Boosting Graph Contrastive Learning via Adaptive Sampling

Sheng Wan, Yibing Zhan, *Member, IEEE*, Shuo Chen, Shirui Pan, *Senior Member, IEEE*, Jian Yang, *Member, IEEE*, Dacheng Tao, *Fellow, IEEE*, and Chen Gong, *Senior Member, IEEE*

***Abstract*— Contrastive learning (CL) is a prominent technique for self-supervised representation learning, which aims to contrast semantically similar (i.e., positive) and dissimilar (i.e., negative) pairs of examples under different augmented views. Recently, CL has provided unprecedented potential for learning expressive graph representations without external supervision. In graph CL, the negative nodes are typically uniformly sampled from augmented views to formulate the contrastive objective. However, this uniform negative sampling strategy limits the expressive power of contrastive models. To be specific, not all the negative nodes can provide sufficiently meaningful knowledge for effective contrastive representation learning. In addition, the negative nodes that are semantically similar to the anchor are undesirably repelled from it, leading to degraded model performance. To address these limitations, in this article, we devise an adaptive sampling strategy termed "AdaS." The proposed AdaS framework can be trained to adaptively encode the importance of different negative nodes, so as to encourage learning from the most informative graph nodes. Meanwhile, an auxiliary polarization regularizer is proposed to suppress the adverse impacts of the false negatives and enhance the discrimination ability of AdaS. The experimental results on a variety of real-world datasets firmly verify the effectiveness of our AdaS in improving the performance of graph CL.**

***Index Terms*— Contrastive graph representation learning, graph convolutional network, negative mining.**

Manuscript received 10 March 2022; revised 6 August 2022, 18 November 2022, and 8 March 2023; accepted 20 June 2023. This work was supported in part by the NSF of China under Grant 61973162 and Grant U1713208, in part by the NSF of Jiangsu Province under Grant BZ2021013, in part by the NSF for Distinguished Young Scholar of Jiangsu Province under Grant BK20220080, in part by the Fundamental Research Funds for the Central Universities under Grant 30920032202 and Grant 30921013114, in part by the CAAI-Huawei MindSpore Open Fund, in part by the "111" Program under Grant B13022, and in part by the Program for Changjiang Scholars. *(Corresponding author: Chen Gong.)*

Sheng Wan, Jian Yang, and Chen Gong are with the PCA Laboratory, the Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, the Jiangsu Key Laboratory of Image and Video Understanding for Social Security, and the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: wansheng315@hotmail.com; csjyang@njust.edu.cn; chen.gong@njust.edu.cn).

Yibing Zhan is with JD Explore Academy, JD.com, Beijing 100176, China (e-mail: zhanyibing@jd.com).

Shuo Chen is with the RIKEN Center for Advanced Intelligence Projection, Tokyo 351-0198, Japan (e-mail: shuo.chen.ya@riken.jp).

Shirui Pan is with the School of Information and Communication Technology, Griffith University, Gold Coast, QLD 4215, Australia (e-mail: s.pan@griffith.edu.au).

Dacheng Tao is with the School of Computer Science, Faculty of Engineering, The University of Sydney, Darlington, NSW 2008, Australia (e-mail: dacheng.tao@sydney.edu.au).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2023.3291358.

Digital Object Identifier 10.1109/TNNLS.2023.3291358

NOMENCLATURE

$\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$	Graph $\mathcal{G}$ with the node set $\mathcal{V}$ and edge set $\mathcal{E}$.
$\mathbf{A}$	Adjacency matrix of $\mathcal{G}$.
$\mathbf{x}_i$	Feature vector of the ith graph node.
$\mathbf{X}$	Feature matrix of $\mathcal{G}$ with the ith row corresponding to $\mathbf{x}_i$.
$(\mathbf{x}'_i, \mathbf{x}''_i)$	Two augmented versions of $\mathbf{x}_i$.
f	GNN encoder.
$h(\mathbf{x}_i)$	Class label of $\mathbf{x}_i$.

I. INTRODUCTION

GRAPH representation learning is a fundamental task in various applications, such as molecular properties' prediction in drug discovery [1] and community analysis in social networks [2]. Recently, graph neural networks (GNNs) have received a surge of research attention and showed their effectiveness in learning graph representations [3], [4]. However, most existing GNN models are trained in a supervised fashion, and thus hinge on the availability of a large quantity of label information that is usually expensive to collect [5]. To address this challenge, self-supervised approaches [6], [7], [8] are coupled with GNN models to enable graph representation learning with unlabeled data. Among many others, contrastive learning (CL) has emerged as a powerful tool and shown its capabilities to learn generalizable, transferable, and robust graph representations [9].

CL owes its success to the "alignment" of features from positive pairs and the "uniformity" of representations on the hypersphere [10]. Here, the "alignment" favors encoders that assign similar features to positive pairs, while the "uniformity" depends on the separation of negative pairs. Empirically, the positive pairs are often acquired by taking two independently randomly augmented versions of the same example, and steady progress has been made in developing data augmentation techniques [6], [11], [12], [13], [14], such as random cropping and rotation of images [6]. However, these techniques are unavailable for graph-structured data, which possess inherent non-Euclidean properties. Fortunately, researchers have explored diverse types of effective and efficient augmentation techniques for graph-structured data [5], [9], [15], [15], [16],

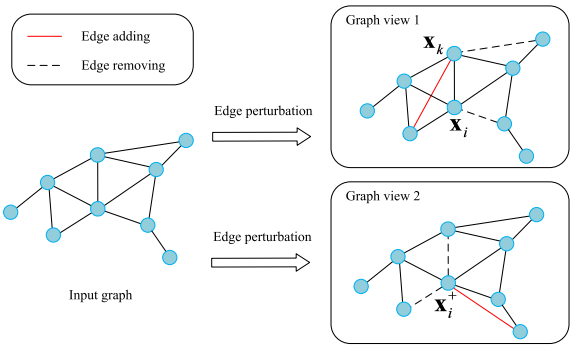


Fig. 1. Example illustrating the potential risk of edge perturbation to graph CL. Two augmented graphs can be generated using different edge perturbation operations, which have been shown in graph views 1 and 2, respectively. For graph CL, the nodes $(\mathbf{x}_k, \mathbf{x}_i)$ in graph view 1 form a negative pair naturally. However, it is noticeable that the neighborhood of $\mathbf{x}_k$ and $\mathbf{x}_i$ is highly overlapped. Since the graph convolution operation aims to aggregate information from the neighborhood, the representations of $\mathbf{x}_k$ and $\mathbf{x}_i$ tend to be homogeneous with successive graph convolution, and $(\mathbf{x}_k, \mathbf{x}_i)$ could be a false negative pair. Finally, repelling $(\mathbf{x}_k, \mathbf{x}_i)$ in the embedding space may not provide meaningful information for contrastive representation learning.

such as edge perturbation and feature masking, to generate faithful positive pairs.

To achieve the desirable "uniformity" characteristics in contrastive representation learning, negative pairs should be pushed apart in an embedding space [10]. However, most existing graph CL methods assume that all the negative nodes are equally important and they usually uniformly sample negative nodes [5], [11], [17]. This could result in limited representation power, since not all the negatives can provide informative signals for CL. Specifically, the negatives that are hard to discriminate from the anchor are the most beneficial to the learning objectives, while the easy-to-discriminate negatives provide less benefit [18], [19]. In addition, the uniform sampling strategy might involve false negatives, which are semantically similar to the anchor and should be considered as positives instead [20]. This sampling error caused by false negatives will damage the structure of the embedding space, leading to degenerated model performance. Moreover, the commonly used graph augmentation techniques might impair graph structures or node features, inevitably resulting in more false negatives. This can exacerbate the problem of sampling error and further degrade the representation ability of the contrastive models. Fig. 1 provides a detailed explanation of the frequently used augmentation technique, i.e., edge perturbation, with a toy example.

To boost the performance of graph CL, we focus on the design of the sampling strategy for negative nodes, to facilitate contrastive representation learning with the most informative nodes. According to hard negative mining [21], the negatives close to the anchor (also termed "hard negatives") are most useful and provide significant gradient information during training. However, in practice, the hard negatives sampled in this way [21] may not be truly informative for graph CL, as some of them can be false negatives. Apart from this, manually choosing the hard negatives may yield an objective that no longer bounds mutual information (MI), and thereby removing a theoretical connection that is critical to graph CL and the downstream tasks. Considering the aforementioned limitations, in this work, we design an adaptive sampling strategy and develop a new framework dubbed "AdaS," which is applicable to most existing graph CL methods. To encourage learning from the most informative nodes, our AdaS adaptively reweights all the negatives for an anchor. Specifically, in AdaS, a tunable sampling distribution is designed to emphasize the importance of the negatives that are not too hard or too easy to discriminate in the embedding space. Besides, an auxiliary polarization regularizer is used to enhance the influence of the informative nodes. As a consequence, the devised sampling strategy can not only exploit the hard levels of embeddings but also suppress the sampling error caused by false negatives. To summarize, the main contributions of this article are as follows.

1) We design a general framework for unsupervised contrastive graph representation learning, where the importance of different graph nodes can be adaptively encoded during model training.
2) We implicitly devise a novel sampling strategy, which can be further enhanced by an auxiliary polarization regularizer. By this means, the proposed AdaS encourages learning from the most informative negatives and suppresses the sampling error in graph CL, simultaneously.
3) Systematic studies have been performed on node classification tasks using various public benchmark datasets, revealing the effectiveness of our proposed AdaS framework.

II. Related Work

In this section, we will review some representative works on GNNs and contrastive representation learning, since they are closely related to this article.

