Deep Graph Structural Infomax

Wenting Zhao[1], Gongping Xu[1], Zhen Cui[1*], Siqiang Luo[2], Cheng Long[2], Tong Zhang[1]

[1]Nanjing University of Science and Technology, Nanjing, China.
[2]Nanyang Technological University, Singapore.
{wtingzhao, xugongping, zhen.cui, tong.zhang}@njust.edu.cn, {siqiang.luo, c.long}@ntu.edu.sg

Abstract

In the scene of self-supervised graph learning, Mutual Information (MI) was recently introduced for graph encoding to generate robust node embeddings. A successful representative is Deep Graph Infomax (DGI), which essentially operates on the space of node features but ignores topological structures, and just considers global graph summary. In this paper, we present an effective model called Deep Graph Structural Infomax (DGSI) to learn node representation. We explore to derive the structural mutual information from the perspective of Information Bottleneck (IB), which defines a trade-off between the sufficiency and minimality of representation on the condition of the topological structure preservation. Intuitively, the derived constraints formally maximize the structural mutual information both edge-wise and local neighborhood-wise. Besides, we develop a general framework that incorporates the global representational mutual information, local representational mutual information, and sufficient structural information into the node representation. Essentially, our DGSI extends DGI and could capture more fine-grained semantic information as well as beneficial structural information in a self-supervised manner, thereby improving node representation and further boosting the learning performance. Extensive experiments on different types of datasets demonstrate the effectiveness and superiority of the proposed method.

1 Introduction

Graph is a flexible and universal data structure consisting of a set of nodes and edges, where nodes can represent any kind of object and edges indicate some relationships between node pairs. For mining graph data, Graph Neural Networks (GNNs) have achieved remarkable performance on various tasks in different domains, such as node/graph classification (Hamilton, Ying, and Leskovec 2017; Li et al. 2021; Zhao et al. 2021; Veličković et al. 2018; Jiang et al. 2019; Zhang et al. 2021), computer vision (Han et al. 2022; Cui et al. 2022), recommendation (Zhang et al. 2020b; Wang et al. 2021), biomolecules (Morehead, Chen, and Cheng 2022; Somnath, Bunne, and Krause 2021).

However, most of the existing GNNs require sufficient labeled data to obtain satisfactory generalization abilities, and the labeled data usually needs expensive cost to collect while the unlabeled data is easy to access. Thus, research community has proposed techniques (Veličković et al. 2019; Jing, Park, and Tong 2021; Kipf and Welling 2016b; Hong et al. 2021; Yang et al. 2020; Liu et al. 2022) for learning representation in a self-supervised manner, which designs and extracts supervision signals from the unlabeled data itself. Deep InfoMax (DIM) (Hjelm et al. 2019) was first proposed in computer vision, Deep Graph Infomax(DGI) (Veličković et al. 2019) then extended it to graph domain and achieved a promising result by maximizing the mutual information between local node representation and global graph summary. As a pioneer work, however, DGI has two major limitations on the use of mutual information. First, after obtaining the encoded node representation and computing the graph summary by a readout function, DGI maximizes the mutual information between local node representation and global graph summary, which indeed ignores more fine-grained mutual information in different-sized receptive fields. Second, DGI just considers the mutual information in the feature space, and does not explore the structural mutual information that is actually meaningful for graph data. As a result, only less informative node representation is obtained, leading to a suboptimal performance on the downstream tasks. Peng et al. (Peng et al. 2020) proposed to learn topology-aware mutual information by reconstructing the adjacency matrix, which is insufficient to capture the diversity of local neighborhoods.

The Information Bottleneck (IB) principle is appealing, since it defines what we mean by a good representation, in terms of the fundamental trade-off between having a concise representation and one with good predictive power (Tishby and Zaslavsky 2015). Information Bottleneck theory provides a framework for constraining task-relevant information sufficiently retained in the learned representation, which encodes representation that is maximally informative about the target while being minimally informative about input data (Yang et al. 2021; Alemi et al. 2017). Essentially, the Information Bottleneck seeks a trade-off between data fit and model generalization. And based on the IB principle, the learned representation is naturally more robust. IB has been applied to learn informative representation in the fields of computer vision and natural language processing. For irregular data, Yu et al. (Yu et al. 2020, 2021) learned the max-

[*] The corresponding author

imally informative subgraph for graph classification via IB. VIB-GSL (Sun et al. 2022) learned graph structure to distill the actionable information under a variational information bottleneck framework. Both of them adopt IB to construct subgraphs for classification in a supervised framework.

In this paper, we present an effective self-supervised method called Deep Graph Structural Infomax (DGSI) to enrich node representation. As a universal data structure, the graph consists of nodes and edges, both of which are vital for mining meaningful information. Although some work has proposed to design/apply mutual information between features/representations, to the best of our knowledge, few studies have focused on the mutual information on structure, which is a challenging but significant problem. Advanced by Information Bottleneck principle, we explicitly derive the structural information constraints to guide the network learning in a self-supervised manner. Intuitively, the derived constraints formally maximize the structural information both edge-wise and irregular local neighborhood-wise. Therefore, the learned node representation not only encodes the semantic information of the node, but also contains the topological structure information. Besides, for the learning of semantic information, we not only maximize the mutual information between node representation and graph summary but also constrain a more fine-grained connection, i.e., between the node and its local receptive field region, which enables the node representation to characterize both local and global semantic information. Ultimately, we integrate structural and representational mutual information constraints as graphical mutual information constraint, and develop a general framework to incorporate them into the node representations. Extensive experiments on different types of datasets demonstrate the superiority of our proposed method in a self-supervised paradigm.

