DisenSemi: Semi-Supervised Graph Classification via Disentangled Representation Learning

Yifan Wang, Xiao Luo, Chong Chen, *Member, IEEE*, Xian-Sheng Hua, *Fellow, IEEE*, Ming Zhang, *Member, IEEE*, and Wei Ju, *Member, IEEE*

***Abstract*— Graph classification is a critical task in numerous multimedia applications, where graphs are employed to represent diverse types of multimedia data, including images, videos, and social networks. Nevertheless, in the real world, labeled graph data are always limited or scarce. To address this issue, we focus on the semi-supervised graph classification task, which involves both supervised and unsupervised models learning from labeled and unlabeled data. In contrast to recent approaches that transfer the entire knowledge from the unsupervised model to the supervised one, we argue that an effective transfer should only retain the relevant semantics that align well with the supervised task. We introduce a novel framework termed DisenSemi in this article, which learns disentangled representation for semi-supervised graph classification. Specifically, a disentangled graph encoder is proposed to generate factorwise graph representations for both supervised and unsupervised models. Then, we train two models via supervised objective and mutual information (MI)-based constraints, respectively. To ensure the meaningful transfer of knowledge from the unsupervised encoder to the supervised one, we further define an MI-based disentangled consistency regularization between two models and identify the corresponding rationale that aligns well with the current graph classification task. Experiments conducted on various publicly available datasets demonstrate the effectiveness of our DisenSemi.**

***Index Terms*— Disentangled representation learning, graph neural networks (GNNs), semi-supervised graph classification.**

I. Introduction

GRAPHS are among the most versatile data structures in many real-world multimedia applications across various domains, e.g., biology networks [1], molecule structures [2], citation networks [3], social networks [4], and recommender system [5]. One fundamental task for graph-structured data is graph classification, which endeavors to capture the characteristics of the entire graph and has become an important research hot spot in numerous multimedia tasks, including chemical compound prediction [2], text categorization [6], and social network analysis [4].

To tackle the graph classification problem, many machine learning-based methods are proposed. Graph kernel methods leverage structural patterns, such as shortest paths [7], subtrees [8], subgraphs [9], and graphlets [10] to measure similarity among graphs. These methods then use the similarity matrix with a kernel-based supervised algorithm, like the support vector machine (SVM), to perform classification. Instead of hand-crafted feature extraction in graph kernel methods, more recently, graph neural networks (GNNs) can extract graph structural features in a supervised end-to-end manner. Despite the superior performance, GNNs often require a substantial amount of labeled data, which can be costly and time-consuming in real-world scenarios [11].

In practice, there always exist massive unlabeled graph samples. Although labels of these graphs are unavailable, the inherent knowledge of the graph may help to enhance GNNs' encoding more expressive and discriminative. Semi-supervised classification methods combine both supervised and unsupervised models, where the unsupervised model learns from unlabeled data and can serve as a regularizer. Indeed, there are a handful of semi-supervised works proposed for graph classification, which can be categorized into several types, i.e., pseudolabeling [12], [13], [14] and consistency learning [2], [15], [16], [17], [18], [19]. Especially for consistency learning, methods like InfoGraph [16] and GraphSpa [19] jointly learn supervised and unsupervised models from labeled and unlabeled data. Meanwhile, a popular fashion (i.e., GraphCL [15] and JOAO [17]) has been developed recently, which first pretrain the model in an unsupervised manner using unlabeled data and then fine-tune the model with labeled data.

However, these previous semi-supervised graph classification methods still face the following limitations. First, *neglect the intricate interactions of latent factors.* In the real world, the formation of graphs often results from the intricate interactions of various latent factors. For instance, a molecular graph may include diverse groups of atoms and bonds that represent distinct functional units [20]. Existing works often

Manuscript received 29 November 2023; revised 10 May 2024; accepted 12 July 2024. This work was supported in part by the "Fundamental Research Funds for the Central Universities" in University of International Business and Economics (UIBE) under Grant 23QN02 and in part by the National Natural Science Foundation of China (NSFC) under Grant 62306014 and Grant 62276002. *(Corresponding author: Wei Ju.)*

Yifan Wang is with the School of Information Technology and Management, University of International Business and Economics, Beijing 100029, China (e-mail: yifanwang@uibe.edu.cn).

Xiao Luo is with the Department of Computer Science, University of California at Los Angeles, Los Angeles, CA 90095 USA (e-mail: xiaoluo@cs.ucla.edu).

Chong Chen and Xian-Sheng Hua are with the Terminus Group, Beijing 100027, China (e-mail: chenchong.cz@gmail.com; huaxiansheng@gmail.com).

Ming Zhang is with the School of Computer Science, National Key Laboratory for Multimedia Information Processing, Peking University-Anker Embodied AI Laboratory, Peking University, Beijing 100871, China (e-mail: mzhang_cs@pku.edu.cn).

Wei Ju is with the College of Computer Science, Sichuan University, Chengdu 610064, China (e-mail: juwei@pku.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2024.3431871

overlook the interactions of latent factors and the extracted features are holistic, which harms interpretability and leads to suboptimal performance for the graph classification task. Second, *mismatch semantics between supervised and unsupervised tasks.* Supervised and unsupervised tasks on labeled and unlabeled graph data may capture different information or occupy separate semantic spaces, corresponding to distinct graph factors. Simply combining two tasks may lead to a "negative transfer" [21]. Therefore, we argue that a more efficient way is to disentangle the graph representation learned from unsupervised tasks into distinct latent factors and transfer the corresponding factorwise information well aligned to semantically enhance the supervised task for semi-supervised graph classification.

Recently, disentangled representation learning has gained much attention, aiming at learning factorized representations that are capable of uncovering information about the salient (or explanatory) properties of the data. Moreover, disentangled representations have been demonstrated to be more generalizable and resilient to complex variants, i.e., the learned factors are supposed to be independent and less sensitive to the noises from other factors in the limited observed training data [22]. For graph-structured data, where multiple heterogeneous relations are always mixed and collapsed into one graph, disentangled representation learning tails to decompose underlying relations and is mainly focused on the supervised link prediction tasks (i.e., recommender systems [5], [23]), supervised graph classification tasks [24], [25], and unsupervised graph generation/identification process [26], [27]. However, the semi-supervised method of bridging the supervised and unsupervised tasks remains largely unexplored.