A. Graph Neural Networks

As one of the hottest topics, GNN extends the deep neural networks by defining convolutions and readout operations on irregular graph-structured data [3], [17], [22], [23]. The concept of GNN was first proposed in [24]. Afterward, various spectral-based GNNs were proposed to define filters from the perspective of graph signal processing [25], [26]. Particularly, GCN [27], which performed a localized first-order approximation of spectral graph convolution, has demonstrated its power in node representation learning [28], [29], [30]. In follow-up works, Chen et al. [31] designed an efficient GCN model named FastGCN for inductive node classification, which can be further enhanced via importance sampling. Wu et al. [32] proposed the simple graph convolution to significantly reduce the complexity of GCN, where the nonlinearities are replaced with a single linear transformation.

In addition to the spectral GNNs, research efforts have also been made in the spatial methods, which directly define convolution for each node as a weighted average function over its neighboring nodes [33]. For instance, GraphSAGE [34] defined the weighting function as various aggregators over neighboring nodes. In graph attention networks (GATs) [35],

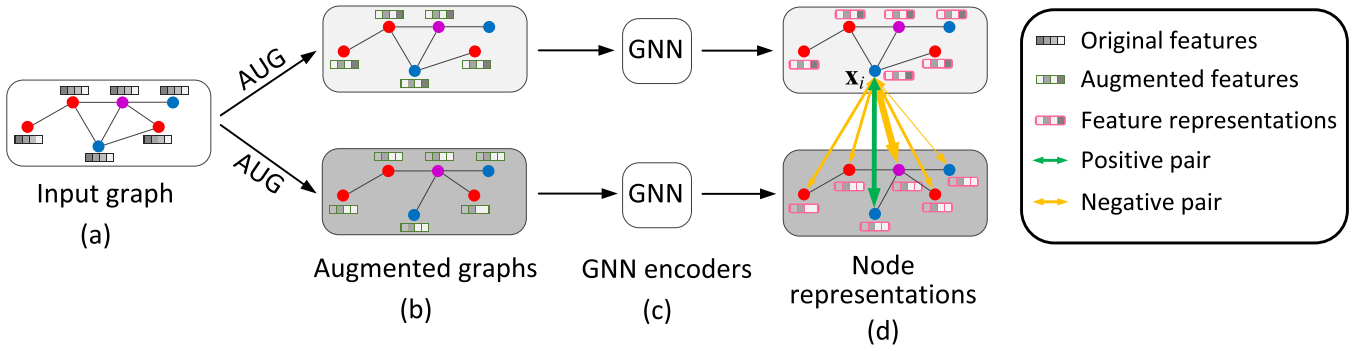


Fig. 2. Conceptual framework of our proposed algorithm. (a) Input graph. (b) Graph views generated by augmentation techniques. (c) GNN encoders, which can be shared across different graph views. In (d), the representations are produced by the encoders. For each anchor $\mathbf{x}_i$, the importance of its negatives is adaptively encoded to improve the performance of graph CL. Here, the linewidth of the orange arrows denotes the corresponding importance, and the green arrow connects the positive node pair.

the weighting function was defined by the learnable self-attention mechanism. To handle large-scale graphs and use deep architectures simultaneously, Cluster-GCN [36] was devised based on a graph clustering algorithm, which performed graph convolution within the sampled subgraph efficiently.

B. Contrastive Representation Learning

As a significant brunch of self-supervised learning [37], [38], contrastive methods aim at learning discriminative representations by contrasting the positive pairs against negative pairs. In [39], Deep InfoMax was proposed to learn the embeddings of images by maximizing the MI between a local patch and its global context. Then the framework of contrastive predictive coding was presented in [40], where a probabilistic contrastive objective was used to capture information for future sample prediction. Besides, MoCo [11] used a momentum-updated encoder to update the network parameters and generate contrastive embeddings. In addition, other contrastive methods, such as SimCLR [6], BYOL [41], and SimSiam [42], have also attracted increasing research attention. It is noteworthy that conventional contrastive objective used in most existing CL methods is biased, since the semantically similar data pairs (i.e., false negatives) could be pushed apart during the repelling of all the negative pairs [43]. To address this issue, Li et al. [44] applied the clustering method to the generated embeddings to gather similar instances, but the reliability of the clustering results depended largely on the learned embeddings. In [20], positive-unlabeled (PU) learning [45] was adopted to correct for the sampling of same-label data points.

By adapting the idea of CL to graph domains, graph CL has emerged with promising representation learning performance [46], [47]. For example, DGI [9] married the power of GNN and CL by maximizing the MI between the global graph-level and local node-level embeddings. Inspired by DGI, a multiview graph CL framework was proposed in [48], where the graph diffusion [49] was adopted to augment the original graph view. Different from DGI, Zhu et al. [15] focused on maximizing the agreement of node embeddings, rather than graph embeddings, across two randomly corrupted graph views. To further enhance the expressive power of the representations, the graph CL methods focus on developing more flexible data augmentation techniques, such as [16], [50], and [51]. Most existing graph CL methods focus on developing graph augmentation techniques to produce reliable positive node pairs, while neglecting the impacts of negative pairs. Consequently, we propose a negative sampling strategy termed "AdaS," which can boost the performance of graph CL by adaptively learning from the most informative negatives.

It is worth noting that a previous work HBNM [18] also develops a sampling strategy to select informative negative examples. However, there are two primary differences between our work and HBNM. First, the sampling strategy in HBNM simply upweights the negatives that are close to the anchor, while our proposed AdaS adaptively encodes the importance of all the negatives. As a result, our AdaS encourages learning from the most informative negatives. Second, HBNM takes the viewpoint of PU learning [45] to address the issue of false negatives. Specifically, HBNM decomposes the true negative distribution and assumes a close-to-uniform class distribution, which may not always hold in practice. Consequently, the false negatives may possess relatively large importance weights in CL. Unlike HBNM, the proposed AdaS devises an auxiliary polarization regularizer to explicitly suppress the importance weights of false negatives, which is independent of the class-balance assumption.

III. Methodology

This section details our proposed AdaS framework, of which the schematic is exhibited in Fig. 2. In the following, the critical steps will be detailed by introducing the graph CL setup (Section III-B), explaining the adaptive sampling technique (Section III-C), elaborating the polarization regularizer (Section III-D), and describing the ultimate contrastive objective (Section III-E).

A. Preliminaries

Let $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ denote a graph, where $\mathcal{V}$ is the node set containing all the nodes/examples and $\mathcal{E}$ is the edge set modeling the similarity among the nodes. We denote $\mathbf{X} \in \mathbb{R}^{n \times d}$ as the feature matrix with the ith row formed by the feature

vector of the ith node (i.e., $\mathbf{x}_i$). The adjacency matrix of $\mathcal{G}$ is denoted as $\mathbf{A} \in \mathbb{R}^{n\times n}$, where $\mathbf{A}_{ij} = 1$ if there exists an edge between $\mathbf{x}_i$ and $\mathbf{x}_j$ and $\mathbf{A}_{ij} = 0$ otherwise. In our method, class information of the nodes in $\mathcal{G}$ is unavailable. The objective is to learn a GNN encoder $f(\mathbf{X}, \mathbf{A})$ which receives the feature and adjacency matrices of the graph as input and produces embeddings for each graph node. The learned node representations can be further used in downstream tasks, such as node classification and community detection [16]. Some important notations used in this article are summarized in Nomenclature.

B. Graph CL Setup

To ensure the compatibility of our proposed AdaS framework, we follow the paradigm that is widely adopted by the existing contrastive GNN models. Concretely, two different graph views $\mathcal{G}'$ and $\mathcal{G}''$ are first generated via graph augmentation techniques, wherein $(\mathbf{x}'_i, \mathbf{x}''_i)$ can be obtained from $\mathbf{x}_i$ correspondingly. Then representations can be obtained from the two graph views after successive graph convolution. Here, a pairwise contrastive objective $\mathcal{L}_{\text{pw}}(\mathbf{x}'_i, \mathbf{x}''_i)$ [5], [15], [16] can be used to enforce maximizing the consistency between the representations of a node in the two views, namely,

$$-\log \frac{e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}''_i)/\sigma}}{e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}''_i)/\sigma} + \sum_{j\neq i} e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}'_j)/\sigma} + \sum_{j\neq i} e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}''_j)/\sigma}} \tag{1}$$

where $f(\mathbf{x}'_i)$ denotes the embeddings of $\mathbf{x}'_i$, and $\sigma > 0$ is the temperature parameter. In (1), the negative examples are composed of all the nodes in the two graph views except for $\mathbf{x}'_i$ and $\mathbf{x}''_i$. Note that the objective of graph CL can be characterized more analytically based on (1), so that our proposed AdaS algorithm can be derived naturally.

Following the setup of [18], [20], [52], we can reasonably assume that $\mathcal{C}$ is an underlying set of discrete latent classes representing semantic content and the pair of nodes $(\mathbf{x}'_i, \mathbf{x}''_i)$ belong to the same latent class. Denoting the distribution over the latent class by $\rho(c)$ for $c \in \mathcal{C}$, the joint distribution can be defined as $p_{\mathbf{x}'_i,c}(\mathbf{x}'_i, c) = p(\mathbf{x}'_i|c)\rho(c)$. Let $h(\mathbf{x}'_i)$ represent the class label of $\mathbf{x}'_i$, and then the probability of observing $\mathbf{x}'_j$ as a positive example for $\mathbf{x}'_i$ can be expressed as $p^+_{\mathbf{x}'_i}(\mathbf{x}'_j) = p(\mathbf{x}'_j|h(\mathbf{x}'_j) = h(\mathbf{x}'_i))$ and the probability of a negative example can be expressed as $p^-_{\mathbf{x}'_i}(\mathbf{x}'_j) = p(\mathbf{x}'_j|h(\mathbf{x}'_j) \neq h(\mathbf{x}'_i))$. For simplicity, let $\mathbf{x}'_i \sim p$ denote a node sampled from p. Then the ideal pairwise contrastive loss $\tilde{\mathcal{L}}_{\text{pw}}(\mathbf{x}'_i, \mathbf{x}'^+_i)$ can be obtained as

$$\mathbb{E}_{\substack{\mathbf{x}'_i\sim p, \mathbf{x}'^+_i\sim p^+_{\mathbf{x}'_i} \\ \mathbf{x}'^-_j\sim q}} \left[-\log \frac{e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}'^+_i)/\sigma}}{e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}'^+_i)/\sigma} + \frac{Q}{N}\sum_{j=1}^{N} e^{f(\mathbf{x}'_i)^\top f(\mathbf{x}'^-_j)/\sigma}} \right] \tag{2}$$

where the positive example $\mathbf{x}'^+_i$ has the same label as $\mathbf{x}'_i$, the N negative examples $\{\mathbf{x}'^-_j\}_{i=1}^N$ sampled from the negative sampling distribution q have different labels with the anchor $\mathbf{x}'_i$, p is the marginal distribution $p(\mathbf{x}'_i)$ of $p_{\mathbf{x}'_i,c}(\mathbf{x}'_i, c)$, the number of negative examples $N = 2n - 2$, and Q is the weighting parameter introduced for the analysis. When the number of negative examples N is finite, we set $Q = N$, in agreement with the standard contrastive objective. In practice, the negative sampling distribution q is not accessible, and a direct solution is to sample the negative examples $\mathbf{x}'^-_j$ uniformly from the marginal distribution p instead. However, as illustrated in the introduction, not all the node representations can provide meaningful information for contrastive model learning. As a consequence, the improvement of model performance will be limited if all the negative examples are equally pushed away from the anchor during training.