2 Preliminaries and Notations

We start by formally introducing the notations used in this paper. A graph is denoted as $\mathcal{G}\left(\mathcal{V},\mathcal{E}\right)$, where $\mathcal{V}$ represents a set of nodes with the number $|\mathcal{V}| = n$ and $\mathcal{E}$ is a set of edges with the number $|\mathcal{E}| = e$. According to the link relation in $\mathcal{E}$, the corresponding adjacency matrix can be defined as $\mathbf{A} \in \mathbb{R}^{n\times n}$, if $(v_i, v_j) \in \mathcal{E}$, we set $\mathbf{A}_{ij} = 1$, otherwise $\mathbf{A}_{ij} = 0$. $\mathbf{A}_{ij}$ may be assigned to a real value when edges have different weights. Besides, each node usually carries a feature description with a d_x-dimension vector $\mathbf{x} \in \mathbb{R}^{d_x}$. And the features of all nodes could be stacked by rows into a feature matrix $\mathbf{X} \in \mathbb{R}^{n\times d_x}$. To state conveniently, we use $\mathbf{X}_{i\cdot}$ or $\mathbf{x}_i$ to denote the feature of the i-th vertex.

The goal is to learn an encoder in a self-supervised manner where the graph is the input and the learned node representations are the output, formally, $g_{enc} : \mathbb{R}^{n\times d_x} \times \mathbb{R}^{n\times n} \longrightarrow \mathbb{R}^{n\times d_h}$. Node features are then abstracted and transformed into high-level semantic representations $h \in \mathbb{R}^{d_h}$. Such representations can then be used for various downstream tasks, such as node classification, link prediction, etc. Below we illustrate two basic concepts before introducing our method.

Mutual Information (MI): For two random variables x and y whose joint probability distribution is $p(x, y)$, the mutual information (Thomas, Thomas, and Joy 1991; Shannon and Weaver 1949) between them is given by,

$$I(x; y) = \int dx\, dy\ \ p(x, y) \log(\frac{p(x, y)}{p(x)p(y)}). \qquad (1)$$

Mutual information is one of many quantities that measures how much one random variable tells us about another. It can be thought of as the reduction in uncertainty about one random variable given knowledge of another. DGI (Veličković et al. 2019) first extended DIM (Hjelm et al. 2019) to the graph domain by maximizing MI between node and graph representations. Since then, increasing attention (Jing, Park, and Tong 2021; Dong et al. 2022; Sun et al. 2020, 2021; Peng et al. 2020) has been paid to extending DGI, where GMI (Peng et al. 2020) was proposed to learn graphical MI by reconstructing the adjacency matrix, which is insufficient to capture the diversity of local neighborhoods.

Information Bottleneck (IB): Given an input x and its target y, the Information Bottleneck principle aims to learn the minimal sufficient representation h satisfying,

$$\max IB = I(h, y) - \beta I(h, x), \qquad (2)$$

where β is the Lagrangian multiplier trading off sufficiency and minimality. A variational approximation is developed by Alemi et al. (Alemi et al. 2017) to solve the above objective function, approximately equivalent to optimizing,

$$L = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[-\log q(y_i|h) + \beta D_{KL}(p(h|x_i)||r(h))], \quad (3)$$

where $q(y_i|h)$ is the variational approximation of the true posterior $p(y_i|h)$, and $r(h)$ is a variational approximation to distribution $p(h)$.

Information Bottleneck was proposed for data compression (Tishby, Pereira, and Bialek 1999) while maximally preserving the useful information. Recently, GIB (Wu et al. 2020) extended IB into graph neural networks, regularizing structural as well as the feature information to enhance the robustness of the learned representation. Then some works (Yang et al. 2021; Xu et al. 2021; Yu et al. 2020, 2021; Sun et al. 2022; Suresh et al. 2021) introduced IB to learn meaningful representations or structures on graphs. Differently, we explicitly derive the structural information constraints based on IB principle to guide the network learning in a self-supervised manner.

3 Methodology

In this section, we present our proposed DGSI in detail. Specifically, we build our model from two aspects: Structural Mutual Information (SMI) and Representational Mutual Information (RMI), then we formulate the final Graphical Mutual Information (GMI) with a self-supervised inference objective. The whole framework of DGSI is illustrated in Figure 1, where topological structures and representation constraints are jointly optimized. The proposed framework allows various choices of network architectures. We opt for

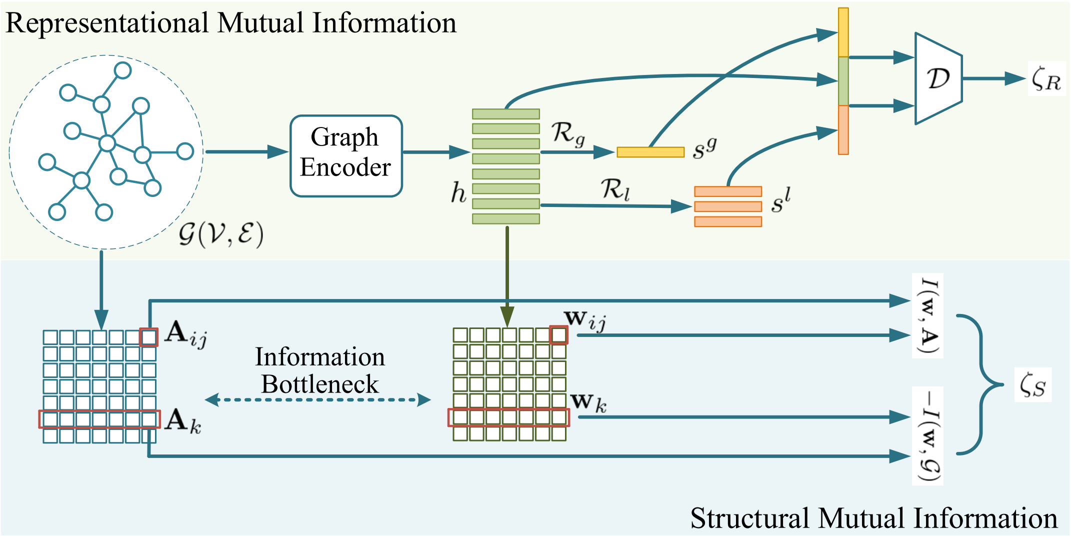


Figure 1: The framework of our deep graph structural infomax.

simplicity and adopt the commonly used GCN (Kipf and Welling 2016a) as our basic graph encoder, i.e.,

$$h_i = \sigma(\sum_{j\in\mathcal{N}_i}(\hat{\mathbf{D}}^{-\frac{1}{2}}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-\frac{1}{2}})_{ij} f(\mathbf{x}_j)), \tag{4}$$

where $\hat{\mathbf{A}} = \mathbf{A}+\mathbf{I}$ and $\hat{\mathbf{D}}_{ii} = \sum_j \hat{\mathbf{A}}_{ij}, \mathcal{N}_i$ is the set including the neighbors of node v_i and itself. f is a linear transformation. σ is the activation function such as ReLU.