Toward this end, we propose *DisenSemi* in this article, a novel *Disen*tangled graph representation learning framework for *Semi*-supervised graph classification. This framework enhances the supervised prediction task with the unsupervised representation learning tasks in a semantically meaningful way. In contrast to existing works that directly utilize an entangled representation learned from unlabeled data for prediction, our proposed approach emphasizes regularizing the rationale to explicitly exploit transferable factor information between supervised and unsupervised tasks. Specifically, we propose a disentangled graph encoder that characterizes global-level topological semantics. The encoder first decomposes the entire graph into several factor graphs. Then, the factorwise interpretable graph representation is extracted through multiple channels of the message-passing layer. Each channel is specifically designed to propagate features within one-factor graph, with a separate readout operation in each channel summarizing a distinct aspect of the graph. For labeled data, we train the model using a supervised objective function while for unlabeled data, the model is trained with mutual information (MI)-based constraints for the input factorized graph and its corresponding features to ensure the disentanglement. Next, we conduct MI maximization between the supervised and unsupervised models under each latent factor instead of in the whole feature space for disentangled consistency regularization. Compared with the existing works, this novel factorwise MI estimation strategy can ensure the regularized factor is best pertinent to the aspect bridging supervised and unsupervised models for the current semi-supervised graph classification task. More importantly, we demonstrate that our framework can be formalized as a problem of maximizing the log-likelihood solved by expectation maximization (EM).

To summarize, we make the following contributions.

1) *Conceptual:* We propose a novel disentangled representation learning framework for the semi-supervised graph classification task, which explicitly models the rationale factor that fits well between supervised and unsupervised learning models. Also, the proposed framework can be applied to other semi-supervised learning tasks.
2) *Methodological:* We propose a graph disentangled encoder that produces factorwise graph representations under decomposed factor graphs. Moreover, different MI-based constraints and consistency regularization are proposed to capture the characteristic differences and the connections between supervised and unsupervised learning models.
3) *Experiment:* We perform extensive experiments on a range of public datasets to evaluate the performance of DisenSemi. Experimental results demonstrate the efficiency and outstanding interpretability of our proposed framework for semi-supervised graph classification tasks.

II. Related Works

A. Semi-Supervised Graph Classification

Semi-supervised learning has garnered increasing attention in recent years. It is associated with a paradigm that learns from both labeled and unlabeled data and encompasses two prominent techniques. The first line is the consistency regularization methods, which rely on the manifold or smoothness assumption and posit that minor perturbations of the data points should not affect the model's output. The most common structure is the teacher–student framework, which involves two models: the student and the teacher. This framework applies a consistency constraint between the predictions made by the student and the teacher models [28], [29], [30]. The second line is pseudolabeling methods, which involve predicting the label distribution for unlabeled samples and selecting the most confident samples to provide additional guidance during the training process [31], [32].

For graph-structured data, there are also attempts using consistency regularization [11], [14], [15], [16], [18], pseudolabeling [12], [13], [33], [34], or other approaches [19], [35] for semi-supervised learning. Typically, InfoGraph [16] learns supervised and unsupervised models, respectively, and estimates the MI between two models. GraphCL [15] and GLA [14] use the contrastive learning to get graph representations, which are then used in the fine-tuning step for semi-supervised graph classification. DualGraph [13] and UGNN [34] jointly learn the prediction and retrieval modules via posterior regularization during the pseudolabeling process. GraphSpa [19] employs an active learning approach to select informative graphs for semi-supervised model training. However, these approaches fail to disentangle the diverse

underlying factors behind the graph data, making it challenging to identify suitable alignments between supervised and unsupervised models.

B. Disentangled Representation Learning

The goal of disentangled representation learning is to acquire factorized representations capable of discerning and separating underlying latent factors within the observed data [22]. Existing efforts in disentangled representation learning are primarily focused on computer vision [36], [37], natural language processing [3], [38], and recommendation [23], [39].

Recently, there has been a notable surge of interest in applying disentangled representation learning techniques to graph-structured data [5], [24], [25], [26]. DisenGCN [24] learns disentangled node representations using a neighborhood routing mechanism, which partitions the node's neighborhood into several mutually exclusive parts. DisenHAN [5] learns disentangled representation in heterogeneous information network (HIN) by iteratively identifying the primary aspects of the relationships between node pairs and semantically propagating the corresponding information. FactorGCN [25] takes the whole graph as input and produces blockwise interpretable graph-level features for classification. DGCL [26] proposes a self-supervised graph disentangled representation framework via the contrastive learning method. UMGRL [27] provides a self-supervised disentangled representation learning method for the multiplex graph to capture common and private graph information. Our work focuses on learning disentangled representation for the semi-supervised graph classification task. And, our primary objective is to design a framework, where a supervised model can learn from an unsupervised model through the latent semantic space that aligns well with the current graph classification task.

III. Problem Definition and Preliminaries

A. Problem Definition

Definition 1 (Graph): We define a graph as $G = (V, E, X)$, where V and $E \in V \times V$ denote the node set and edge set of the graph, respectively. $X \in \mathbb{R}^{|V| \times d'}$ denote the node feature matrix, where each row $x_i \in \mathbb{R}^{d'}$ represents the initial feature of node i and d' is the node feature dimension. A graph is typically labeled if it contains a class label $y \in \{0, 1\}^C$, where C denotes the number of classes. Otherwise, the graph is considered unlabeled when the class label is unknown.

Definition 2 (Semi-Supervised Graph Classification): Given a set of graphs $\mathcal{G} = \{\mathcal{G}^L, \mathcal{G}^U\}$, where $\mathcal{G}^L = \{G_1, \ldots, G_{|\mathcal{G}^L|}\}$ and $\mathcal{G}^U = \{G_{|\mathcal{G}^L|+1}, \ldots, G_{|\mathcal{G}^L|+|\mathcal{G}^U|}\}$ represent the labeled and unlabeled graphs, respectively. Let $\mathcal{Y}^L = \{y_1, \ldots, y_{|\mathcal{G}^L|}\}$ be the label corresponding to $\mathcal{G}^L$. Semi-supervised graph classification seeks to learn a prediction function that can assign class labels to the unlabeled graph data in $\mathcal{G}^U$ based on the class labels available in $\mathcal{G}^L$.

B. Preliminaries

Preliminary 1 (MI Estimation): MI is a key metric for assessing the dependence between two random variables, and it has been utilized across a diverse array of tasks [37], [40]. However, estimation MI is intractable especially in high-dimensional continuous settings. Since earlier nonparametric binning methods perform poor, a neural estimator MINE [41] formats MI between x and y as Kullback–Leibler (KL)-divergence between their joint distribution p_{xy} and the product of their marginal distributions $p_x \otimes p_y$

$$I_{\mathrm{KL}}(x, y) := D_{\mathrm{KL}}(p_{xy} || p_x \otimes p_y) \tag{1}$$

and derive a lower bound of the KL-divergence for estimation. Following previous works [16], [40], we can further rely on non-KL divergence (i.e., Jensen–Shannon divergences) as an alternative for the MI estimator

$$I_{\mathrm{JS}}(x, y) = \mathbb{E}_{p_{xy}}[-T(x, y)] - \mathbb{E}_{p_x \otimes p_y}[\log(1 - T(x, y))] \tag{2}$$

where $T(\cdot, \cdot)$ is a critic (or score) function approximated by a neural network. In this way, the loss can be seen as a standard binary cross-entropy (BCE) loss between positive samples from the joint distribution and negative samples from the product of the marginals.