C. Adaptive Sampling for Graph CL

To mitigate the problems raised by the negative sampling strategy in the conventional CL, we propose a new sampling distribution to make the learning of contrastive objective benefit from the most informative graph nodes. Since class information is unavailable, the devised distribution mainly depends on node representations. Motivated by [18] and [20], in our proposed AdaS, the negatives can be sampled from the distribution q^-_α, which is defined as

$$\begin{aligned} q^-_\alpha(\mathbf{x}'^-_j) &:= q_\alpha(\mathbf{x}'^-_j | h(\mathbf{x}'_i) \neq h(\mathbf{x}'^-_j)), \quad \text{where} \\ q_\alpha(\mathbf{x}'^-_j) &\propto e^{-(\mathcal{D}(\mathbf{x}'_i,\mathbf{x}'^-_j)-\alpha)^2} \cdot p(\mathbf{x}'^-_j). \end{aligned} \tag{3}$$

Here, $\mathbf{x}'_i$ is the anchor, $\mathcal{D}(\mathbf{x}'_i, \mathbf{x}'^-_j) = f(\mathbf{x}'_i) \cdot f(\mathbf{x}'^-_j)/\sigma$, and α is a hyperparameter. Intuitively, the sampling strategy obtained from (3) can upweight the negatives that are not too close or too far to the anchor by choosing a reasonable value for α. As a result, our proposed method is able to emphasize the importance of the true hard negatives without becoming vulnerable to the false negatives.

To better understand the merits of the distribution q^-_α, Fig. 3 visualizes the differences among three negative sampling strategies, namely, the uniform sampling [6], [15], the hard sampling [18], and the devised AdaS sampling strategies. As revealed by Fig. 3(a), the uniform sampling strategy treats all the negatives equally, and thereby failing to capture the important information for contrastive model learning. In the hard sampling strategy [see Fig. 3(b)], the negatives closest to the anchor are regarded as the most informative examples, which could magnify the importance of the false negatives unexpectedly. Different from these two strategies, our proposed sampling strategy [see Fig. 3(c)] encourages learning from the negatives with moderate distance to the anchor, wherein the hyperparameter α helps determine which are the most informative ones. Note that when $\alpha \geq 1$, our proposed AdaS sampling can be considered as an approximation of hard sampling [18]. Empirically, we find that a relatively large α can often yield strong contrastive representations, which will be discussed in the experiments.

D. Polarization of Sampling Weights

As introduced in Section III-C, the contrastive model could be encouraged to learn from the informative negatives with the negative sampling distribution q^-_α, while there still exist some potential flaws in this strategy. Specifically, as the false negatives are often semantically similar to the anchor, their

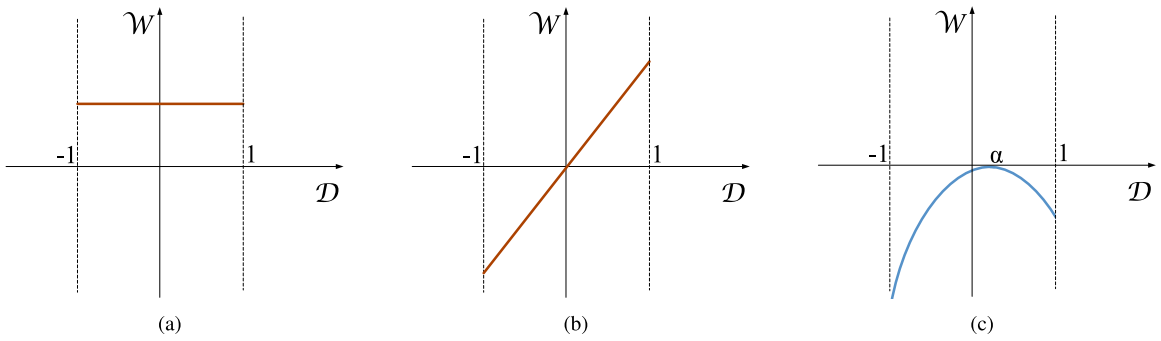


Fig. 3. Visual illustration of different sampling strategies, where $\mathcal{W}$ denotes the importance weights of negative examples, and $\mathcal{D}$ indicates the pairwise cosine similarity between the anchor and a negative example. (a) Uniform sampling, (b) hard sampling, and (c) AdaS sampling.

corresponding importance weights should be set to small values in CL. However, the weights assigned to the false negatives could occasionally be even larger than those of the true negatives, which is revealed in Fig. 4(a). Assume that the pairwise cosine similarity covers the range of $[0, 1]$. In Fig. 4(a), we find that when choosing a relatively large value for α (larger than 0), the false negatives tend to be more dominant than some true negatives [i.e., the suppressed true negatives in Fig. 4(a)] in graph CL. Consequently, directly adopting the sampling distribution q_α^- might bring obstacles to estimating the importance of different negative examples, and thus could be harmful to contrastive representation learning.

Beyond the adaptive sampling strategy mentioned above, we would like to upweight the true negatives, especially the suppressed ones in Fig. 4(a), and decrease the importance of the false negatives in contrastive objective, simultaneously. Following this criterion, we propose an auxiliary polarization regularizer to impose a constraint on the pairwise similarity, which can help adequately distinguish different negative examples. Concretely, the polarization regularizer of the anchor node $\mathbf{x}_i'$ can be given as

$$\mathcal{L}_{\text{pr}}(\mathbf{x}_i') = \log \frac{\sum_{j=1}^{N} e^{\text{ReLU}(\alpha - f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma)}}{\sum_{j=1}^{N} e^{\text{ReLU}(f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma - \alpha)}} \tag{4}$$

where the hyperparameter α is also used in (3) to control the importance of different negative examples. Minimizing the polarization regularizer in (4) can make the pairwise similarity $f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma$ close to α if $f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma < \alpha$ and make $f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma$ as large as possible if $f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma > \alpha$, simultaneously. Coupling the sampling distribution (3) with this polarization regularizer, the impact of the negatives that are too close to the anchor, which could be false negatives, will be weakened; meanwhile, the true negatives can be more dominant in graph CL. Here, choosing a reasonable value for α is critical for accurate example reweighting. For example, the informative negatives will not guide the contrastive model learning as expected when α gets too small, which will be analyzed in experiments. Fig. 4(b) visualizes the function of (4) for better understanding. Ideally, the importance weights of the true negatives can be larger than those of the false negatives by an explicit margin, which is denoted by the gray shading in Fig. 4(b).

E. Optimization of Ultimate Objective

Without loss of generality, for most existing graph CL models, our AdaS framework can build the following contrastive objective based on (2) and (4):

$$\mathcal{L}_{\text{con}} = \frac{1}{2n} \sum_{i=1}^{2n} \left[\tilde{\mathcal{L}}_{\text{pw}}\left(\mathbf{x}_i', \mathbf{x}_i'^+\right) + \lambda \mathcal{L}_{\text{pr}}\left(\mathbf{x}_i'\right) \right] \tag{5}$$

where λ is the weight assigned to $\mathcal{L}_{\text{pr}}$. In $\tilde{\mathcal{L}}_{\text{pw}}(\mathbf{x}_i', \mathbf{x}_i'^+)$, it is not clear how to sample efficiently from the distribution q_α^-. Inspired by PU learning [20], we have

$$q_\alpha^-\left(\mathbf{x}_j'^-\right) = \left(q_\alpha\left(\mathbf{x}_j'^-\right) - \tau^+ q_\alpha^+\left(\mathbf{x}_j'^-\right)\right)/\tau^- \tag{6}$$

with $q_\alpha^+(\mathbf{x}_j'^-) := q_\alpha(\mathbf{x}_j'^- | h(\mathbf{x}_i') = h(\mathbf{x}_j'^-))$. Here, $\tau^- = 1 - \tau^+$, where τ^+ is the class prior and can be estimated from data or treated as a hyperparameter [53]. Afterward, with reference to the importance sampling approach in [18], we can rewrite $\tilde{\mathcal{L}}_{\text{pw}}(\mathbf{x}_i', \mathbf{x}_i'^+)$ in (2) to

$$\mathbb{E}_{\substack{\mathbf{x}_j'^- \sim p, \\ \mathbf{x}_i'^+ \sim p_{\mathbf{x}_i'}^+}} \left[-\log \frac{e^{f(\mathbf{x}_i')^\top f(\mathbf{x}_i'^+)/\sigma}}{e^{f(\mathbf{x}_i')^\top f(\mathbf{x}_i'^+)/\sigma} + Q \mathbb{E}_{\mathbf{x}_j'^- \sim q}[e^{f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma}]} \right] \tag{7}$$

by fixing Q and taking the limit $N \to \infty$. Then $Q\mathbb{E}_{\mathbf{x}_j'^- \sim q}[e^{f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma}]$ equals to

$$\frac{Q}{\tau^-}\left(\mathbb{E}_{\mathbf{x}_j'^- \sim q_\alpha}\left[e^{f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma}\right] - \tau^+ \mathbb{E}_{\mathbf{v} \sim q_\alpha^+}\left[e^{f(\mathbf{x}_i')^\top f(\mathbf{v})/\sigma}\right]\right) \tag{8}$$

by letting $q = q_\alpha^-$. As a result, we only need to approximate the expectations $\mathbb{E}_{\mathbf{x}_j'^- \sim q_\alpha}[e^{f(\mathbf{x}_i')^\top f(\mathbf{x}_j'^-)/\sigma}]$ and $\mathbb{E}_{\mathbf{v} \sim q_\alpha^+}[e^{f(\mathbf{x}_i')^\top f(\mathbf{v})/\sigma}]$, which can be accessible with classical Monte Carlo importance sampling techniques according to [18]. Finally, the overall contrastive objective (5) can be optimized with gradient descent. The detailed description of our proposed AdaS framework is provided in Algorithm 1.