3.1 Structural Mutual Information

Topological structures describe the inherent property of graphs, thus those graph-related tasks cannot be divorced from the encoding or representation of graph structures. Although it is challenging due to the irregularity, we aim to formulate mutual information constraint on graph structures for enhancing the network learning process and improving the node representation, differently from only encoding node features in (Veličković et al. 2019). Specifically, we derive structural mutual information constraint based on the theory of information bottleneck (Yang et al. 2021; Alemi et al. 2017) which can learn maximally informative and robust representation.

In the embedding space, the structural relationship may be directly proportional to the relations of the associated nodes, simply speaking, $\mathbf{w}_{ij} = R(h_i, h_j) \doteq \sigma(h_i^\top h_j)$, where h_i, h_j are the hidden node representations to be learned, σ is the activation function such as sigmoid, and $\mathbf{w}_{ij}$ is the expected edge connection strength between the i-th and j-th node. Taking the adjacency matrix $\mathbf{A}$ of graph $\mathcal{G}$ as the self-supervised signal, we explicitly define the structural information bottleneck to derive the structural mutual information constraint.

Definition 1 *(Structural Information Bottleneck) Given a node v_i, the information bottleneck with its the structural representation can be formulated as*

$$SIB_\Theta = max\, I_\Theta(\mathbf{w}_{ij}, \mathbf{A}_{ij}) - \beta I_\Theta(\mathbf{w}_{ij}, \mathcal{G}_i), \tag{5}$$

where $\mathcal{G}_i = \{\mathbf{X}_k, \mathbf{A}_k | k \in \mathcal{N}_i\}$ represents the subgraph including node features and edges related to central node v_i, $\mathcal{N}_i$ is the set including the neighbors of node v_i and itself. $\mathbf{A}_{ij}$ is the element at row i and column j from the adjacency matrix. $\mathbf{w}_{ij}$ denotes the learned local structural relation and is computed from the node representation. Θ represents the network parameters to be learned.

We explicitly model the structural constraint based on information bottleneck theory. The first term $I_\Theta(\mathbf{w}_{ij}, \mathbf{A}_{ij})$ encodes the mutual information between the learned structures and the adjacency matrix. The second term $I_\Theta(\mathbf{w}_{ij}, \mathcal{G}_i)$ is the mutual information between the learned structures and the input subgraph. A Lagrangian multiplier β is introduced to balance these two terms. Ideally, we expect the learned node representation is maximally informative about the target while preserving the necessary information about input data. In other words, we aim to encode sufficient and robust node representation containing structural information.

Due to the high-dimensional feature vector, it is difficult to accurately compute the mutual information between two variables. To solve Eqn. (5), we resort to the variational inference technique. Denote $q(\mathbf{A}_{ij}|\mathbf{w}_{ij})$ as the variational approximation of the true posterior $p(\mathbf{A}_{ij}|\mathbf{w}_{ij})$, we can derive a lower bound of the first term as,

$$\begin{aligned} I_\Theta(\mathbf{w}_{ij}, \mathbf{A}_{ij}) \geq \int d\mathcal{G}_i\, d\mathbf{w}_{ij}\, d\mathbf{A}_{ij}\;\; & p(\mathcal{G}_i)p(\mathbf{w}_{ij}|\mathcal{G}_i) \\ & p(\mathbf{A}_{ij}|\mathcal{G}_i)\log q(\mathbf{A}_{ij}|\mathbf{w}_{ij}). \end{aligned} \tag{6}$$

Let $r(\mathcal{G}_i)$ denote a variational approximation to prior distribution $p(\mathcal{G}_i)$, the second term could reach an upper bound as,

$$I_\Theta(\mathbf{w}_{ij}, \mathcal{G}_i) \leq \int d\mathbf{w}_{ij}\, d\mathcal{G}_i\;\; p(\mathcal{G}_i)p(\mathbf{w}_{ij}|\mathcal{G}_i)\log\frac{p(\mathbf{w}_{ij}|\mathcal{G}_i)}{r(\mathbf{w}_{ij})}. \tag{7}$$

By plugging Eqn. (6) and Eqn. (7) into Eqn. (5), we can derive the entire lower bound of structural information bottleneck as:

$$SIB_\Theta \geq \int d\mathcal{G}_i \, d\mathbf{w}_{ij} \, d\mathbf{A}_{ij} \\ p(\mathcal{G}_i)p(\mathbf{w}_{ij}|\mathcal{G}_i)p(\mathbf{A}_{ij}|\mathcal{G}_i) \log q(\mathbf{A}_{ij}|\mathbf{w}_{ij}) \\ -\beta \int d\mathbf{w}_{ij} \, d\mathcal{G}_i \;\; p(\mathcal{G}_i)p(\mathbf{w}_{ij}|\mathcal{G}_i) \log \frac{p(\mathbf{w}_{ij}|\mathcal{G}_i)}{r(\mathbf{w}_{ij})}. \tag{8}$$

We maximize the SIB_Θ by optimizing the lower bound, which is approximately equivalent to minimizing the following objective formula,

$$\zeta_S = \frac{1}{n^2} \sum_{ij} \mathbb{E}\left[-\log q(\mathbf{A}_{ij}|\mathbf{w}_{ij})\right] \\ + \beta \, D_{KL}\left[p(\mathbf{w}_{ij}|\mathcal{G}_i) || r(\mathbf{w}_{ij})\right]. \tag{9}$$