Preliminary 2 (MI and Contrastive Learning): More recently, InfoNCE [42] propose a noise contrastive estimation (NCE [43])-based lower bound for MI estimation, defined as follows:

$$I_{\mathrm{NCE}}(x, y) := \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp^{T(x_i, y_i)}}{\frac{1}{N}\sum_{j=1}^{N} \exp^{T(x_i, y_j)}}\right] \tag{3}$$

where N samples $\{(x_i, y_i)\}_{i=1}^N$ drawn from p_{xy} are used to compute the expectation. Contrastive learning encourages the model where representations between positive samples are pulled closer (e.g., graph representations corresponding to the same factor bridging supervised and unsupervised tasks) and representations among negative samples are pushed apart.

IV. Proposed Model

A. Overview

For semi-supervised graph learning, the objective is to smooth the label information over graph data with regularization. Also, a straightforward way for the objective is to combine the purely supervised loss and the unsupervised objective function. Formally, previous semi-supervised learning frameworks try to minimize the following objective function:

$$L_{\mathrm{total}} = \sum_{i=1}^{|\mathcal{G}^L|} L_S(G_i, y_i) + \lambda \sum_{j=1}^{|\mathcal{G}|} L_U(G_j) \tag{4}$$

where L_S denotes the supervised loss, which quantifies the difference between the predictions of the supervised model and the actual labels of the graph data, L_U denotes the unsupervised loss acting as a regularization term, and λ is the relative weight between the two losses.

However, we argue that supervised and unsupervised tasks have different optimization targets, which correspond to different semantic spaces of the graph data. Thus, we propose DisenSemi for semi-supervised graph classification. The basic idea is to explicitly infer the latent factors underlying a

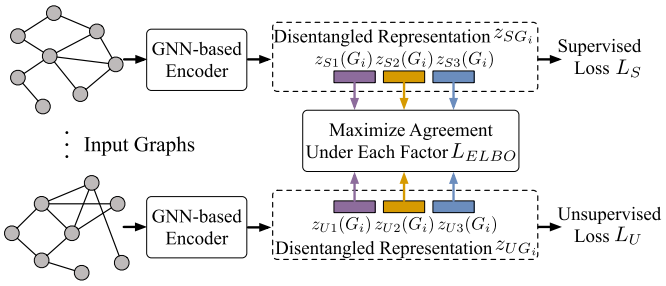


Fig. 1. Schematic of the DisenSemi, which consists of a supervised and an unsupervised model. We extract factorwise graph representations from two models and maximize the agreement in a factorwise manner.

substantial amount of unlabeled graph data and transfer the well-aligned knowledge to enhance the supervised task in a factorwise manner. As shown in Fig. 1, our framework is composed of a supervised model and an unsupervised model, where graph-level representations of both models are learned via the GNN-based encoder. Given the graph data, the GNN-based encoder first factorizes the input graph into several factor graphs, and then gets graph representation by propagating and aggregating features on the corresponding factor graph. For the supervised model, we merge all the extracted factor graph representations to predict graph labels. For the unsupervised model, we introduce several MI-based constraints among factorized graphs and their corresponding extracted representations. Finally, a disentangled consistency regularization is conducted to explicitly identify the rationale between the two models and transfer the learned knowledge semantically via factorwise MI maximization. The process is equivalent to a variational EM algorithm maximizing the log-likelihood.

B. GNN-Based Encoder

GNNs have recently become powerful tools for learning from graph-structured data. Current GNNs typically use neural message passing to update node features and apply a readout function to obtain graph-level representations. However, we argue that these methods are holistic and neglect the hidden factors stemming from different aspects. In fact, multiple relations between nodes are always mixed together and represented as one single edge, and these relations correspond to distinct discriminant aspects of the graph. Thus, given input graph G_i, we factorize it into K factor graphs $\{G_{ik}\}_{k=1}^{K}$ to separately encode their features. Specifically, for each edge $e_m = \{x_u, x_v\}$ in edge set $E(G_i) = \{e_1, \ldots, e_M\}$, where M is the total edge of graph G_i, we project the node feature $x_v \in \mathbb{R}^{d'}$ with a transformation matrix $W \in \mathbb{R}^{d \times d'}$ and calculate the factor coefficients as follows:

$$E_{uvk}(G_i) = \frac{1}{1 + \exp^{-\Psi_k(Wx_u, Wx_v)}} \tag{5}$$

where Ψ_k is the function that computes the attention score for edge e_m under factor k with the features of nodes u and v as input, the function can be implemented via a multilayer perceptron (MLP). Following that, the attention score is normalized to get $E_{uvk}(G_i)$, representing the coefficient of edge e_m contained in the factor graph G_{ik}.

We represent each factor graph by its own edge coefficient $E_{uvk}(G_i)$. By stacking L message-passing layers, the node embedding of each node v in G_{ik} is updated by recursively aggregating and combining its neighborhood features. Formally, the embedding of node v at the lth layer can be

$$\begin{aligned} h_{N(v)}^{(l)}(G_i) &= \mathcal{A}_k^{(l)}\left(\left\{h_{U.K.}^{(l-1)}(G_i), \forall u \in N(v)\right\}, E_{uvk}(G_i)\right) \\ h_{vk}^{(l)}(G_i) &= \mathcal{C}_k^{(l)}\left(h_{vk}^{(l-1)}(G_i), h_{N(v)}^{(l)}(G_i)\right) \\ h_{vk}^{(0)}(G_i) &= W_k x_v \end{aligned} \tag{6}$$

where $N(v)$ represents the neighborhood of node v, $\mathcal{A}_k^{(l)}$ represents the aggregation operation corresponding to the coefficient $E_{uvk}(G_i)$ as edge weight for factor graph G_{ik}, and $\mathcal{C}_k^{(l)}$ represents the combination operations at layer l. In this article, we adopt GraphConv [44] as our message-passing layer due to its strong expressive power and initialize the node embedding of each factor graph via a factor-specific transformation matrix $W_K \in \mathbb{R}^{(d/K)\times d'}$.

Finally, for each factor graph G_{ik}, the graph-level representations are derived by aggregating the embeddings of all nodes from the last Lth layer through a readout function. Formally,

$$z_k(G_i) = \text{READOUT}\left(\left\{h_{vk}^{(L)}(G_i)\right\}_{v \in V(G_i)}\right) \tag{7}$$

where $V(G_i)$ denotes the node set of G_i and READOUT can be implemented as a simple permutation invariant function (i.e., mean function). By considering all the hidden factors of graph data, we can get a factorwise graph representation, namely, $Z_{G_i} = [z_1(G_i), \ldots, z_K(G_i)]$.