F. Discussion

Here, we intend to discuss the connections between our proposed contrastive loss $\mathcal{L}_{\text{con}}$ and MI maximization of node features and the embeddings. MI quantifies the amount of information acquired from one random variable by observing the other random variable.

For simplicity, we assume that $\mathbf{U}$ and $\mathbf{V}$ are two random variables denoting the embeddings obtained from

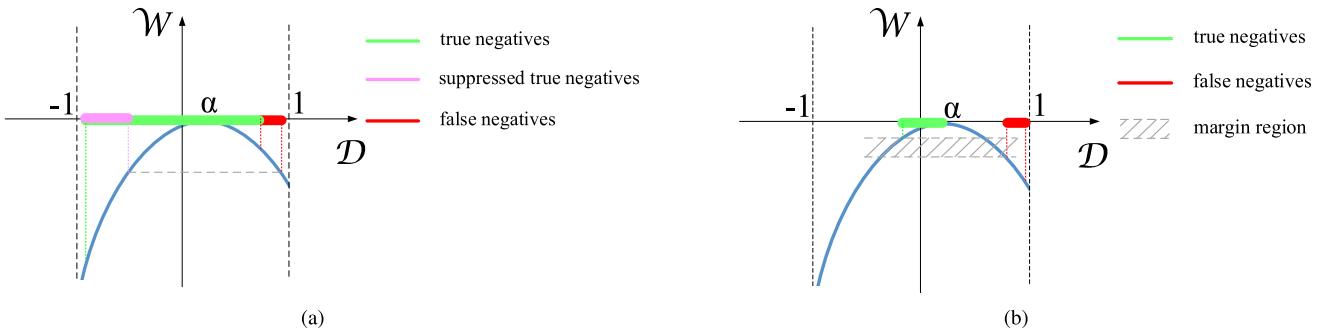


Fig. 4. Visual illustration of the importance weights assigned to different negative examples. (a) Without the polarization regularizer. The green/violet/red lines denote the pairwise cosine similarity between the anchor and the true negatives/suppressed true negatives/false negatives, respectively. (b) With the polarization regularizer. The green/red lines denote the pairwise cosine similarity between the anchor and the true negatives/false negatives, respectively. The gray shading indicates the gap of the importance weights between true and false negatives.

Algorithm 1 Proposed AdaSA Framework

Input: Feature matrix $\mathbf{X}$; adjacency matrix $\mathbf{A}$; maximum number of iterations $\mathcal{T}$; hyperparameters α and λ.

Output: Node embeddings generated by the GNN encoder f.

Randomly initialize the network parameters of the GNN encoder;
// Training phase
for $t = 1$ to $\mathcal{T}$ **do**
    Obtain node embeddings from two augmented graph views, respectively;
    Compute the overall objective function $\mathcal{L}_{con}$ based on (5);
    Update the network parameters using stochastic gradient descent;
end for
// Inference phase
Calculate the node embeddings for all the graph nodes with the trained GNN encoder;

two graph views. Meanwhile, we also define $\tilde{\mathcal{L}}_{\text{pw}} = (1/2n)\sum_{i=1}^{2n}\tilde{\mathcal{L}}_{\text{pw}}(\mathbf{x}'_i, \mathbf{x}'^{+}_i)$ and $\mathcal{L}_{\text{pr}} = (1/2n)\sum_{i=1}^{2n}\mathcal{L}_{\text{pr}}(\mathbf{x}'_i)$. Then the InfoNCE objective can be defined as [54]

$$I_{\text{NCE}}(\mathbf{U};\mathbf{V}) := E_{\prod_i \tilde{p}(\mathbf{u}_i,\mathbf{v}_i)}\left[\frac{1}{n}\sum_{i=1}^{n}\log\frac{e^{\mathbf{u}_i{}^{\top}\mathbf{v}_i}}{\frac{1}{n}\sum_{j=1}^{n}e^{\mathbf{u}_i{}^{\top}\mathbf{v}_j}}\right] \quad (9)$$

where $\mathbf{u}_i$ and $\mathbf{v}_i$ represent the embeddings of the ith node in $\mathbf{U}$ and $\mathbf{V}$, respectively, $\tilde{p}(\mathbf{u}_i, \mathbf{v}_i)$ is the joint distribution of $\mathbf{u}_i$ and $\mathbf{v}_i$. Similar to [15] and [16], we have $-2\tilde{\mathcal{L}}_{\text{pw}} \leq I_{\text{NCE}}(\mathbf{U},\mathbf{V}) + I_{\text{NCE}}(\mathbf{V},\mathbf{U})$. In addition, the InfoNCE estimator has been proven to be a lower bound of the true MI, i.e., $I_{\text{NCE}}(\mathbf{U};\mathbf{V}) \leq I(\mathbf{U};\mathbf{V})$. Hence, we can derive that

$$-\tilde{\mathcal{L}}_{\text{pw}} \leq I(\mathbf{U};\mathbf{V}). \quad (10)$$

According to the data processing inequality, we have $I(\mathbf{U};\mathbf{V}) \leq I(\mathbf{U};\mathbf{X})$ and $I(\mathbf{X};\mathbf{U}) \leq I(\mathbf{X};\mathbf{U},\mathbf{V})$. On this basis, we arrive at the inequality

$$I(\mathbf{U};\mathbf{V}) \leq I(\mathbf{X};\mathbf{U},\mathbf{V}). \quad (11)$$

Following (10) and (11), we can obtain the inequality:

$$-\mathcal{L}_{\text{con}} = -\tilde{\mathcal{L}}_{\text{pw}} - \lambda\mathcal{L}_{\text{pr}} \leq I(\mathbf{X};\mathbf{U},\mathbf{V}) \quad (12)$$

since $\lambda\mathcal{L}_{\text{pr}} > 0$. This demonstrates that the objective $-\mathcal{L}_{\text{con}}$ to be maximized is a lower bound of the MI between the input feature $\mathbf{X}$ and the embeddings in two graph views. That is, minimizing the ultimate loss $\mathcal{L}_{\text{con}}$ is equivalent to explicitly maximizing a lower bound of the MI, which is able to power unsupervised representation learning [54], [55], [56].

TABLE I
DATASET STATISTICS

Dataset	# Nodes	# Edges	# Features	# Classes
Cora	2,708	5,429	1,433	7
CiteSeer	3,327	4,732	3,703	6
PubMed	19,717	44,338	500	3
Amazon Computers	13,752	245,861	767	10
Amazon Photo	7,650	119,081	745	8
Coauthor CS	18,333	81,894	6,805	15

IV. EXPERIMENTS

In this section, we conduct extensive experiments to reveal the effectiveness of our proposed AdaS framework. First, we introduce the datasets used for the validation and the experimental settings. Then the results including the comparison of performance, ablation study, and parametric sensitivity are demonstrated.

A. Datasets

We evaluate the proposed AdaS framework on six widely used benchmark datasets for node classification, including three widely used citation networks (i.e., Cora, CiteSeer, and PubMed) [57], [58], two Amazon product co-purchase networks (i.e., Amazon Computers and Amazon Photo) [59], and one coauthor network subjected to computer science (i.e., Coauthor CS) [59]. The datasets used here are collected from real-world networks from different fields, and their statistics have been demonstrated in Table I.

1) *Citation Networks:* Cora, CiteSeer, and PubMed are three publicly available datasets composed of scientific publications. In these networks, published papers are denoted by the graph nodes and the citation relationships between papers are represented by the graph edges. Each node is associated with a bag-of-words feature vector and a ground-truth label.