The first term indicates that the learned structural relationship should be as close to the original adjacency as possible for node v_i, which can be written as $L_{CE}(\mathbf{A}_{ij}, \mathbf{w}_{ij})$, where L_{CE} is the cross-entropy function. The second term represents the Kullback-Leibler (KL) divergence between the $p(\mathbf{w}_{ij}|\mathcal{G}_i)$ and its prior distribution $r(\mathbf{w}_{ij})$. We can obtain the statistic prior distribution from $\mathbf{A}_{ij}$. Since the value of $\mathbf{w}_{ij}$ is predicted as 1 or 0 indicating if there exists an edge, the KL divergence term in practice has the same effect as the first term, i.e., constraining the learned $\mathbf{w}_{ij}$ to be as close to the target $\mathbf{A}_{ij}$ as possible. Thus, we instead compute the KL divergence between the learned local structural distribution and its prior distribution, i.e., $D_{KL}(\mathbf{w}_i || \mathbf{A}_i)$. So far we reach the final structural mutual information constraint,

$$\min\nolimits_\Theta \;\; \zeta_S = \zeta_{CE} + \beta \zeta_{KL} \\ = \frac{1}{n^2} \sum_{ij} L_{CE}(\mathbf{A}_{ij}, \mathbf{w}_{ij}) + \beta \frac{1}{n} \sum_i D_{KL}(\mathbf{w}_i || \mathbf{A}_i). \tag{10}$$

Intuitively, the first term is an edge-wise prediction term, which encourages that the edge values can be preserved in the transformed feature space. The second term is the neighborhood-wise prediction term, which promotes the connection distribution of the local neighborhood centered by node v_i close to the statistic prior distribution. By jointly constraining these two terms, we can more comprehensively encode structure information into the node representation. Please refer to the supplementary material[1] for a detailed derivation about Eqns. (6)-(9).

3.2 Representational Mutual Information

Representational Mutual Information mainly builds the encoding process on node features. To make full use of different-sized receptive fields, we maximize representational mutual information from local to global receptive spaces. Next we elaborate on the local and global mutual information constraints between representations, respectively.

Local Mutual Information Maximization: Given hidden node representation h, we define a local readout function as $\mathcal{R}_l : \mathbb{R}^{|\mathcal{N}| \times d_h} \longrightarrow \mathbb{R}^{d_h}$ to acquire the neighborhood summary. Taking node v_i and its neighbors as an example,

$$s_i^l = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} h_j, \tag{11}$$

where $\mathcal{N}_i$ is the set including the neighbors of node v_i and itself, $|\mathcal{N}_i|$ is the number of neighbors. s_i^l is local neighborhood summary centered by node v_i.

According to the information theory, the mutual information between node $j \in \mathcal{N}_i$ and the local neighborhood summary s_i^l can be written as

$$I_\Theta(h_j; s_i^l) = \int dh_j \, ds_i^l \;\; p(h_j, s_i^l) \log(\frac{p(h_j, s_i^l)}{p(h_j)p(s_i^l)}). \tag{12}$$

We maximize the local mutual dependence $I_\Theta(h_j; s_i^l)$ by approximately minimizing the following objective based on the Jensen Shannon divergence,

$$\min\nolimits_\Theta \;\; \zeta_l = -\mathbb{E}[\log \mathcal{D}(h_j; s_i^l)] - \mathbb{E}[\log(1 - \mathcal{D}(\tilde{h}_j; s_i^l))], \tag{13}$$

where $\tilde{h}_j$ is the output of shuffled graph $\tilde{\mathcal{G}}$ through the same graph encoder, and $\mathcal{D}$ is the discriminator function similar to work (Veličković et al. 2019). Note that Eqn. (13) could be implemented by the Gibbs sampling strategy, to cover as many samples as possible, and we resample the positive and negative pairs for each epoch during the training process.

Global Mutual Information Maximization: We employ a global readout function as $\mathcal{R}_g : \mathbb{R}^{n \times d_h} \longrightarrow \mathbb{R}^{d_h}$ to acquire the graph summary s^g like DGI (Veličković et al. 2019),

$$s^g = \frac{1}{n} \sum_{k \in \mathcal{V}} h_k, \tag{14}$$

where $\mathcal{V}$ includes all nodes in the graph, and $n = |\mathcal{V}|$ is the number of nodes.

Thus the mutual information between node representation and global graph summary can be represented as

$$I_\Theta(h_k; s^g) = \int dh_k \, ds^g \;\; p(h_k, s^g) \log(\frac{p(h_k, s^g)}{p(h_k)p(s^g)}). \tag{15}$$

Similarly, we maximize $I(h_k; s^g)$ by optimizing the following objective function,

$$\min\nolimits_\Theta \;\; \zeta_g = -\mathbb{E}[\log \mathcal{D}(h_k; s^g)] - \mathbb{E}[\log(1 - \mathcal{D}(\tilde{h}_k; s^g))]. \tag{16}$$

Finally, we incorporate the local and global mutual information maximization by a hyperparameter α to formulate the final representational mutual information as,

$$\min\nolimits_\Theta \;\; \zeta_R = \zeta_g + \alpha \zeta_l. \tag{17}$$

For the representational mutual information, we maximize local mutual information to encourage the node representation to capture local neighborhood information while maximizing global mutual information to facilitate the node representation to embody global graph information. By the joint self-supervised constraints, we learn node representation that indicates both local and global characteristics.

[1] https://github.com/wtzhao1631/dgsi

3.3 Graphical Mutual Information

We define graph mutual information as consisting of two parts: structural mutual information and representational mutual information. Given a graph $\mathcal{G}(\mathbf{X}, \mathbf{A})$, and the node representation h learned by an encoder $g_{enc} : \mathbb{R}^{n\times d_x} \times \mathbb{R}^{n\times n} \longrightarrow \mathbb{R}^{n\times d_h}$, then the graphical mutual information is denoted as,

$$I(h;\mathcal{G}) = I_{RMI} + I_{SMI}. \tag{18}$$

Accordingly, we incorporate the representational and structural constraints with a hyperparameter γ to formulate the final graph objective function,

$$\min\nolimits_{\Theta}\ \zeta_{\mathcal{G}} = \zeta_R + \gamma\zeta_S. \tag{19}$$

We develop a general framework that incorporates the global representational mutual information, local representational mutual information, and sufficient structural information into the node representation. As a result, the DGSI model can capture more fine-grained semantic information and beneficial structural information.