C. Supervised Loss

With the GNN-based encoder in the supervised task, a factorwise representation $Z_{SG_i} = [z_{S1}(G_i), \ldots, z_{SK}(G_i)]$ is extracted from labeled data, where different factor graphs will result in different features of the graph. Toward this end, we define K learnable prototypes $\mathcal{C} = \{c_k\}_{k=1}^{K}$ to obtain the attention weight $\alpha_{G_{ik}}$ of factor k for the input G_i

$$\alpha_k(G_i) = \text{Softmax}(\phi(z_{Sk}(G_i), c_k)) \tag{8}$$

where ϕ denotes the similarity function (i.e., cosine similarity). After that, we weighted the sum all latent factors to get the graph representation and fed it into a prediction model to get the output label. The process is shown in Fig. 2(b)

$$\begin{aligned} Z'_{SG_i} &= \sum_{k=1}^{K} \alpha_k(G_i) \cdot z_{Sk}(G_i) \\ \hat{y}_i &= \text{Softmax}\left(\text{MLP}\left(Z'_{SG_i}\right)\right) \end{aligned} \tag{9}$$

where we adopt a two-layer MLP to map the graph representation extracted from different factor graphs to label predictions. For the supervised loss, we adopt cross-entropy, defined as follows:

$$L_S = -\sum_{c=1}^{C} y_{ic} \log \hat{y}_{ic} \tag{10}$$

where $y_i \in \mathbb{R}^C$ is the label with the one-hot encoding for the graph G_i and C corresponds to the total number of classes in the graph dataset.

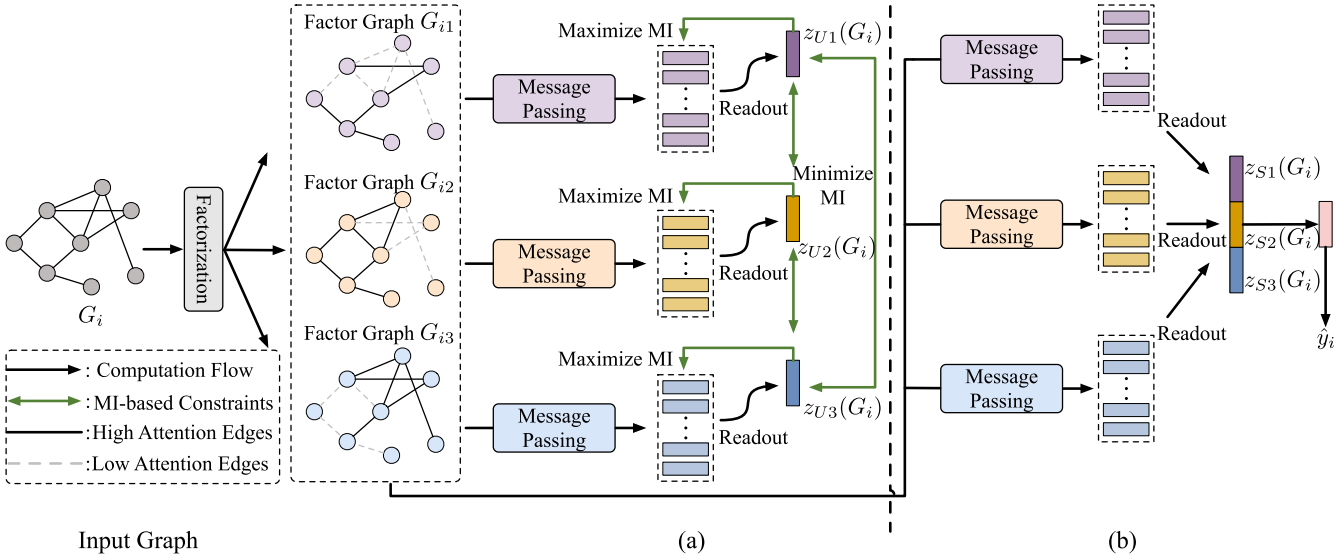


Fig. 2. Illustration of the supervised and unsupervised training modules. We assume that there are three aspects for the input graphs and factorize them into different factor graphs. For the unsupervised module, we maximize the intrafactor and minimize the interfactor MI to disentangle the graph representation effectively. Unlike MI-based constraints in the unsupervised module, we merge all extracted factorwise graph representations to predict graph labels in the supervised module. (a) Unsupervised module. (b) Supervised module.

D. MI-Based Representation Disentanglement

For unsupervised learning, we can also get a factorwise graph representation $Z_{UG_i} = [z_{U1}(G_i), \ldots, z_{U.K.}(G_i)]$ via the GNN-based encoder. Since some of the decomposed factor graphs may exhibit similar structures without additional constraints, we propose several MI-based constraints to effectively disentangle the graph representation, as shown in Fig. 2(a).

1) Intrafactor MI: As each factor graph corresponds to an interpretable relation between nodes, we seek to obtain the discriminant representation that captures the global content of the graph. Inspired by the deep InfoMax [40], we maximize the MI between the decomposed factor graph and the corresponding graph-level representation. For factor k, we define the intrafactor MI on global–local pairs, maximizing the estimated MI over all the nodes in factor graph G_{ik}, which is shown as follows:

$$L^k_{\text{intra}} = \frac{1}{|V|} \sum_{v \in V(G_i)} -I\left(z_{Uk}(G_i), h^{(L)}_{vk}(G_i)\right) \tag{11}$$

where $I(z_{Uk}(G_i), h^{(L)}_{vk}(G_i))$ is the MI estimator, modeled by a discriminator D_k and implemented through a neural network. We use the BCE loss between positive samples from the joint distribution and negative samples from the product of the marginals as the objective

$$\begin{aligned} &-I\left(z_{Uk}(G_i), h^{(L)}_{vk}(G_i)\right) \\ &\quad = \log \sigma\left(D_k(z_{Uk}(G_i), h^{(L)}_{vk}(G_i))\right) \\ &\qquad + \mathbb{E}_{p\otimes\tilde{p}}\left[\log\left(1-\sigma\left(D_k\left(z_{Uk}(G_i), h^{(L)}_{vk}(\tilde{G}_i)\right)\right)\right)\right] \end{aligned} \tag{12}$$

where $h^{(L)}_{vk}(\tilde{G}_i)$ is generated by the negative input graph $\tilde{G}_i$. We sample $\tilde{G}_i$ from a distribution $\tilde{p} = p$ to match the empirical distribution of the input space. In practice, we could sample negative samples by an explicit (stochastic) corruption function or directly sample negative graph instances in a batch.