TABLE II

CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE CORA, CITESEER, PUBMED, AMAZON COMPUTERS, AMAZON PHOTO, AND COAUTHOR CS DATASETS. IN THE SECOND COLUMN, WE HIGHLIGHT THE KIND OF DATA AVAILABLE TO EACH METHOD DURING TRAINING (**X**: FEATURES, **A**: ADJACENCY MATRIX, **Y**: LABELS). ADAS-GR, ADAS-MV, ADAS-ER, AND ADAS-FD INDICATE THE MODEL VARIANTS THAT INCORPORATE THE PROPOSED SAMPLING STRATEGY TO GRACE, MVGRL, CONEDGEREM, AND CONFEADROP, RESPECTIVELY

Method	Training data	Cora	CiteSeer	PubMed	Amazon Computers	Amazon Photo	Coauthor CS
GCN [27]	X, A, Y	79.56±1.52	69.02±0.91	77.54±2.30	74.41±4.65	82.72±2.83	89.10±2.22
GAT [35]	X, A, Y	81.90±2.00	69.74±0.77	77.48±2.45	71.58±1.09	81.46±2.05	89.15±1.08
DGI [9]	X, A	80.50±1.86	71.79±0.69	77.10±2.35	78.44±1.23	89.03±0.85	89.87±0.79
GMI [61]	X, A	78.96±1.16	66.80±2.46	77.35±2.77	77.54±2.58	89.37±1.16	88.83±1.53
GRACE [15]	X, A	80.31±1.32	69.35±1.87	78.67±2.31	78.02±1.16	84.46±1.18	89.37±1.09
AdaS-GR	X, A	**83.51±1.18**	72.19±0.86	79.86±0.94	78.23±1.20	**90.63±1.13**	90.35±0.82
MVGRL [48]	X, A	81.82±1.09	70.90±0.54	76.70±3.38	77.19±1.60	85.35±1.79	90.11±0.44
AdaS-MV	X, A	82.81±0.86	**73.16±0.78**	**80.47±1.94**	**79.10±1.65**	90.04±1.47	**91.63±0.57**
ConEdgeRem [60]	X, A	78.09±1.65	69.53±1.48	78.94±1.44	73.99±1.20	84.66±0.84	88.30±1.15
AdaS-ER	X, A	81.07±1.60	72.07±0.86	80.11±1.51	79.02±1.82	88.97±1.16	89.43±0.73
ConFeaDrop [60]	X, A	78.34±1.43	68.92±0.59	76.15±1.38	74.45±0.96	86.10±0.15	89.12±0.57
AdaS-FD	X, A	81.33±1.55	72.10±1.03	77.24±1.30	78.32±1.12	89.60±0.26	89.94±0.12

2) *Co-Purchase Networks:* Amazon Computers and Amazon Photo are two networks of co-purchase relationships constructed from Amazon, where the nodes represent goods and the links indicate that two goods are frequently bought together. The node features are bag-of-words encoded product reviews, and class labels correspond to the product categories.
3) *Coauthor Network:* Coauthor CS is a coauthorship graph based on the Microsoft Academic Graph from the KDD Cup 2016 challenge, where the nodes are authors and are connected by an edge if they coauthor a paper. The node features represent keywords for each author's papers and the class labels denote the most active fields of study for each author.

B. Experimental Settings

Here, we will introduce the settings of our experiments, including the baseline methods used for comparison and the evaluation protocol. As our AdaS aims to improve the performance of graph CL models, we apply it as a modification to the state-of-the-art contrastive methods. To be specific, the recently proposed GRACE [15] and MVGRL [48] have been adopted as baseline methods, and our proposed AdaS is implemented under these two graph CL frameworks. Meanwhile, to further demonstrate the generalization ability of our proposed method under different augmentation techniques, we apply AdaS to graph CL frameworks with edge removing (ConEdgeRem) and feature dropout (ConFeaDrop) [60], respectively. Besides, to compare our framework with supervised counterparts, the two representative models GCN [27] and GAT [35] are adopted here and are trained in an end-to-end fashion. Finally, two additional state-of-the-art graph CL methods, including DGI [9] and GMI [61], are also used for comparison here.

In the experiments, we follow the linear evaluation scheme introduced by [9]. Specifically, the graph encoder is first trained in an unsupervised manner, and then the node embeddings generated by the trained encoder, together with the given node labels, are used to train and test a classifier, such as the logistic regression classifier. Following [59], in all the cases, we use 20 labeled nodes per class as the training set, 30 nodes per class as the validation set, and the rest as the test set. All the experiments have been repeated for ten times, and we report the average performance on each dataset for fair evaluation.

C. Performance of Node Classification

The empirical results of node-level representation learning obtained by different methods are presented in Table II, where the highest performance of all the methods is highlighted in boldface on each dataset. Meanwhile, the visual comparison between the graph CL methods (i.e., GRACE, MVGRL, ConEdgeRem, and ConFeaDrop) and their corresponding model variants are exhibited in Fig. 5. From Table II, we clearly observe that our proposed AdaS could improve the performance of four graph CL methods (i.e., GRACE, MVGRL, ConEdgeRem, and ConFeaDrop) on all the six datasets, which reveals the good compatibility of AdaS with different types of graph CL models. We infer that this strong performance benefits from the adaptive sampling strategy for the negative node pairs. In particular, the improvements of about 5% can be observed over GRACE and MVGRL on the Amazon Photo dataset, and improvements of about 3% can be found over GRACE, ConEdgeRem, and ConFeaDrop, respectively, on the Cora dataset. These statistical results demonstrate the power of our AdaS framework in boosting the representation ability of graph CL models.

Comparing the performance of supervised and unsupervised baseline methods, we find that GAT can achieve better classification results than the contrastive models without using the proposed adaptive sampling strategy on the Cora dataset. We consider this can be attributed to the naive selection of negative node pairs, which might prevent these contrastive models from learning sufficient useful information during training. In contrast to most current graph CL models, the proposed AdaS-GR and AdaS-MV encourage learning from the informative

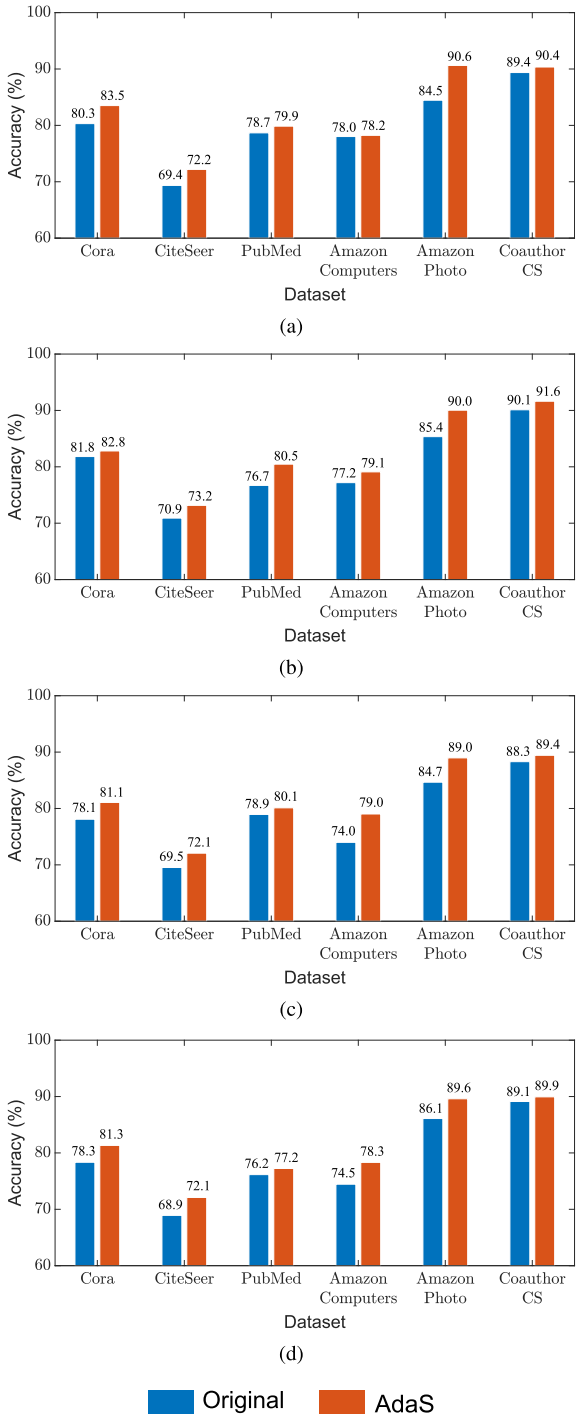


Fig. 5. Comparison of classification accuracies (%) between the graph CL models and their variants using AdaS. (a) GRACE, (b) MVGRL, (c) ConEdgeRem, and (d) ConFeaDrop.

negative examples, which can help the encoders learn strong representations. As a consequence, they can outperform GAT even without using label information. Similar observations can be found in other datasets, such as CiteSeer and PubMed. It is also notable that the performance of ConEdgeRem and ConFeaDrop seems relatively poor on the Amazon Computers dataset compared with GRACE, which could be due to that using edge removing or feature dropout alone is not able to produce sufficient meaningful contrastive pairs. Nevertheless, our AdaS can mitigate the undesirable effect caused by graph augmentation via appropriately adjusting the importance weights of different pairs.

TABLE III

CLASSIFICATION ACCURACIES (%) OBTAINED BY APPLYING DIFFERENT NEGATIVE SAMPLING STRATEGIES TO GRACE ON THE CORA, CITESEER, AND PUBMED DATASETS. "GR" IS SHORT FOR GRACE

Method	Cora	CiteSeer	PubMed
GRACE [15]	80.31±1.32	69.35±1.87	78.67±2.31
HBNM [18] + GR [15]	83.02±1.18	70.72±1.00	79.30±1.78
DCL [20] + GR [15]	81.33±1.08	70.65±0.97	79.50±1.03
AdaS + GR [15]	**83.51±1.18**	**72.19±0.86**	**79.86±0.94**

TABLE IV

CLASSIFICATION ACCURACIES (%) OBTAINED BY APPLYING DIFFERENT NEGATIVE SAMPLING STRATEGIES TO MVGRL ON THE CORA, CITESEER, AND PUBMED DATASETS. "MV" IS SHORT FOR MVGRL

Method	Cora	CiteSeer	PubMed
MVGRL [48]	81.82±1.09	70.90±0.54	76.70±3.38
HBNM [18] + MV [48]	82.12±1.08	71.19±0.99	79.44±1.69
DCL [20] + MV [48]	82.29±1.06	70.15±2.03	77.97±3.38
AdaS + MV [48]	**82.81±0.86**	**73.16±0.78**	**80.47±1.94**

TABLE V

CLASSIFICATION ACCURACIES (%) OBTAINED BY APPLYING DIFFERENT NEGATIVE SAMPLING STRATEGIES TO CONEDGEREM ON THE CORA, CITESEER, AND PUBMED DATASETS. "CER" IS SHORT FOR CONEDGEREM

Method	Cora	CiteSeer	PubMed
ConEdgeRem [60]	78.09±1.65	69.53±1.48	78.94±1.44
HBNM [18] + CER [60]	79.13±2.43	70.61±0.69	79.58±1.16
DCL [20] + CER [60]	78.26±1.59	70.27±0.90	79.41±1.18
AdaS + CER [60]	**81.07±1.60**	**72.07±0.86**	**80.11±1.51**