4 Experiments

4.1 Experimental Settings

Datasets. We conduct experiments on six real-world node classification datasets: Cora, Pubmed, Citeseer (Kipf and Welling 2016a), Cora-Full, Amazon Photo and Amazon Computers (Shchur et al. 2018). The first four datasets are constructed as citation networks, Cora-Full is an extended version of Cora, Amazon Photo and Amazon Computers are segments of Amazon co-purchase graph, where nodes represent goods, node features are bag-of-words indicating product reviews, there exists an edge if two goods are frequently bought together, and labels are given by product category.

Baselines. We compare our DGSI with state-of-the-arts on node classification task, which can be divided into two types. The supervised approaches include GCN (Kipf and Welling 2016a), GraphNAS (Gao et al. 2020), Graph-Bert (Zhang et al. 2020a), g-U-Nets (Gao and Ji 2019), JKNet (Xu et al. 2018), and GAT (Veličković et al. 2018). The self-supervised approaches include GAE (Kipf and Welling 2016b), DGI (Veličković et al. 2019), GraphCL (You et al. 2020), mvgrl (Hassani and Khasahmadi 2020), SUGRL (Mo et al. 2022), GCA (Zhu et al. 2021) and InfoGCL (Xu et al. 2021).

Implementation Details. A two-layer GCN is adopted as the basic framework for self-supervised learning on graphs, and the outputs of the first and second layer are summed as the learned node representation. Both the number of hidden units and the dimension of learned representation are set as 512. Prelu is leveraged as a nonlinear activation function. The learning rate is 0.0005 for all datasets. We determine the weight of each term in the objective by grid search. The node representation is first obtained by self-supervised learning, then fed into a fully-connected layer to predict the node target, which is a two-phase learning process (Veličković et al. 2019; Jing, Park, and Tong 2021; Peng et al. 2020). For the second phase, we carry out two experimental settings to evaluate the node representation obtained by DGSI. The first one is to follow DGI, adopting a widely-used train/validation/test set on the Cora, Citeseer, and Pubmed datasets, where 20 training samples per class. The second one randomly samples $[1, 5]$ labeled data per class to train the network, which is a label scarcity setting, and all six datasets are evaluated under this setting. Besides, we also set an early stopping with patience as 20. Classification accuracy is the metric, and results are reported by the way of "mean accuracy ± standard deviation".

Method	Cora	Citeseer	Pubmed
	Supervised Approaches		
GCN	81.5	70.3	79.0
GraphNAS	83.7	73.5	80.5
Graph-Bert	84.3	71.2	79.3
g-U-Nets	84.4 ± 0.6	73.2 ± 0.5	79.6 ± 0.2
JKNet	82.7 ± 0.4	73.0 ± 0.5	77.9 ± 0.4
GAT	83.0 ± 0.7	72.5 ± 0.7	79.0 ± 0.3
	Self-supervised Approaches		
GAE	71.5 ± 0.4	65.8 ± 0.4	72.1 ± 0.5
DGI	83.8 ± 0.5	72.0 ± 0.6	77.9 ± 0.3
GraphCL	82.5 ± 0.1	73.1 ± 0.2	-
mvgrl*	83.5 ± 0.5	73.3 ± 0.5	80.1 ± 0.7
SUGRL	83.4 ± 0.5	73.0 ± 0.4	**81.9 ± 0.3**
GCA	81.8 ± 0.2	71.9 ± 0.4	81.0 ± 0.3
InfoGCL	83.5 ± 0.3	73.5 ± 0.4	79.1 ± 0.2
DGSI(ours)	**84.62 ± 0.16**	**74.02 ± 0.15**	81.37 ± 0.14

Table 1: Comparison with state-of-the-arts on citation dataset. Accuracy is used as the metric.

4.2 Experimental Results

We first show the performance of our proposed DGSI and baseline approaches on three citation datasets in Table 1. The mvgrl* indicates that we show a reproduce accuracy on the Cora dataset due to the different dataset setting. We can observe that our proposed DGSI obtains a remarkable improvement compared to self-supervised and supervised methods for all datasets. Specifically, compared to DGI, the DGSI model achieves superior performance and obtains 0.82%, 2.02% and 3.47% improvements on Cora, Citeseer and Pubmed datasets respectively. We attribute the gain in performance to the introduced structural as well as local representational mutual information constraints.

Next, we present the classification results of the proposed DGSI and the baselines under the setting in which a few labeled data is used. Here, we take 1-5 labeled samples for each class respectively to evaluate the node representation obtained by the proposed DGSI. We run the released code by authors of four existing methods as our baselines, where GCN and GAT are supervised methods and DGI and GraphCL are self-supervised. The experimental results are reported in Table 2. Overall, DGSI renders an impressive performance. We further observe that our DGSI achieves substantial performance gains under this setting. On the contrary, the supervised GCN and GAT have very low accuracies, especially on Cora, Citeseer, Cora-Full, and Amazon Photo datasets. The reason for this may be that extremely rare supervision signals are difficult to guide the training of