2) Interfactor MI: For graph representations, Z_{UG_i} extracted from different factor graphs, the MI between every two of them reaches its minimum value 0 when $p(z_{Uk}(G_i), z_{Uk'}(G_i)) = p(z_{Uk}(G_i))p(z_{Uk'}(G_i))$, which means that $z_{Uk}(G_i)$ and $z_{Uk'}(G_i)$ are independent to each other. Thus, MI minimization among them encourages the factorwise graph representation to learn different aspects of information for the graph. Recently, several MI upper bounds [45], [46] have been introduced for MI minimization. However, the estimation MI upper bound among these K graph representations requires $K(K-1)$ times estimation, resulting in far more costs especially, when K is large. To alleviate this problem, as orthogonality is a specific instance of linear independence, we loosen the constraint of minimization MI to orthogonality, and the method has also been demonstrated to be effective by many previous studies [47]

$$\begin{aligned} L_{\text{inter}} &= \left| {Z'_{UG_i}}^{\mathrm{T}} Z'_{UG_i} - I \right| \\ Z'_{UG_i} &= z_{U1}(G_i) || \ldots || z_{UK}(G_i) \end{aligned} \tag{13}$$

where $|\cdot|$ is the L_1 norm, I is the identity matrix, and $||$ is the concatenation operation.

Remark: We have also tested other MI minimization methods, such as contrastive log-ratio upper bound (CLUB) [46], which estimates MI by the log-ratio difference between the conditional distribution of positive and negative pairs. Also, these methods do not significantly impact the performance.

We maximize the MI between the factor graph and its corresponding graph representation to characterize each hidden factor while minimizing the MI among different factor graph representations to enforce representation disentanglement. Formally, unsupervised loss of DisenSemi can be

$$L_U = \sum_{k=1}^{K} L^k_{\text{intra}} + L_{\text{inter}}. \tag{14}$$

E. Disentangled Consistency Regularization

For transferring the learned disentangled graph representations from the unsupervised encoder to the supervised encoder, we encourage representations extracted from two encoders with high MI when they correspond to the same factor of one graph data bridging supervised and unsupervised models. Specifically, we maximize NCE-based MI lower bound [42], which seeks to bring similar representations closer together while pushing dissimilar instances further apart. We treat each representation of a graph data instance as a distinct class and distinguish it from representations extracted from other graph data instances. Formally, given graph dataset $\mathcal{G} = \{G_i\}_{i=1}^{|\mathcal{G}^L|+|\mathcal{G}^U|}$, we assign each graph G_i with a unique surrogate label $s_i = i$. Unlike the label y_i contained in the labeled graph, s_i can be seen as the corresponding ID of the instance in the dataset $\mathcal{G}$. In this way, the MI can be defined as follows:

$$\begin{aligned} I_{\text{NCE}}(Z_{SG_i}, Z_{UG_i}) &:= \log p(s_i|G_i) \\ &= \log \mathbb{E}_{p(k|G_i)}[p(s_i|G_i, k)] \end{aligned} \tag{15}$$

where $p(s_i|G_i)$ corresponding to the identify discrimination over dataset and $p(k|G_i)$ denotes the probability distribution that G_i is reflected via the kth latent factor, namely, can be $\alpha_k(G_i)$. Also, $p(s_i|G_i, k)$ denotes to the discrimination subtask to identify the corresponding ID under the kth latent factor, which can be defined as follows:

$$p(s_i|G_i, k) = \frac{\exp \phi(z_{Sk}(G_i), z_{Uk}(G_i))}{\sum_{j=1}^{|\mathcal{G}|} \exp \phi(z_{Sk}(G_i), z_{Uk}(G_j))} \tag{16}$$

where the subtask is over all graphs in the dataset. ϕ is the similarity function, adopting cosine similarity in this article.

In practice, maximizing the log-likelihood across the entire graph dataset is difficult due to the latent factors. Therefore, we calculate the posterior distribution with Bayes' theorem, which can be defined as follows:

$$p(k|G_i, s_i) = \frac{p(k|G_i)p(s_i|G_i, k)}{\sum_{k=1}^{K} p(k|G_i)p(s_i|G_i, k)}. \tag{17}$$

Compared with prior distribution $p(k|G_i)$ that inferred only given the graph G_i, $p(k|G_i, s_i)$ is the probability that reflects how well the kth latent factor aligns both supervised and unsupervised models when applied to the same graph data. However, computing the posterior probability is prohibitive due to the term $p(s_i|G_i, k)$ in (16), which requires considering all instances in the dataset to compute the denominator. Therefore, we alternatively maximize the evidence lower bound (ELBO) of the log-likelihood, which is defined as follows:

$$\begin{aligned} \log p(s_i|G_i) \geq L_{\text{ELBO}} &:= \mathbb{E}_{q(k|G_i,s_i)}[\log p(s_i|G_i, k)] \\ &\quad - D_{\text{KL}}(q(k|G_i, s_i), p(k|G_i)) \end{aligned} \tag{18}$$

where $q(k|G_i, s_i)$ denotes a variational distribution that approximates the posterior probability $p(k|G_i, s_i)$. Here, we formalize the variational distribution by

$$q(k|G_i, s_i) = \frac{p(k|G_i)\tilde{p}(s_i|G_i, k)}{\sum_{k=1}^{K} p(k|G_i)\tilde{p}(s_i|G_i, k)} \tag{19}$$

where $\tilde{p}(s_i|G_i, k)$ can be defined with NT-Xent loss [48] on a minibatch $\mathcal{B} \subseteq \mathcal{G}$ of graph data

$$\tilde{p}(s_i|G_i, k) = \frac{\exp \phi(z_{Sk}(G_i), z_{Uk}(G_i))}{\sum_{j\in\mathcal{B}, j\neq i}^{|\mathcal{B}|} \exp \phi(z_{Sk}(G_i), z_{Uk}(G_j))}. \tag{20}$$

Note that the process is a variant of the variational EM algorithm, where $q(k|G_i, s_i)$ is inferred during the E-step, and the ELBO is optimized during the M-step.

F. Optimization and Complexity Analysis

1) Optimization: We combine the supervised classification loss L_S, unsupervised loss L_U as well as consistency regularization loss L_{ELBO} together. Also, the overall objective for our semi-supervised graph classification is defined as follows:

$$L_{\text{total}} = \sum_{i=1}^{|\mathcal{G}^L|} L_S + \lambda \sum_{j=1}^{|\mathcal{G}|} L_U - \gamma \sum_{j=1}^{|\mathcal{G}|} L_{\text{ELBO}} \tag{21}$$

where λ is the relative weight between losses and γ is a tunable weight for consistency regularization loss.

2) Complexity Analysis: To ensure scalability with large-scale datasets, we leverage mini-batch with size $B = B^L + B^U$ to compute gradients. The computational consumption is mainly composed of three parts: 1) the unsupervised module; 2) the supervised module; and 3) the disentangled consistency regularization. Given the graph with an average number of nodes $|V|$ and edges $|E|$, the number of GNN layer and factor graphs is L and K, and the representation dimension is d. For 1) and 2), the time complexity of the GNN-based encoder $O(|E|Ld)$. For 1), the additional computational complexity of intrafactor and interfactor MI constraints is $O(|V|d)$ and $O((K-1)d/2)$, respectively. For 2), the additional computational complexity of the label prediction model is $O(d + (d/K)^2 + Cd/K)$ with C classes in the task. For 3), we perform disentangled consistency regularization within minibatch, which takes $O(Bd)$ for each graph. To summarize, we have the overall complexity of DisenSemi, $O((2|E|L+|V|+(K+1)/2+d/K^2+C/K+B)d)$, which scales linearly w.r.t. the number of nodes and edges in each graph. Moreover, DisenSemi requires K factor graphs, prototypes, and corresponding GNN-based encoder in both supervised and unsupervised modules, the space complexity is $O((|V| + |E|)Ld + d + K|E|)$.