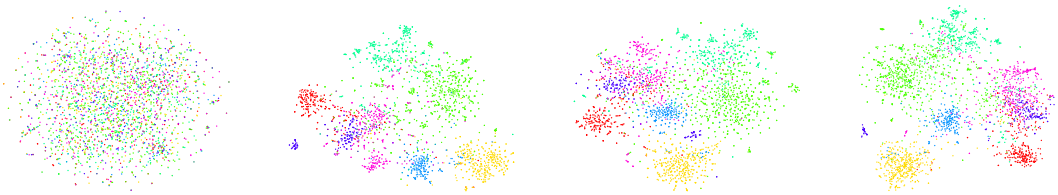
TABLE VI

CLASSIFICATION ACCURACIES (%) OBTAINED BY APPLYING DIFFERENT NEGATIVE SAMPLING STRATEGIES TO CONFEADROP ON THE CORA, CITESEER, AND PUBMED DATASETS. "CFD" IS SHORT FOR CONFEADROP

Method	Cora	CiteSeer	PubMed
ConFeaDrop [60]	78.34±1.43	68.92±0.59	76.15±1.38
HBNM [18] + CFD [60]	79.65±1.73	70.96±0.77	76.97±1.45
DCL [20] + CFD [60]	79.41±1.18	70.79±1.00	76.50±1.90
AdaS + CFD [60]	**81.33±1.55**	**72.10±1.03**	**77.24±1.30**

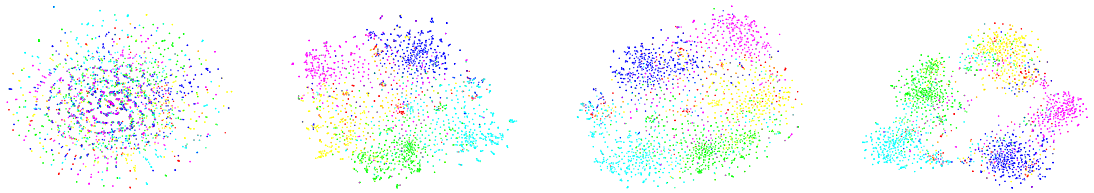
D. Comparison of Different Sampling Strategies

To demonstrate the effectiveness of the proposed adaptive sampling strategy, we use two additional sampling strategies, namely, HBNM [18] and DCL [20], as baseline methods. Here, HBNM proposes to upweight the hard negatives and DCL develops debiasing terms to select true negatives in contrastive model learning. We implement different sampling strategies on four graph CL frameworks, i.e., GRACE [15], MVGRL [48], ConEdgeRem [60], and ConFeaDrop [60], and the results are exhibited in Tables III–VI, respectively. We observe that both HBNM and our AdaS can enhance the performance of these



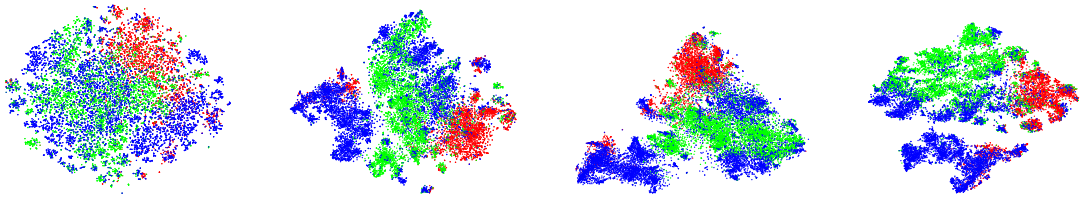
(a) (b) (c) (d)

Fig. 6. Visualization of t-SNE embeddings from (a) raw features, (b) MVGRL, (c) AdaS-MV (w/o PoR), and (d) AdaS-MV, on the Cora dataset.



(a) (b) (c) (d)

Fig. 7. Visualization of t-SNE embeddings from (a) raw features, (b) MVGRL, (c) AdaS-MV (w/o PoR), and (d) AdaS-MV, on the CiteSeer dataset.



(a) (b) (c) (d)

Fig. 8. Visualization of t-SNE embeddings from (a) raw features, (b) MVGRL, (c) AdaS-MV (w/o PoR), and (d) AdaS-MV, on the PubMed dataset.

contrastive models, indicating the benefits of hard negative sampling to contrastive model learning. Meanwhile, it is worth noting that our proposed AdaS consistently outperforms HBNM on all the three datasets, possibly due to the adaptive sampling strategy that is able to suppress the adverse impacts of false negatives. Although DCL attempts to address the false negative problem, there remains performance gap between DCL and our AdaS, revealing the superiority of AdaS in correcting the sampling error of false negatives.

E. Ablation Study

As described in the introduction, our AdaS framework ameliorates the conventional graph contrastive objective by adaptively reweighting the negative examples, where an auxiliary polarization regularizer is also used to enhance the discrimination ability of AdaS. To shed light on the contributions of these two components, we report the classification results of the AdaS models when each of the two components is removed on three widely used datasets, namely, Cora, CiteSeer, and PubMed. The data splits are kept identical to those in Section IV-C. For simplicity, we adopt "AdaS-GR (w/o PoR)" and "AdaS-MV (w/o PoR)" to represent the reduced model variants by removing the polarization regularizer. The comparative results have been exhibited in Tables VII and VIII. It is apparent that our AdaS is an effective framework to improve the performance of graph CL models. Meanwhile, improvements over the original graph CL models (i.e., GRACE and MVGRL) can still be achieved by only using the adaptive sampling strategy, which verifies the effectiveness of this module. Besides, the performance margin between our AdaS and its reduced models without the polarization

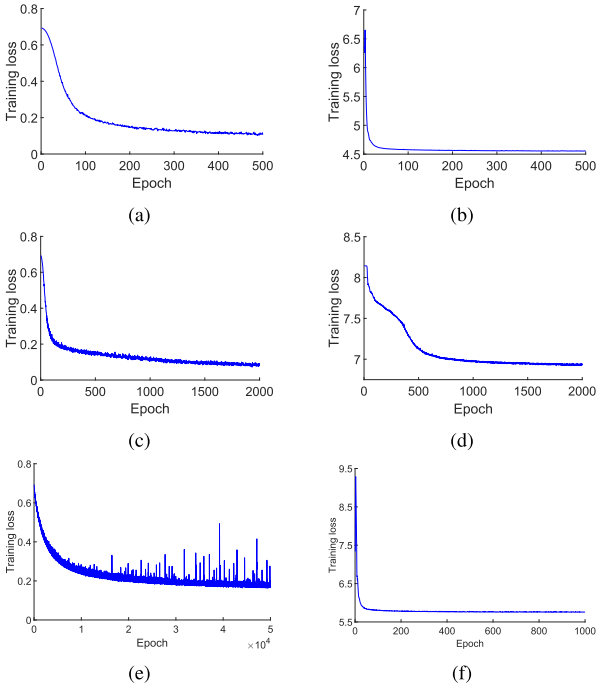


Fig. 9. Convergence analysis of MVRGL on (a) Cora, (c) CiteSeer, and (e) PubMed datasets, and AdaS-MV on (b) Cora, (d) CiteSeer, and (f) PubMed datasets, respectively.

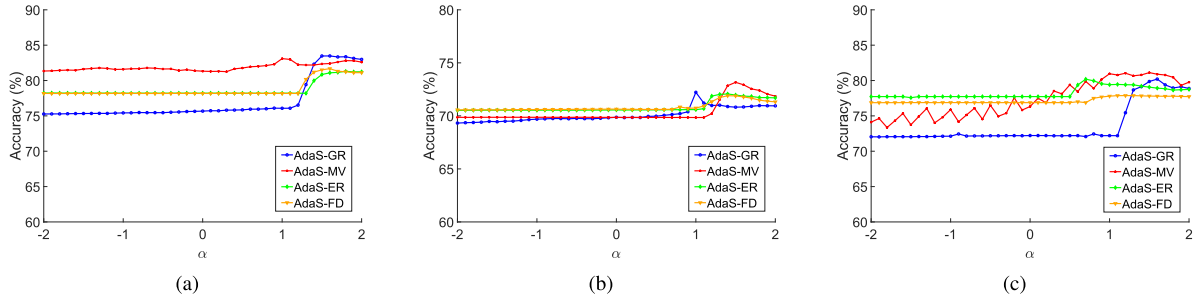


Fig. 10. Sensitivity analysis of α in different model variants. (a) Cora dataset, (b) CiteSeer dataset, and (c) PubMed dataset.

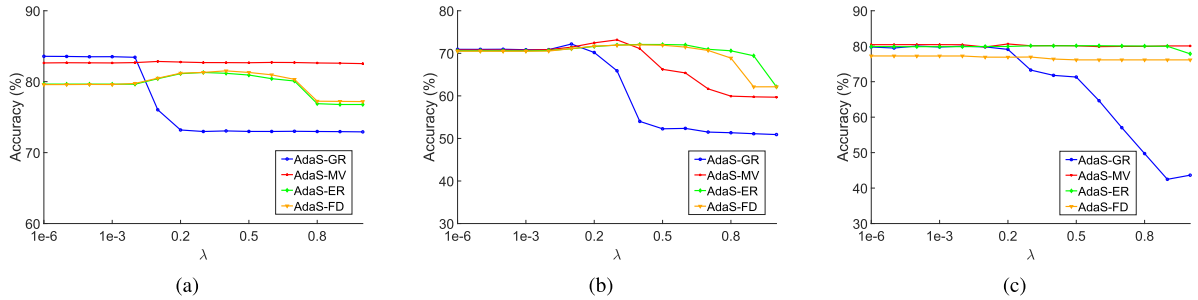


Fig. 11. Sensitivity analysis of λ in different model variants. (a) Cora dataset, (b) CiteSeer dataset, and (c) PubMed dataset.