Dataset	Method	1	2	3	4	5
Cora	GCN	54.44	57.29	60.31	64.66	69.83
	GAT	55.32	60.67	63.18	68.99	71.05
	GraphCL	60.51±0.17	71.85±0.13	73.88±0.14	75.51±0.13	76.90±0.08
	DGI	66.32 ± 0.16	73.47 ± 0.17	74.39 ± 0.11	75.07 ± 0.15	77.08 ± 0.11
	DGSI(ours)	**73.24 ± 0.13**	**75.67 ± 0.14**	**77.40 ± 0.15**	**78.88 ± 0.07**	**78.17 ± 0.12**
Citeseer	GCN	23.03	30.10	47.98	42.02	55.51
	GAT	20.12	40.19	39.66	45.21	53.26
	GraphCL	36.41±0.20	47.43±0.21	57.13±0.22	61.650.18	65.08±0.12
	DGI	38.26 ± 0.21	45.60 ± 0.23	59.95 ± 0.16	61.44 ± 0.12	64.60 ± 0.14
	DGSI(ours)	**52.33 ± 0.20**	**54.58 ± 0.16**	**62.05 ± 0.15**	**66.43 ± 0.11**	**68.14 ± 0.11**
Pubmed	GCN	48.94	62.41	65.04	68.02	72.32
	GAT	**63.13**	48.54	54.10	60.51	68.46
	GraphCL	61.74±0.16	**66.88±0.06**	**71.21±0.12**	74.45±0.09	77.26±0.07
	DGI	59.52 ± 0.12	65.48 ± 0.06	70.23 ± 0.15	73.87 ± 0.10	75.73 ± 0.10
	DGSI(ours)	55.17 ± 0.31	63.35 ± 0.12	70.68 ± 0.18	**75.95 ± 0.08**	**80.49 ± 0.05**
Cora-Full	GCN	3.11	4.01	5.04	6.07	5.29
	GAT	18.28	28.37	27.79	30.07	26.01
	GraphCL	27.72±0.15	33.69 ±0.16	38.05±0.37	38.30±0.19	41.58±0.20
	DGI	26.15 ± 0.17	32.46 ± 0.09	37.27 ± 0.21	37.96 ± 0.15	40.88 ± 0.13
	DGSI(ours)	**32.93 ± 0.07**	**40.28 ± 0.06**	**48.11 ± 0.05**	**49.24 ± 0.05**	**51.14 ± 0.06**
Amazon Photo	GCN	27.06	49.54	42.89	65.53	58.34
	GAT	42.00	**67.84**	53.82	60.23	78.79
	GraphCL	46.80±0.23	63.16±0.30	64.28±0.15	67.06±0.23	68.46±0.40
	DGI	49.14 ± 0.18	67.12 ± 0.21	67.78 ± 0.16	70.33 ± 0.30	71.73 ± 0.45
	DGSI(ours)	**60.96 ± 0.10**	66.98 ± 0.06	**71.49 ± 0.04**	**74.49 ± 0.05**	**77.15 ± 0.07**
Amazon Computer	GCN	17.88	23.90	34.38	25.15	28.25
	GAT	53.79	**70.87**	**74.88**	**69.65**	**76.45**
	GraphCL	54.89±0.22	56.42±0.23	58.56±0.45	57.96±0.19	57.31±0.15
	DGI	51.63 ± 0.26	56.27 ± 0.18	58.14 ± 0.36	57.17 ± 0.37	56.05 ± 0.24
	DGSI(ours)	**55.28 ± 0.63**	66.60 ± 0.35	68.36 ± 0.34	69.72 ± 0.20	72.13 ± 0.35

Table 2: Classification accuracy under the setting with different numbers of labeled data per class.

neural networks, which is consistent with the fact that convolutional neural networks require a large amount of training data to obtain good results. Contrastively, self-supervised learning is not vulnerable to the reduction in the number of labels. And proper designing of structural mutual information constraint further can enhance the learning capability of encoder. Note that the GAT model achieves better results than others on Amazon Computer dataset, but our proposed DGSI is still optimal under the self-supervised setting.

4.3 Ablation Study

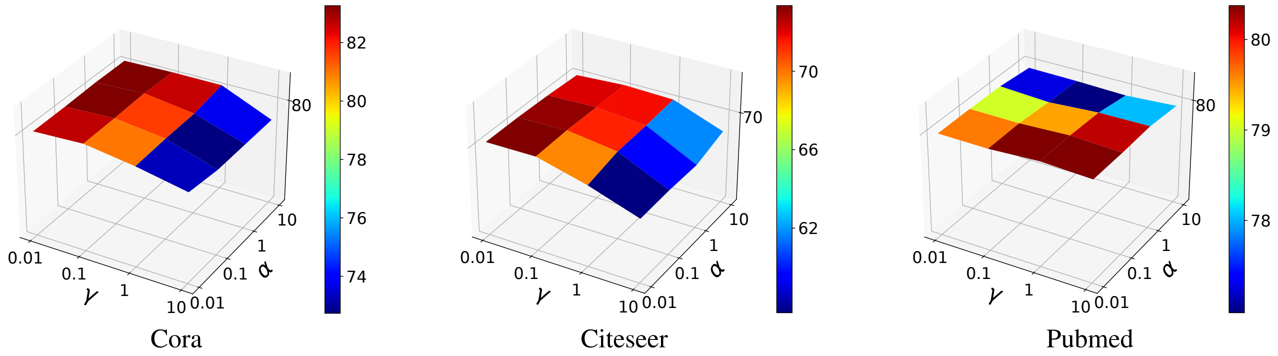
Inspired by the information bottleneck principle, our proposed DGSI model presents a general framework composed of both representational and structural mutual information constraints, where the representational mutual information maximization (Eqn. (17)) includes global mutual information constraint ζ_g and local mutual information constraint ζ_l, the structural mutual information constraint (Eqn. (11)) includes cross entropy constraint ζ_{CE} and Kullback-Leibler divergence ζ_{KL}. Compared with DGI which only develops global mutual information maximization, DGSI additionally introduces other three terms together to enhance the capabilities of the model and improve node representation. Thus, we analyze the effect of each component and their combination. Table 3 shows the classification accuracy on three citation datasets, and we have the following several insights,

- For rows 1 to 4, the accuracies are overall not desirable when removing the global mutual information constraint. Thus ζ_g always exists as a basic term.
- For row 5, when only ζ_g is used, the accuracies are still significantly lower than the best results.
- For rows 6 and 7, when adding ζ_l or ζ_{CE} to ζ_g, there is little performance improvement. But if only removing ζ_l or ζ_{CE}, the performance on the Cora and Citeseer datasets in rows 10 and 11 is not the best either.
- For row 8, ζ_{KL} achieves relatively large gains than that in row 5 on all three datasets. On this basis, adding ζ_l (row 10) or ζ_{CE} (row 11) has a small gain, but for Pubmed, achieving a new SOTA of 81.78%.
- For row 12, when all four items are adopt, leading to the best accuracies on Cora and Citeseer datasets. For no same effect in pubmed, we attribute this to different dataset properties.