V. Experiment

We describe the experimental setup and present a series of extensive experiments conducted on ten real-world benchmark datasets to evaluate the performance of our proposed method. Our goal is to address the following research questions.

1) *RQ1:* How does our proposed method compare with other advanced models in semi-supervised graph classification?
2) *RQ2:* Are the key components of our method (GNN-based encoder, MI-based representation disentanglement, and consistency regularization) essential?
3) *RQ3:* How do different hyperparameters affect the performance of our method?

TABLE I

PERFORMANCE EVALUATION ACROSS TEN BENCHMARK DATASETS. WE REPORT THE AVERAGE PREDICTION ACCURACY, ALONG WITH THE STANDARD DEVIATION FROM TEN RUNS WITH VARIED RANDOM SEEDS (EXPRESSED IN %). BOLDED RESULTS HIGHLIGHT THE BEST PERFORMANCE

Methods	MUTAG	PTC-MR	NCI1	PROTEINS	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M5K	COLLAB	OGB-HIV
GK	82.9 ± 2.1	56.7 ± 2.6	64.7 ± 0.2	67.8 ± 1.0	57.9 ± 0.7	43.9 ± 0.4	68.4 ± 0.5	27.1 ± 0.6	62.3 ± 0.6	94.1 ± 0.4
SP	84.5 ± 2.5	56.4 ± 2.5	69.8 ± 0.3	72.9 ± 0.9	51.9 ± 0.6	34.7 ± 0.7	76.0 ± 1.1	41.3 ± 0.9	58.6 ± 0.3	94.6 ± 0.5
WL	86.8 ± 2.1	58.2 ± 2.8	76.2 ± 0.3	73.5 ± 0.6	69.2 ± 1.0	45.3 ± 0.8	67.7 ± 1.1	44.7 ± 0.8	74.3 ± 0.2	94.3 ± 0.6
MLG	83.5 ± 1.9	58.8 ± 2.2	63.0 ± 0.6	72.3 ± 1.1	49.9 ± 0.3	38.8 ± 0.6	72.7 ± 0.9	35.3 ± 0.8	56.5 ± 0.1	94.8 ± 0.5
DGK	82.9 ± 2.6	59.3 ± 2.5	78.8 ± 0.6	73.7 ± 0.7	69.9 ± 0.7	46.6 ± 1.0	78.7 ± 0.6	48.6 ± 0.1	77.4 ± 0.2	95.0 ± 0.4
Sub2Vec	75.0 ± 2.7	57.6 ± 2.2	58.7 ± 1.5	63.8 ± 1.2	56.1 ± 0.7	35.0 ± 1.1	68.9 ± 0.7	34.0 ± 0.6	58.3 ± 0.5	94.4 ± 0.6
Graph2Vec	74.9 ± 1.4	59.2 ± 2.1	70.6 ± 0.3	70.1 ± 0.9	63.7 ± 1.1	50.4 ± 0.7	76.6 ± 1.0	-	-	-
MVGRL	88.8 ± 0.7	69.5 ± 1.5	72.6 ± 0.5	73.6 ± 0.8	74.5 ± 0.7	50.7 ± 0.6	90.0 ± 0.2	53.1 ± 0.4	77.8 ± 0.2	95.6 ± 0.4
GraphCL	89.4 ± 0.8	70.0 ± 1.6	77.4 ± 0.3	76.1 ± 0.6	74.0 ± 0.5	50.3 ± 0.5	91.7 ± 0.2	57.2 ± 0.6	79.1 ± 0.3	95.8 ± 0.5
JOAO	91.0 ± 0.7	70.6 ± 1.3	77.8 ± 0.3	76.7 ± 0.8	75.2 ± 0.8	50.8 ± 0.5	91.8 ± 0.3	57.4 ± 0.2	79.6 ± 0.1	96.4 ± 0.3
DGCL	90.5 ± 0.9	71.4 ± 1.4	78.1 ± 0.4	76.8 ± 0.7	75.7 ± 0.6	51.7 ± 0.7	91.5 ± 0.5	56.1 ± 0.2	80.3 ± 0.4	96.4 ± 0.2
RGCL	90.5 ± 0.5	71.2 ± 1.2	78.2 ± 0.3	75.0 ± 0.6	75.9 ± 0.6	50.1 ± 0.4	91.1 ± 0.4	56.4 ± 0.3	79.4 ± 0.3	96.3 ± 0.4
InfoGraph	88.9 ± 1.1	70.6 ± 1.4	75.1 ± 0.6	74.9 ± 0.8	74.9 ± 0.8	51.5 ± 0.6	92.0 ± 0.4	56.9 ± 0.5	79.8 ± 0.5	95.6 ± 0.3
GLA	91.6 ± 1.0	70.6 ± 1.7	77.0 ± 0.5	77.1 ± 0.6	75.8 ± 0.6	51.1 ± 0.5	92.2 ± 0.3	56.7 ± 0.2	80.2 ± 0.2	96.8 ± 0.3
TGNN	91.3 ± 0.6	68.3 ± 1.4	75.2 ± 0.7	76.4 ± 1.0	75.1 ± 0.7	50.9 ± 0.5	91.5 ± 0.5	55.6 ± 0.6	77.6 ± 0.4	96.2 ± 0.4
GraphSpa	91.5 ± 0.9	70.2 ± 1.3	77.2 ± 0.5	76.5 ± 0.8	75.4 ± 0.6	51.2 ± 0.6	91.9 ± 0.6	56.0 ± 0.5	78.9 ± 0.3	95.9 ± 0.4
DisenSemi	$\mathbf{92.6 \pm 0.6}$	$\mathbf{72.4 \pm 1.1}$	$\mathbf{78.9 \pm 0.4}$	$\mathbf{78.4 \pm 0.5}$	$\mathbf{76.7 \pm 0.7}$	$\mathbf{52.5 \pm 0.5}$	$\mathbf{93.2 \pm 0.3}$	$\mathbf{57.6 \pm 0.2}$	$\mathbf{81.5 \pm 0.3}$	$\mathbf{97.2 \pm 0.2}$

4) *RQ4:* Can DisenSemi ensure the regularized factor pertinent to the aspect bridging supervised and unsupervised tasks semantically?