TABLE VII
ABLATION STUDY OF THE ADAPTIVE SAMPLING MODULE AND POLARIZATION REGULARIZER FOR ADAS-GR METHOD

Method	Cora	CiteSeer	PubMed
GRACE	80.31±1.32	69.35±1.87	78.67±2.31
AdaS-GR (w/o PoR)	82.79±1.01	71.12±1.39	79.35±1.27
AdaS-GR	**83.51±1.18**	**72.19±0.86**	**79.86±0.94**

TABLE VIII
ABLATION STUDY OF THE ADAPTIVE SAMPLING MODULE AND POLARIZATION REGULARIZER FOR ADAS-MV METHOD

Method	Cora	CiteSeer	PubMed
MVGRL	81.82±1.09	70.90±0.54	76.70±3.38
AdaS-MV (w/o PoR)	82.05±1.09	71.23±1.71	76.78±2.91
AdaS-MV	**82.81±0.86**	**73.16±0.78**	**80.47±1.94**

regularizer demonstrates that the polarization regularizer is able to further enhance the discrimination ability of our proposed sampling strategy. To illustrate the critical roles of these two components intuitively, Figs. 6–8 display t-SNE [62] plots of the node embeddings on the Cora, CiteSeer, and PubMed datasets, respectively. We can observe that the proposed AdaS can obtain more distinguishable clusters. In particular, on the Cora dataset, it is worth noting that the embeddings generated by AdaS-MV exhibit more discernible clusters than those generated by MVGRL and AdaS-MV (w/o PoR).

Here, we also present the convergence curves of AdaS-MV and MVGRL throughout training in Fig. 9. Observe the curves, it is apparent that AdaS-MV takes fewer iterations to converge on the Cora and PubMed datasets, compared with MVGRL. On the CiteSeer dataset, although the loss curve of MVGRL is fluctuating during training, it can be noted that our AdaS-MV is still able to converge within 1500 iterations. In Fig. 9(e), we find that MVGRL takes more than 10 000 iterations to converge, while our AdaS-MV requires less than 200 iterations for convergence. Moreover, the convergence curve of MVGRL appears jagged fluctuations during training. Comparatively, the proposed AdaS-MV achieves a more stable convergent curve than MVGRL, as shown in Fig. 9(f). Consequently, we speculate that the informative node pairs provide nonnegligible gradient information during contrastive model learning, which can accelerate the convergence of representation learning in the training phase.

F. Parametric Sensitivity

In our proposed AdaS framework, there exist two critical hyperparameters to be pretuned manually, namely, α which is used in the sampling distribution (3) and λ which is the weight assigned to $\mathcal{L}_{\text{pr}}$ in (5). Therefore, in this section, we will evaluate in detail the sensitivity of the performance to different hyperparameter settings. To be specific, we examine the test accuracy of AdaS-GR, AdaS-MV, AdaS-ER, and AdaS-FD by varying α or λ, and meanwhile fixing the other hyperparameter to a constant value.

The results on the Cora, CiteSeer, and PubMed datasets are shown in Figs. 10 and 11, respectively. By setting the temperature parameter σ to 0.5 throughout the experiment, the dot product between normalized node embeddings is mapped to the range of $[-2, 2]$. Hence, we vary the value of α from -2 to 2 with an interval of 0.1 to comprehensively reveal the role of our proposed sampling distribution. From Fig. 10, we observe that the classification accuracy is relatively low when α is small. It can be inferred that adopting the sampling distribution with a small value of α can restrict the capacity

of AdaS to learn from the informative negatives. Similarly, when α gets too large, the sampling error [20] raised by false negatives may deteriorate the representations and thus limiting the performance of downstream tasks. Interestingly, we find that choosing a relatively large value for α often leads to high accuracy on all the three datasets. This is mainly due to that the sampling error can be eliminated to some extent as illustrated in Section III, and meanwhile the model is encouraged to learn from the informative negatives.

The results of λ are shown in Fig. 11. We note that the performance is relatively stable when λ is not too large on the three datasets, which demonstrates that our AdaS framework is insensitive to λ. However, when λ gets large, the accuracy could seriously decline. For example, when $\lambda > 0.8$ on the CiteSeer dataset, the classification accuracy of AdaS-GR can be even lower than 50%. We speculate that overemphasizing the role of the polarization regularizer may eliminate the meaningful information in negative pairs, thereby making the sampling strategy invalid for graph CL.

V. Conclusion

In this article, we propose a new graph CL framework termed "AdaS" via using an adaptive sampling strategy. The advantages of AdaS are threefold.

1) *Adaptability:* By designing a novel negative sampling distribution, constrained with an auxiliary polarization regularizer, our AdaS framework is able to adaptively learn from the most informative negative examples, offering better utilization of the embedding space than the conventional graph CL objectives.
2) *Stability:* The stability of contrastive model training can be enhanced by incorporating our AdaS, as the informative negative examples could provide meaningful gradient information for model learning.
3) *Generality:* The proposed AdaS framework can accommodate to different types of graph CL models, such as GRACE and MVGRL. This is due to that AdaS mainly focuses on optimizing the sampling strategy of negative examples without changing the typical contrastive objective.

Based on the above merits, our AdaS has revealed superior performance to various state-of-the-art contrastive GNN models on multiple real-world datasets. Nevertheless, the potential risk of false positives has not been investigated in this article, which could deteriorate the model performance. In addition, the hyperparameter α should be manually tuned in practical use.

Our future work will focus on extending the proposed AdaS to semi-supervised learning. To be specific, in semi-supervised learning, the scarce but valuable label information can "teach" the model to identify the false negatives. As a result, there is no need to tune the hyperparameter relating to our sampling strategy manually, by which the efficiency and flexibility could be improved.

References

[1] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1263–1272.

[2] H. Huang, Y. Song, Y. Wu, J. Shi, X. Xie, and H. Jin, "Multitask representation learning with multiview graph convolutional networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 3, pp. 983–995, Mar. 2022.

[3] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.

[4] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 494–514, Feb. 2022.

[5] S. Wan, S. Pan, J. Yang, and C. Gong, "Contrastive and generative graph convolutional networks for graph-based semi-supervised learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 11, 2021, pp. 10049–10057.

[6] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1597–1607.

[7] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "4S-DT: Self-supervised super sample decomposition for transfer learning with application to COVID-19 detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 7, pp. 2798–2808, Jul. 2021.

[8] Y. Liu, Z. Li, S. Pan, C. Gong, C. Zhou, and G. Karypis, "Anomaly detection on attributed networks via contrastive self-supervised learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2378–2392, Jun. 2022.

[9] P. Veličković, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, "Deep graph infomax," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018.

[10] T. Wang and P. Isola, "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 9929–9939.

[11] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9726–9735.

[12] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen, "Graph contrastive learning with augmentations," in *Proc. NIPS*, vol. 33, 2020, pp. 5812–5823.

[13] M. Wu, C. Zhuang, M. Mosse, D. Yamins, and N. Goodman, "On mutual information in contrastive learning for visual representations," 2020, *arXiv:2005.13149*.

[14] T. Xiao, X. Wang, A. A. Efros, and T. Darrell, "What should not be contrastive in contrastive learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.

[15] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, "Deep graph contrastive representation learning," in *Proc. ICML Workshop Graph Represent. Learn. Beyond*, 2020.

[16] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, "Graph contrastive learning with adaptive augmentation," in *Proc. Web Conf.*, Apr. 2021, pp. 2069–2080.

[17] S. Wan, Y. Zhan, L. Liu, B. Yu, S. Pan, and C. Gong, "Contrastive graph Poisson networks: Semi-supervised learning with extremely limited labels," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021.

[18] J. D. Robinson, C.-Y. Chuang, S. Sra, and S. Jegelka, "Contrastive learning with hard negative samples," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.

[19] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

[20] C.-Y. Chuang, J. Robinson, Y.-C. Lin, A. Torralba, and S. Jegelka, "Debiased contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020.

[21] C. Huang, C. C. Loy, and X. Tang, "Local similarity-aware deep feature embedding," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 29, 2016.

[22] X. Zhang, C. Xu, X. Tian, and D. Tao, "Graph edge convolutional neural networks for skeleton-based action recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 8, pp. 3047–3060, Aug. 2020.

[23] W. Lu et al., "SkipNode: On alleviating performance degradation for deep graph convolutional networks," 2021, *arXiv:2112.11628*.

[24] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proc. IJCNN*, vol. 2, 2005, pp. 729–734.

[25] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2013.

[26] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Conf. Adv. Neural Inf. Process. Syst. (NIPS)*, 2016, pp. 3844–3852.

[27] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017.

[28] W. Liu et al., "Item relationship graph neural networks for E-commerce," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 9, pp. 4785–4799, Sep. 2022.

[29] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multiscale dynamic graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3162–3177, May 2020.

[30] H. Cai, S. Lv, G. Lu, and T. Li, "Graph convolutional networks for fast text classification," in *Proc. 4th Int. Conf. Natural Lang. Process. (ICNLP)*, Mar. 2022, pp. 420–425.

[31] J. Chen, T. Ma, and C. Xiao, "FastGCN: Fast learning with graph convolutional networks via importance sampling," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018.

[32] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, "Simplifying graph convolutional networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 6861–6871.

[33] B. X. B. Yu, Y. Liu, K. C. C. Chan, Q. Yang, and X. Wang, "Skeleton-based human action evaluation using graph convolutional network for monitoring Alzheimer's progression," *Pattern Recognit.*, vol. 119, Nov. 2021, Art. no. 108095.

[34] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Conf. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 1024–1034.

[35] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018.

[36] W.-L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, and C.-J. Hsieh, "Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 257–266.

[37] S. Gu, X. Wang, C. Shi, and D. Xiao, "Self-supervised graph neural networks for multi-behavior recommendation," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 1–7.

[38] L. Yang and S. Hong, "Omni-granular ego-semantic propagation for self-supervised graph representation learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2022, pp. 25022–25037.