From the above observations, we can conclude that the proposed DGSI model indeed obtains a remarkable improvement compared to DGI, where the term ζ_{KL} is more obvious. Although adding ζ_l or ζ_{CE} alone has a marginal effect, their combinations with ζ_{KL} render better results,

Row	ζ_g	ζ_l	ζ_{CE}	ζ_{KL}	Cora	Citeseer	Pubmed
1		✓			76.20 ± 0.26	58.91 ± 0.55	68.23 ± 0.32
2			✓		73.02 ± 0.25	63.43 ± 0.37	73.93 ± 0.24
3				✓	57.75 ± 0.46	35.27 ± 0.51	67.26 ± 0.82
4		✓	✓	✓	82.31 ± 0.21	43.03 ± 0.42	66.43 ± 0.67
5	✓				81.39 ± 0.11	73.11 ± 0.11	80.67 ± 0.13
6	✓	✓			81.39 ± 0.21	73.04 ± 0.16	80.57 ± 0.13
7	✓		✓		81.93 ± 0.11	72.89 ± 0.12	79.94 ± 0.08
8	✓			✓	83.11 ± 0.16	73.74 ± 0.13	81.27 ± 0.12
9	✓	✓	✓		82.0 ± 0.15	72.72 ± 0.15	79.90 ± 0.08
10	✓	✓		✓	84.23 ± 0.12	73.84 ± 0.12	$\mathbf{81.78 \pm 0.13}$
11	✓		✓	✓	83.41 ± 0.14	73.7 ± 0.12	81.38 ± 0.10
12	✓	✓	✓	✓	$\mathbf{84.62 \pm 0.16}$	$\mathbf{74.02 \pm 0.15}$	81.37 ± 0.14

Table 3: The effect of different components.

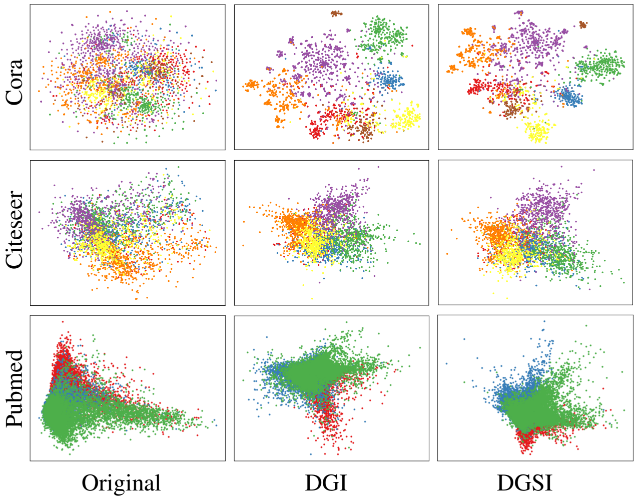


Cora

Citeseer

Pubmed

Figure 2: Hyperparameter sensitivity of α and γ.



Original

DGI

DGSI

Figure 3: Visualization of original data and node representations learned from DGI and DGSI (Ours).

which indicating they are mutually beneficial and can collaborate to obtain promising results. At the same time, they have different effects on different datasets.

For the hyperparameter sensitivity, there are three hyperparameters α, β, and γ. In experiments, we usually regard the structural constraint as a whole and set $\beta = 1$. Then we mainly tune the hyperparameters α and γ for the local representational and structural constraints respectively. We vary the values in the range $[0.01, 0.1, 1, 10]$. The corresponding results are shown in Figure 2. We find that the behavior of the proposed DGSI framework is relatively stable. Besides, γ plays a significant role in Cora and Citeseer datasets while α has a greater impact on Pubmed.

The computational complexity of GCN (Kipf and Welling 2016a) is about $\mathcal{O}(ed_x + nd_xd_h)$, where n, e is the number of nodes and edges, d_x and d_h are the dimensions of the input/hidden layer. The computational complexity of the proposed DGSI is $\mathcal{O}(ed_x + nd_xd_h) + \mathcal{O}(nd_h^2)$. This extra item is caused by the discriminator function $\mathcal{D}$ in Eqns. (13)/(16).

Also, Figure 3 shows the visualization. The left is the original unprocessed data projected into 2D space through principal component analysis. The medium and right are node representations obtained by self-supervised learning from DGI and our proposed DGSI. Compared to DGI, the DGSI presents relatively clearer boundaries.

5 Conclusion

In this paper, we present an effective self-supervised model called Deep Graph Structural Infomax (DGSI) for learning node representation. We derive the structural mutual information constraints from the Information Bottleneck theory and impose restrictions on both edge-wise and neighborhood-wise structural topology. We formulate the graphical mutual information constraint by fusing the structural and representational constraints as a general framework. Extensive experimental results on different types of datasets demonstrate the superiority of the proposed DGSI.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62072244), the Natural Science Foundation of Shandong Province (Grant No. ZR2020LZH008), in part by the Innovation Foundation of Maritime Defense Technologies Innovation Center, in part by State Key Laboratory of High-end Server & Storage Technology, Singapore MOE AcRF Tier 1 Funding (RG18/21, RS05/21) and NTU SUG.

References

Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2017. Deep variational information bottleneck. In *Proceedings of the International Conference on Learning Representations*.

Cui, Z.; Zhou, L.; Wang, C.; Xu, C.; and Yang, J. 2022. Visual Micro-Pattern Propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Dong, W.; Wu, J.; Luo, Y.; Ge, Z.; and Wang, P. 2022. Node Representation Learning in Graph via Node-to-Neighbourhood Mutual Information Maximization.

Gao, H.; and Ji, S. 2019. Graph u-nets. In *Proceedings of the International Conference on Machine Learning*, 2083–2092.

Gao, Y.; Yang, H.; Zhang, P.; Zhou, C.; and Hu, Y. 2020. Graph Neural Architecture Search. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1403–1409.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, 1024–1034.

Han, K.; Wang, Y.; Guo, J.; Tang, Y.; and Wu, E. 2022. Vision GNN: An Image is Worth Graph of Nodes. In *Advances in Neural Information Processing Systems*.

Hassani, K.; and Khasahmadi, A. H. 2020. Contrastive multi-view representation learning on graphs. In *Proceedings of the International Conference on Machine Learning*, 4116–4126.

Hjelm, R. D.; Fedorov, A.; Lavoie-Marchildon, S.; Grewal, K.; Bachman, P.; Trischler, A.; and Bengio, Y. 2019. Learning deep representations by mutual information estimation and maximization. In *Proceedings of the International Conference on Learning Representations*.

Hong, X.; Zhang, T.; Cui, Z.; Huang, Y.; Shen, P.; Li, S.; and Yang, J. 2021. Graph game embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 7711–7720.