A. Experimental Setups

1) Datasets and Baselines: We apply our model to ten public accessible graph classification benchmark datasets[1] following the previous works [14], [16], which includes four molecule datasets: MUTAG, PTC-MR, NCI1, and OGB-HIV, one bioinformatic dataset: PROTEINS and five social network datasets: IMDB-BINARY (IMDB-B), IMDB-MULTI (IMDB-M), REDDIT-BINARY (REDDIT-B), REDDIT-MULTI-5K (REDDIT-M5K), and COLLAB. Following recent works [15], [16], we adopt one-hot degree and centrality features instead when the datasets do not include node attributes. We compare our proposed DisenSemi with 16 recent proposed baselines including seven traditional graph classification approaches: GK [10], SP [7], WL [8], MLG [9], DGK [49], Sub2Vec [50], and Graph2Vec [51] and nine GNN-based semi-supervised graph classification methods: MVGRL [52], GraphCL [15], JOAO [17], DGCL [26], RGCL [53], InfoGraph [16], GLA [14], TGNN [18], and GraphSpa [19].

2) Evaluation Protocol: We assess the models using tenfold cross-validation. Following the recent works [14], [15], [16], we randomly shuffle and split each dataset into ten parts. For each fold, one part is designated as the test set, another as the validation set, and the remaining parts are used for training. We then choose 30% of the training graphs as labeled examples for each fold and apply semi-supervised learning. Results are reported as the average accuracy (%) with standard deviation from ten repeated experiments.

3) Implementation Details: For all datasets, the embedding size of DisenSemi and other baselines (except GK, SP, WL, and MLG) is fixed to 128. As the exact number of latent factors is unknown, we test different values for the number of factor graphs K from the set $\{2^0, 2^1, 2^2, 2^3, 2^4\}$, corresponding embedding dimensions are $\{128, 64, 32, 16, 8\}$ per factor graph. We set three message-passing layers for our method and the hyperparameters $\lambda = 0.001$ and $\gamma = 0.001$. The training is conducted for 200 epochs across all methods. For all the traditional graph classification approaches, we compute similarities between graphs and feed them into a downstream SVM classifier to evaluate the classification accuracy. For MVGRL, GraphCL, JOVO, DGCL, and RGCL, we first pretrain the model with unlabeled graphs first and then fine-tune it with labeled graphs. For InfoGraph, GLA, TGNN, GraphSpa, and our proposed DisenSemi, we integrate the pretraining and fine-tuning phases together. The source code of DisenSemi can be found at: https://github.com/jamesyifan/DisenSemi.

B. Performance Comparison (RQ1)

1) Overall Comparison: Table I lists the quantitative results of semi-supervised graph classification using a 30% label ratio. From these results, we can draw the following conclusions.

1) Traditional graph classification approaches, which include graph kernel and unsupervised graph-level representation learning methods, perform worse than GNN-based semi-supervised graph classification methods. This indicates that hand-crafted features extracted by traditional methods suffer from poor generalization. Instead, GNN-based methods are more powerful to extract features from graph-structured data for the semi-supervised graph classification task.
2) Among GNN-based semi-supervised graph classification methods, DGCL and RGCL, which leverage disentangled representation for graph classification, always achieve better performance than other baselines on most datasets. It shows that explicitly considering the entanglement of factors for a graph helps to learn better representation for semi-supervised graph classification.
3) Our DisenSemi, outperforms all other models across the ten datasets, showcasing its superior effectiveness.

[1]https://ls11-www.cs.tu-dortmund.de/staff/morris/graphkerneldatasets and https://ogb.stanford.edu/docs/graphprop/

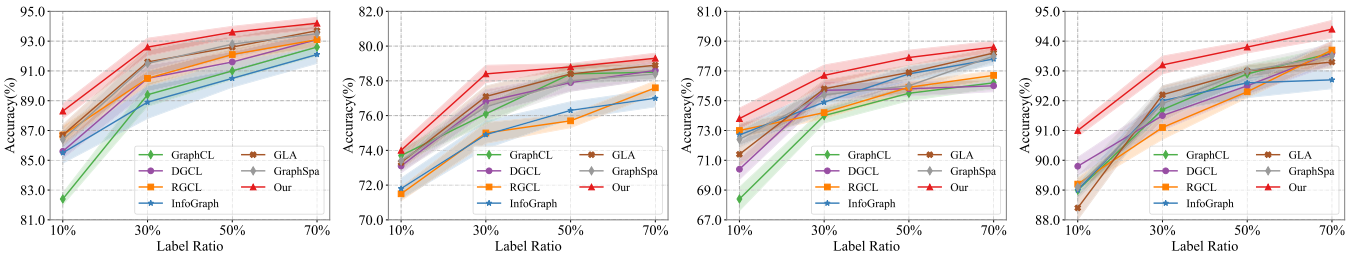


Fig. 3. Performance comparison with different labeling ratios (i.e., %10, %30, %50, and %70) on four datasets. (a) MUTAG. (b) PROTEINS. (c) IMDB-BINARY. (d) REDDIT-BINARY.

TABLE II

ABLATION STUDY ON GNN-BASED ENCODER WITH DIFFERENT TYPES OF MESSAGE-PASSING LAYER AND * MEANS RANDOMLY REMOVED 15% EDGES FROM THE ORIGINAL GRAPH TO OBTAIN DIFFERENT FACTOR GRAPHS

Methods	MUTAG	PTC-MR	NCI1	PROTEINS	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M5K	COLLAB	OGB-HIV
SAGEConv*	89.1 ± 0.4	68.4 ± 1.2	75.8 ± 0.5	74.7 ± 0.5	75.8 ± 0.8	51.9 ± 0.5	91.2 ± 0.4	56.5 ± 0.3	80.6 ± 0.4	95.6 ± 0.5
GATConv*	89.6 ± 0.3	69.2 ± 1.3	75.2 ± 0.6	74.1 ± 0.3	75.6 ± 0.8	51.6 ± 0.6	92.1 ± 0.4	56.8 ± 0.3	80.4 ± 0.3	95.8 ± 0.5
GINConv*	90.2 ± 0.5	68.9 ± 1.1	75.2 ± 0.4	74.7 ± 0.4	76.1 ± 0.9	52.2 ± 0.5	92.3 ± 0.3	56.9 ± 0.2	79.3 ± 0.2	96.0 ± 0.3
GraphConv*	90.1 ± 0.4	69.1 ± 1.0	75.3 ± 0.4	74.6 ± 0.5	76.0 ± 0.9	51.4 ± 0.6	92.1 ± 0.4	56.7 ± 0.3	80.9 ± 0.3	95.8 ± 0.4
GCNConv	90.6 ± 0.5	69.2 ± 1.0	75.5 ± 0.6	75.9 ± 0.5	76.0 ± 0.8	52.4 ± 0.6	92.4 ± 0.2	57.1 ± 0.3	81.2 ± 0.4	96.4 ± 0.3
GraphConv	$\mathbf{92.6 \pm 0.6}$	$\mathbf{72.4 \pm 1.1}$	$\mathbf{78.9 \pm 0.4}$	$\mathbf{78.4 \pm 0.5}$	$\mathbf{76.7 \pm 0.7}$	$\mathbf{52.5 \pm 0.5}$	$\mathbf{93.2 \pm 0.3}$	$\mathbf{57.6 \pm 0.2}$	$\mathbf{81.5 \pm 0.3}$	$\mathbf{97.2 \pm 0.2}$

The results indicate that the advancement beyond state-of-the-art methods stems not only from learning disentangled representation of the graph but also from explicitly regularizing the rationale that fits well between supervised and unsupervised learning models for semi-supervised graph classification.

