality into Equation (A2) gives

$$\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) < \log \frac{e^{\theta(u_i^{(L)},v_i^{(k)})}}{\frac{1}{N}e^{\theta(u_i^{(L)},v_i^{(k)})} + \frac{1}{N}\sum\limits_{j\neq i}^{N} e^{\theta(u_i^{(L)},v_j^{(k)})}}. \tag{A4}$$

By averaging all nodes, we obtain

$$\begin{aligned}
\mathbb{E}^{Lk}[\mathcal{L}_c] &= \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) \\
&< \mathbb{E}\left[\log \frac{e^{\theta(u_i^{(L)},v_i^{(k)})}}{\frac{1}{N}e^{\theta(u_i^{(L)},v_i^{(k)})} + \frac{1}{N}\sum\limits_{j\neq i}^{N} e^{\theta(u_i^{(L)},v_j^{(k)})}}\right] \\
&= I_{NCE}(\mathbf{U}^{(L)}, \mathbf{V}^{(k)}).
\end{aligned} \tag{A5}$$

As InfoNCE is a lower bound of MI, we conclude that

$$\mathbb{E}^{Lk}[\mathcal{L}_c] < I_{NCE}(\mathbf{U}^{(L)}, \mathbf{V}^{(k)}) \leq I(\mathbf{U}^{(L)}, \mathbf{V}^{(k)}). \tag{A6}$$

Consequently, with both symmetric contrasting terms, we have

$$\mathcal{L}_{MUX} < \frac{1}{2}\sum_{k=0}^{L}\lambda_k\Big[I(\mathbf{U}^{(L)}; \mathbf{V}^{(k)}) + I(\mathbf{V}^{(L)}; \mathbf{U}^{(k)})\Big]. \tag{A7}$$

Resorting to the relations derived for GRACE [7],

$$I(\mathbf{U}^{(L)}; \mathbf{V}^{(k)}) \leq I(\mathbf{X}; \mathbf{U}^{(L)}) = I(\mathbf{X}; \mathbf{U}) \leq I(\mathbf{X}; \mathbf{U}, \mathbf{V}), \tag{A8}$$

$$I(\mathbf{V}^{(L)}; \mathbf{U}^{(k)}) \leq I(\mathbf{X}; \mathbf{V}^{(L)}) = I(\mathbf{X}; \mathbf{V}) \leq I(\mathbf{X}; \mathbf{V}, \mathbf{U}), \tag{A9}$$

and noticing the layer-wise coefficients are normalized $\sum\limits_{k=0}^{L}\lambda_k = 1$, we finally have

$$\mathcal{L}_{MUX} < \frac{1}{2}\sum_{k=0}^{L}\lambda_k[I(\mathbf{X}; \mathbf{U}, \mathbf{V}) + I(\mathbf{X}; \mathbf{V}, \mathbf{U})] = I(\mathbf{X}; \mathbf{U}, \mathbf{V}). \tag{A10}$$

We then show that with a statistical significance $\mathcal{L}_{MUX} > \mathcal{L}_{GR}$. We first rewrite the loss function as

$$\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) = \frac{1}{1 + \sum\limits_{j\neq i}\omega_{ij}^{Lk}(e^{\psi_{S,ij}^{Lk}} + e^{\psi_{D,ij}^{Lk}})}, \tag{A11}$$

where

$$\psi_{S,ij}^{Lk} = \theta(u_i^{(L)}, u_j^{(k)}) - \theta(u_i^{(L)}, v_i^{(k)}), \tag{A12a}$$

$$\psi_{D,ij}^{Lk} = \theta(u_i^{(L)}, v_j^{(k)}) - \theta(u_i^{(L)}, v_i^{(k)}). \tag{A12b}$$

The loss function of GRACE can then be written as

$$\mathcal{L}_{GR} = \frac{1}{2}\sum_{i=1}^{N}\Big[\mathcal{L}_g(u_i^{(L)}, v_i^{(k)}) + \mathcal{L}_g(v_i^{(L)}, u_i^{(k)})\Big], \tag{A13}$$

with

$$\mathcal{L}_g(u_i^{(L)}, v_i^{(L)}) = \frac{1}{1 + \sum_{j\neq i}(e^{\psi_{S,ij}^{Lk}} + e^{\psi_{D,ij}^{Lk}})}. \tag{A14}$$