Jiang, J.; Cui, Z.; Xu, C.; and Yang, J. 2019. Gaussian-Induced Convolution for Graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Jing, B.; Park, C.; and Tong, H. 2021. Hdmi: High-order deep multiplex infomax. In *Proceedings of the Web Conference 2021*, 2414–2424.

Kipf, T. N.; and Welling, M. 2016a. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

Kipf, T. N.; and Welling, M. 2016b. Variational graph auto-encoders. In *Advances in Neural Information Processing Systems*.

Li, H.; Wang, X.; Zhang, Z.; Yuan, Z.; Li, H.; and Zhu, W. 2021. Disentangled Contrastive Learning on Graphs. In *Advances in Neural Information Processing Systems*, volume 34.

Liu, Y.; Zheng, Y.; Zhang, D.; Chen, H.; Peng, H.; and Pan, S. 2022. Towards unsupervised deep graph structure learning. In *Proceedings of the ACM Web Conference 2022*, 1392–1403.

Mo, Y.; Peng, L.; Xu, J.; Shi, X.; and Zhu, X. 2022. Simple Unsupervised Graph Representation Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 7797–7805.

Morehead, A.; Chen, C.; and Cheng, J. 2022. Geometric Transformers for Protein Interface Contact Prediction. In *Proceedings of the International Conference on Learning Representations*.

Peng, Z.; Huang, W.; Luo, M.; Zheng, Q.; Rong, Y.; Xu, T.; and Huang, J. 2020. Graph representation learning via graphical mutual information maximization. In *Proceedings of the Web Conference 2020*, 259–270.

Shannon, C. E.; and Weaver, W. 1949. The Mathematical Theory of Communication. *Philosophical Review*, 60(3).

Shchur, O.; Mumme, M.; Bojchevski, A.; and Günnemann, S. 2018. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*.

Somnath, V. R.; Bunne, C.; and Krause, A. 2021. Multi-scale representation learning on proteins. In *Advances in Neural Information Processing Systems*, volume 34, 25244–25255.

Sun, F.-Y.; Hoffmann, J.; Verma, V.; and Tang, J. 2020. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *Proceedings of the International Conference on Learning Representations*.

Sun, Q.; Li, J.; Peng, H.; Wu, J.; Fu, X.; Ji, C.; and Yu, P. S. 2022. Graph Structure Learning with Variational Information Bottleneck.

Sun, Q.; Li, J.; Peng, H.; Wu, J.; Ning, Y.; Yu, P. S.; and He, L. 2021. Sugar: Subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism. In *Proceedings of the Web Conference 2021*, 2081–2091.

Suresh, S.; Li, P.; Hao, C.; and Neville, J. 2021. Adversarial graph augmentation to improve graph contrastive learning. In *Advances in Neural Information Processing Systems*, volume 34, 15920–15933.

Thomas, M.; Thomas; and Joy, A. 1991. Elements of information theory. *Publications of the American Statistical Association*, 103(481): 429–429.

Tishby, N.; Pereira, F. C.; and Bialek, W. 1999. The information bottleneck method. *Proceedings of the 37-th Annual Allerton Conference on Communication Control and Computing*, 368–377.

Tishby, N.; and Zaslavsky, N. 2015. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop*, 1–5.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *Proceedings of the International Conference on Learning Representations*.

Veličković, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep Graph Infomax. In *Proceedings of the International Conference on Learning Representations*, volume 2, 4.

Wang, S.; Hu, L.; Wang, Y.; He, X.; Sheng, Q. Z.; Orgun, M. A.; Cao, L.; Ricci, F.; and Yu, P. S. 2021. Graph Learning based Recommender Systems: A Review. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 4644–4652.

Wu, T.; Ren, H.; Li, P.; and Leskovec, J. 2020. Graph information bottleneck. In *Advances in Neural Information Processing Systems*, volume 33, 20437–20448.

Xu, D.; Cheng, W.; Luo, D.; Chen, H.; and Zhang, X. 2021. Infogcl: Information-aware graph contrastive learning. In *Advances in Neural Information Processing Systems*, volume 34.

Xu, K.; Li, C.; Tian, Y.; Sonobe, T.; Kawarabayashi, K.-i.; and Jegelka, S. 2018. Representation learning on graphs with jumping knowledge networks. In *Proceedings of the International Conference on Machine Learning*, 5453–5462.

Yang, L.; Gu, J.; Wang, C.; Cao, X.; Zhai, L.; Jin, D.; and Guo, Y. 2020. Toward unsupervised graph neural network: Interactive clustering and embedding via optimal transport. In *2020 IEEE International Conference on Data Mining*, 1358–1363.

Yang, L.; Wu, F.; Zheng, Z.; Niu, B.; Gu, J.; Wang, C.; Cao, X.; and Guo, Y. 2021. Heterogeneous Graph Information Bottleneck. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1638–1645.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems*, volume 33, 5812–5823.

Yu, J.; Xu, T.; Rong, Y.; Bian, Y.; Huang, J.; and He, R. 2020. Graph information bottleneck for subgraph recognition. In *Proceedings of the International Conference on Learning Representations*.

Yu, J.; Xu, T.; Rong, Y.; Bian, Y.; Huang, J.; and He, R. 2021. Recognizing predictive substructures with subgraph information bottleneck. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhang, J.; Zhang, H.; Xia, C.; and Sun, L. 2020a. Graph-Bert: Only Attention is Needed for Learning Graph Representations. *arXiv preprint arXiv:2001.05140*.

Zhang, T.; Cui, B.; Cui, Z.; Huang, H.; Yang, J.; Deng, H.; and Zheng, B. 2020b. Cross-graph convolution learning for large-scale text-picture shopping guide in e-commerce search. In *2020 IEEE 36th International Conference on Data Engineering*, 1657–1666. IEEE.

Zhang, T.; Wang, Y.; Cui, Z.; Zhou, C.; Cui, B.; Huang, H.; and Yang, J. 2021. Deep Wasserstein Graph Discriminant Learning for Graph Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 10914–10922.

Zhao, W.; Fang, Y.; Cui, Z.; Zhang, T.; and Yang, J. 2021. Graph Deformer Network. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 1646–1652.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference 2021*, 2069–2080.