[39] R. D. Hjelm et al., "Learning deep representations by mutual information estimation and maximization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018.

[40] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[41] J.-B. Grill et al., "Bootstrap your own latent: A new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020.

[42] X. Chen and K. He, "Exploring simple Siamese representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15745–15753.

[43] S. Chen, G. Niu, C. Gong, J. Li, J. Yang, and M. Sugiyama, "Large-margin contrastive learning with distance polarization regularizer," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2021, pp. 1673–1683.

[44] J. Li, P. Zhou, C. Xiong, and S. Hoi, "Prototypical contrastive learning of unsupervised representations," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.

[45] X. Chen et al., "Self-PU: Self boosted and calibrated positive-unlabeled training," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1510–1519.

[46] S. Li, X. Wang, A. Zhang, Y. Wu, X. He, and T.-S. Chua, "Let invariant rationale discovery inspire graph contrastive learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2022, pp. 13052–13065.

[47] N. Liu, X. Wang, D. Bo, C. Shi, and J. Pei, "Revisiting graph contrastive learning from the perspective of graph spectrum," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2022.

[48] K. Hassani and A. H. Khasahmadi, "Contrastive multi-view representation learning on graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 4116–4126.

[49] J. Klicpera, S. Weißenberger, and S. Günnemann, "Diffusion improves graph learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 13354–13366.

[50] Y. You, T. Chen, Y. Shen, and Z. Wang, "Graph contrastive learning automated," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2021.

[51] S. Suresh, P. Li, C. Hao, and J. Neville, "Adversarial graph augmentation to improve graph contrastive learning," 2021, *arXiv:2106.05819*.

[52] S. Arora, H. Khandeparkar, M. Khodak, O. Plevrakis, and N. Saunshi, "A theoretical analysis of contrastive unsupervised representation learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 5628–5637.

[53] S. Jain, M. White, and P. Radivojac, "Estimating the class prior and posterior from noisy positives and unlabeled data," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 29, 2016, pp. 2693–2701.

[54] B. Poole, S. Ozair, A. Van Den Oord, A. Alemi, and G. Tucker, "On variational bounds of mutual information," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 5171–5180.

[55] A. Alemi, B. Poole, I. Fischer, J. Dillon, R. A. Saurous, and K. Murphy, "Fixing a broken ELBO," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2018, pp. 159–168.

[56] W. Hu, T. Miyato, S. Tokui, E. Matsumoto, and M. Sugiyama, "Learning discrete representations via information maximizing self-augmented training," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1558–1567.

[57] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad, "Collective classification in network data," *AI Mag.*, vol. 29, no. 3, p. 93, Sep. 2008.

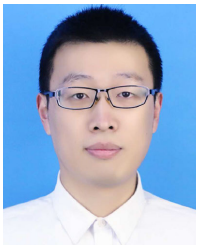
[58] A. Bojchevski and S. Günnemann, "Deep Gaussian embedding of graphs: Unsupervised inductive learning via ranking," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018.

[59] O. Shchur, M. Mumme, A. Bojchevski, and S. Günnemann, "Pitfalls of graph neural network evaluation," 2018, *arXiv:1811.05868*.

[60] Y. Zhu, Y. Xu, Q. Liu, and S. Wu, "An empirical study of graph contrastive learning," 2021, *arXiv:2109.01116*.

[61] Z. Peng et al., "Graph representation learning via graphical mutual information maximization," in *Proc. Web Conf.*, Apr. 2020, pp. 259–270.

[62] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–27, 2008.



Sheng Wan received the Ph.D. degree in computer science and technology from the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2023.

He is currently a Post-Doctoral Researcher with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include graph machine learning, weakly supervised learning, and hyperspectral image processing.

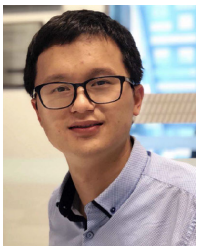
Yibing Zhan (Member, IEEE) received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2012 and 2018, respectively.

From 2018 to 2020, he was an Associate Researcher with the Computer and Software School, Hangzhou Dianzi University, Hangzhou, China. He is currently an Algorithm Scientist at JD Explore Academy, Beijing, China. He has publications on various top conferences and journals, such as Conference on Computer Vision and Pattern Recognition (CVPR), ACM International Conference on Multimedia (ACM MM), Association for the Advancement of Artificial Intelligence (AAAI), *International Journal of Computer Vision* (IJCV), and IEEE TRANSACTIONS ON MULTIMEDIA (T-MM). His research interests include graphical models and multimodal learning, including crossmodal retrieval, scene graph generation, and graph neural networks.

Shuo Chen received the doctoral degree from the Nanjing University of Science and Technology, Nanjing, China, in 2020.

He was a CSC Visiting Student at the University of Pittsburgh, Pittsburgh, PA, USA, from 2018 to 2019. He is currently a Post-Doctoral Researcher at RIKEN Center for Advanced Intelligence Project (RIKEN-AIP), Tokyo, Japan. He has published more than 20 technical papers at top-tier conferences such as International Conference on Machine Learning (ICML), Conference on Neural Information Processing Systems (NeurIPS), Conference on Computer Vision and Pattern Recognition (CVPR), Association for the Advancement of Artificial Intelligence (AAAI), and prominent journals such as IEEE TRANSACTIONS ON IMAGE PROCESSING (T-IP) and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (T-NNLS). His research interests mainly include machine learning and pattern recognition, in particular contrastive learning and metric learning.

Dr. Chen won the "Excellent Achievement Award" of RIKEN, the "Excellent Doctoral Dissertation Award" of Chinese Institute of Electronics (CIE), and the "Excellent Doctoral Dissertation Nomination" of Chinese Association for Artificial Intelligence (CAAI). He has served as the Area Chair (AC) of NeurIPS and ICML, and also the reviewer for more than 20 journals such as *Artificial Intelligence Journal* (AIJ), *Journal of Machine Learning Research* (JMLR), and IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (T-PAMI).



Shirui Pan (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Technology Sydney (UTS), Ultimo, NSW, Australia, in 2015.

He is currently a Full Professor with the School of Information and Communication Technology, Griffith University, Gold Coast, QLD, Australia. He has published more than 100 research papers in top-tier journals and conferences, including the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (T-PAMI), IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (T-KDE), IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (T-NNLS), Conference on Neural Information Processing Systems (NeurIPS), International Conference on Machine Learning (ICML), and ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). His research interests include data mining and machine learning.

Dr. Pan was recognized as one of the AI 2000 Association for the Advancement of Artificial Intelligence (AAAI)/International Joint Conference on Artificial Intelligence (IJCAI) Most Influential Scholars in Australia, in 2021.

Jian Yang (Member, IEEE) received the Ph.D. degree from the Nanjing University of Science and Technology (NJUST), Nanjing, China, in 2002.

From 2006 to 2007, he was a Post-Doctoral Fellow with the Department of Computer Science, New Jersey Institute of Technology, Newark, NJ, USA. He is currently a Professor with the School of Computer Science and Technology, NJUST. He is the author of more than 400 scientific articles in pattern recognition and computer vision. His papers have been cited more than 28 000 times in Google Scholar. His research interests include pattern recognition, computer vision, and machine learning.

Dr. Yang is a fellow of IAPR. He is/was an Associate Editor of *Pattern Recognition* and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (T-NNLS).

Dacheng Tao (Fellow, IEEE) is currently an Advisor and the Chief Scientist of the Digital Science Institute, The University of Sydney, Darlington, NSW, Australia. He mainly applies statistics and mathematics to artificial intelligence and data science, and his research is detailed in one monograph and more than 200 publications in prestigious journals and proceedings at leading conferences.

Dr. Tao is a fellow of the Australian Academy of Science, the American Association for the Advancement of Science (AAAS), and ACM. He received the 2015 Australian Scopus-Eureka Prize, the 2018 IEEE International Conference on Data Mining (ICDM) Research Contributions Award, and the 2021 IEEE Computer Society McCluskey Technical Achievement Award.

Chen Gong (Senior Member, IEEE) received the dual doctoral degree from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2016, and the University of Technology Sydney (UTS), Ultimo, NSW, Australia, in 2017.

Currently, he is a Full Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. He has published more than 100 technical papers at prominent journals and conferences such as *Journal of Machine Learning Research* (JMLR), IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (T-PAMI), IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (T-NNLS), IEEE TRANSACTIONS ON IMAGE PROCESSING (T-IP), IEEE TRANSACTIONS ON CYBERNETICS (T-CYB), IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (T-CSVT), IEEE TRANSACTIONS ON MULTIMEDIA (T-MM), IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (T-ITS), ACM TRANSACTIONS ON INTELLIGENT SYSTEMS AND TECHNOLOGY (T-IST), International Conference on Machine Learning (ICML), Conference on Neural Information Processing Systems (NeurIPS), International Conference on Learning Representations (ICLR), Conference on Computer Vision and Pattern Recognition (CVPR), Association for the Advancement of Artificial Intelligence (AAAI), International Joint Conference on Artificial Intelligence (IJCAI), and IEEE International Conference on Data Mining (ICDM). His research interests mainly include machine learning, data mining, and learning-based vision problems.

Dr. Gong won the "Excellent Doctorial Dissertation Award" of Chinese Association for Artificial Intelligence, "Young Elite Scientists Sponsorship Program" of China Association for Science and Technology, "Wu Wen-Jun AI Excellent Youth Scholar Award," and the Science Fund for Distinguished Young Scholars of Jiangsu Province. He was also selected as the "Global Top Chinese Young Scholars in AI" released by Baidu. He serves as an Associate Editor for IEEE T-CSVT and NePL, and also the Area Chair or Senior PC Member of several top-tier conferences such as AAAI, IJCAI, ACM International Conference on Multimedia (ACM MM), European Conference on Machine Learning (ECML), and International Conference on Artificial Intelligence and Statistics (AISTATS).