Next, we define

$$T_{S,ij}^{Lk} = \psi_{S,ij}^{Lk} - \psi_{S,ij}^{LL} + \log\omega_{ij}^{Lk}, \tag{A15}$$

$$T_{D,ij}^{Lk} = \psi_{D,ij}^{Lk} - \psi_{D,ij}^{LL} + \log\omega_{ij}^{Lk}. \tag{A16}$$

From the statistics, we show that throughout the training, both quantities $T_{S,ij}^{Lk}$ and $T_{D,ij}^{Lk}$ are positive with a great statistical significance. Concretely, as shown in Figure A1, the histograms of these quantities can be well fitted by Gaussian curves at all epochs ranging from 10 to 300 and $T_{S,ij}^{Lk} > 0$ and $T_{D,ij}^{Lk} > 0$ (for $k < L$) within the 95% confidence interval. By comparing the denominators in Equations (A11) and (A14), we can conclude that with a large probability, $\mathcal{L}_c(u_i^{(L)}, v_i^{(k)}) > \mathcal{L}_g(u_i^{(L)}, v_i^{(L)})$. Symmetrically, we also have $\mathcal{L}_c(v_i^{(L)}, u_i^{(k)}) > \mathcal{L}_g(v_i^{(L)}, u_i^{(L)})$. These relations also hold for $k = L$ since $\omega_{ij}^{LL} \in (0, 1)$ for $j \neq i$. Hence, by comparing the total objectives defined in Equations (A1) and (A13), we can finally reach

$$\mathcal{L}_{MUX} > \mathcal{L}_{GR}, \tag{A17}$$

which concludes the proof. □

References

1. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
2. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903.
3. Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; Weinberger, K. Simplifying graph convolutional networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 6861–6871.
4. Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How powerful are graph neural networks? *arXiv* **2018**, arXiv:1810.00826.
5. Xu, B.; Shen, H.; Cao, Q.; Qiu, Y.; Cheng, X. Graph wavelet neural network. *arXiv* **2019**, arXiv:1904.07785.
6. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 13–18 July 2020; pp. 1597–1607.
7. Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; Wang, L. Deep graph contrastive representation learning. *arXiv* **2020**, arXiv:2006.04131.
8. You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; Shen, Y. Graph contrastive learning with augmentations. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 5812–5823.
9. Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; Wang, L. Graph contrastive learning with adaptive augmentation. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 2069–2080.
10. Veličković, P.; Fedus, W.; Hamilton, W.L.; Liò, P.; Bengio, Y.; Hjelm, R.D. Deep graph infomax. *arXiv* **2018**, arXiv:1809.10341.
11. Hassani, K.; Khasahmadi, A.H. Contrastive multi-view representation learning on graphs. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 13–18 July 2020; pp. 4116–4126.
12. Mavromatis, C.; Karypis, G. Graph infoclust: Maximizing coarse-grain mutual information in graphs. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Virtual, 11–14 May 2021; Springer: Berlin/Heidelberg, Germany, pp. 541–553.
13. Cao, J.; Lin, X.; Guo, S.; Liu, L.; Liu, T.; Wang, B. Bipartite graph embedding via mutual information maximization. In Proceedings of the ACM International Conference on Web Search and Data Mining, Virtual, 8–12 March 2021; pp. 635–643.
14. Wang, C.; Liu, Z. Learning graph representation by aggregating subgraphs via mutual information maximization. *arXiv* **2021**, arXiv:2103.13125.

15. Xu, D.; Cheng, W.; Luo, D.; Chen, H.; Zhang, X. InfoGCL: Information-aware graph contrastive learning. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 30414–30425.
16. Xia, J.; Wu, L.; Wang, G.; Chen, J.; Li, S.Z. ProGCL: Rethinking hard negative mining in graph contrastive learning. *arXiv* **2021**, arXiv:2110.02027.
17. Niu, C.; Pang, G.; Chen, L. Affinity Uncertainty-based Hard Negative Mining in Graph Contrastive Learning. *arXiv* **2023**, arXiv:2301.13340. [CrossRef] [PubMed]
18. Li, W.Z.; Wang, C.D.; Xiong, H.; Lai, J.H. HomoGCL: Rethinking Homophily in Graph Contrastive Learning. *arXiv* **2023**, arXiv:2306.09614.
19. Liu, Y.; Jin, M.; Pan, S.; Zhou, C.; Zheng, Y.; Xia, F.; Philip, S.Y. Graph self-supervised learning: A survey. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 5879–5900. [CrossRef]
20. Thakoor, S.; Tallec, C.; Azar, M.G.; Azabou, M.; Dyer, E.L.; Munos, R.; Veličković, P.; Valko, M. Large-scale representation learning on graphs via bootstrapping. *arXiv* **2021**, arXiv:2102.06514.
21. Bielak, P.; Kajdanowicz, T.; Chawla, N.V. Graph barlow twins: A self-supervised representation learning framework for graphs. *Knowl.-Based Syst.* **2022**, *256*, 109631. [CrossRef]
22. Zhang, H.; Wu, Q.; Yan, J.; Wipf, D.; Yu, P.S. From canonical correlation analysis to self-supervised graph neural networks. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 76–89.
23. Grill, J.B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; et al. Bootstrap your own latent-a new approach to self-supervised learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 21271–21284.
24. You, Y.; Chen, T.; Shen, Y.; Wang, Z. Graph contrastive learning automated. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 18–24 July 2021; pp. 12121–12132.
25. Sun, F.Y.; Hoffmann, J.; Verma, V.; Tang, J. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. *arXiv* **2019**, arXiv:1908.01000.
26. Chu, G.; Wang, X.; Shi, C.; Jiang, X. CuCo: Graph Representation with Curriculum Contrastive Learning. In Proceedings of the IJCAI, Montreal, QC, Canada, 19–27 August 2021; pp. 2300–2306.
27. Grover, A.; Leskovec, J. node2vec: Scalable feature learning for networks. In Proceedings of the Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864.
28. Kipf, T.N.; Welling, M. Variational graph auto-encoders. *arXiv* **2016**, arXiv:1611.07308.
29. Yang, Z.; Cohen, W.; Salakhudinov, R. Revisiting semi-supervised learning with graph embeddings. In Proceedings of the International Conference on Machine Learning, PMLR, New York, NY, USA, 19–24 June 2016; pp. 40–48.
30. Shchur, O.; Mumme, M.; Bojchevski, A.; Günnemann, S. Pitfalls of graph neural network evaluation. *arXiv* **2018**, arXiv:1811.05868.
31. Perozzi, B.; Al-Rfou, R.; Skiena, S. DeepWalk: Online learning of social representations. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 701–710.
32. Tsitsulin, A.; Palowitch, J.; Perozzi, B.; Müller, E. Graph clustering with graph neural networks. *J. Mach. Learn. Res.* **2023**, *24*, 5809–5829
33. Mo, Y.; Peng, L.; Xu, J.; Shi, X.; Zhu, X. Simple unsupervised graph representation learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 27 February–2 March 2022; Volume 36, pp. 7797–7805.
34. Zhang, Y.; Zhu, H.; Song, Z.; Koniusz, P.; King, I. COSTA: Covariance-preserving feature augmentation for graph contrastive learning. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 14–18 August 2022; pp. 2524–2534.
35. Zhang, Y.; Zhu, H.; Song, Z.X.; Koniusz, P.; King, I. Spectral feature augmentation for graph contrastive learning and beyond. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 11289–11297.
36. Gong, X.; Yang, C.; Shi, C. MA-GCL: Model augmentation tricks for graph contrastive learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 4284–4292.
37. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
38. Rand, W.M. Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.* **1971**, *66*, 846–850. [CrossRef]