2) Performance on Different Labeling Ratio: We vary the labeling ratio of training data to compare the performance of different models. Traditional methods are omitted due to their lack of competitive performance. Fig. 3 illustrates the performance w.r.t. different labeling ratios of data on four datasets. We observe that the following hold.

1) As the quantity of labeled graph data increases, the performance of all models tends to improve. This demonstrates that the labeling ratio of training data is an important factor of the model. Meanwhile, the growth is not linear, which means that unsupervised data also plays a key role to boost the model performance.
2) Disentangled graph representation model, i.e., DGCL and RGCL, though explicitly considering the entanglement of factors for graph, cannot always guarantee the improvement over other baselines among different labeling ratios. This is attributed to the fact that supervised and unsupervised learning tasks for graph-structured data have different optimization targets.
3) DisenSemi shows the superior performance compared with other baselines for varying labeling ratios of data. Especially when the labeled data are scarce (e.g., less than 70%), the superiority of our model indicates that disentangled graph representation with consistency regularization can effectively improve the semi-supervised classification performance.

C. Ablation Studies (RQ2)

To deeply understand our proposed DisenSemi, we conduct ablation studies over the key components of the model.

1) Effect of GNN-Based Encoder: In DisenSemi, we encode each factor graph via the proposed GNN with its own edge coefficient. To explore whether DisenSemi can derive benefits from the proposed GNN-based encoder, we compare five types of well-known GNN operators, namely, SAGEConv [54], GATConv [55], GINConv [56], GCNConv [57], and GraphConv [44]. Note that for SAGEConv*, GATConv*, and GINConv*, we ignore the edge coefficient and randomly remove 15% edges to obtain different factor graphs. For GraphConv*, we retain the edge coefficients and randomly remove edges separately. The performance of different GNN operators on all ten datasets is shown in Table II. From the results, we can draw the following conclusions.

1) In most cases, the performance of GCNConv is better than other variants, which ignore the edge coefficient and randomly remove edges to obtain different factor graphs (i.e., SAGEConv*, GATConv*, and GINConv*). This indicates that disentangling the graph with heterogeneous relations into different factor graphs can provide more effective and discriminative graph-level representation for the downstream classification task.
2) GraphConv* considers the high-order structure of the factor graph but still exhibits poor performance in cases where edges are randomly removed. Instead, GraphConv consistently outperforms other models when considering the edge coefficient. This outcome further validates the effectiveness of the learned edge coefficient within each factor graph for capturing higher order graph structures.

2) Effect of MI-Based Representation Disentanglement: As disentangled graph representation is implemented via MI

TABLE III

ABLATION STUDY ON MI-BASED REPRESENTATION AND DISENTANGLEMENT CONSISTENCY REGULARIZATION, W/O MEANS REMOVE THE MODULE FROM DISENSEMI AND VARIANT MEANS DIFFERENT DISTRIBUTION OF LATENT FACTORS

Methods	MUTAG	PTC-MR	NCI1	PROTEINS	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M5K	COLLAB	OGB-HIV
w/o Intra-MI	90.8 ± 0.5	68.6 ± 1.0	74.9 ± 0.8	75.9 ± 0.5	75.8 ± 0.8	51.4 ± 0.4	92.6 ± 0.2	56.5 ± 0.3	80.8 ± 0.4	95.9 ± 0.4
w/o Inter-MI	90.4 ± 0.5	69.2 ± 1.2	74.7 ± 0.6	76.2 ± 0.4	76.1 ± 0.7	52.3 ± 0.6	92.8 ± 0.3	56.2 ± 0.4	80.7 ± 0.5	96.2 ± 0.3
w/o MI	89.3 ± 0.4	68.3 ± 1.5	74.1 ± 0.7	73.9 ± 0.6	75.3 ± 0.8	51.9 ± 0.5	91.6 ± 0.3	56.1 ± 0.5	80.1 ± 0.4	95.6 ± 0.5
Variant 1	90.4 ± 0.4	69.8 ± 1.0	75.9 ± 0.6	74.1 ± 0.5	75.7 ± 0.8	51.8 ± 0.5	91.5 ± 0.2	56.3 ± 0.5	80.3 ± 0.3	96.2 ± 0.3
Variant 2	91.5 ± 0.4	68.6 ± 0.9	74.2 ± 0.5	74.5 ± 0.4	76.6 ± 0.7	52.3 ± 0.7	92.0 ± 0.2	56.4 ± 0.4	80.8 ± 0.4	96.5 ± 0.2
DisenSemi	$\mathbf{92.6 \pm 0.6}$	$\mathbf{72.4 \pm 1.1}$	$\mathbf{78.9 \pm 0.4}$	$\mathbf{78.4 \pm 0.5}$	$\mathbf{76.7 \pm 0.7}$	$\mathbf{52.5 \pm 0.5}$	$\mathbf{93.2 \pm 0.3}$	$\mathbf{57.6 \pm 0.2}$	$\mathbf{81.5 \pm 0.3}$	$\mathbf{97.2 \pm 0.2}$

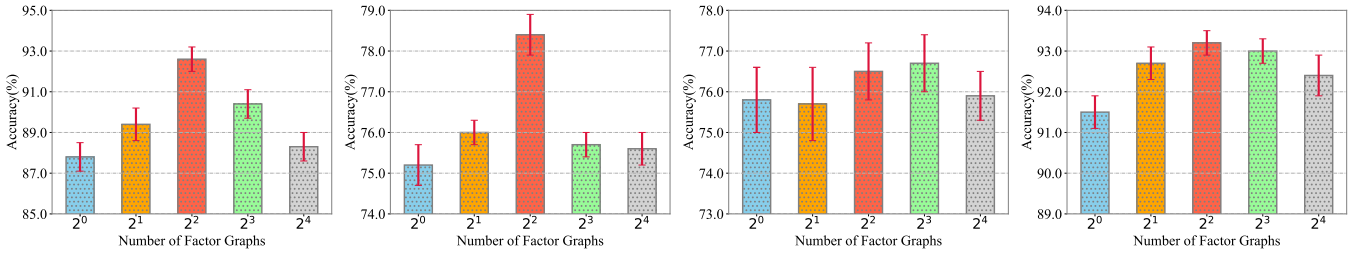


Fig. 4. Performance w.r.t. different numbers of factor graphs in four datasets. (a) MUTAG. (b) PROTEINS. (c) IMDB-BINARY. (d) REDDIT-BINARY.

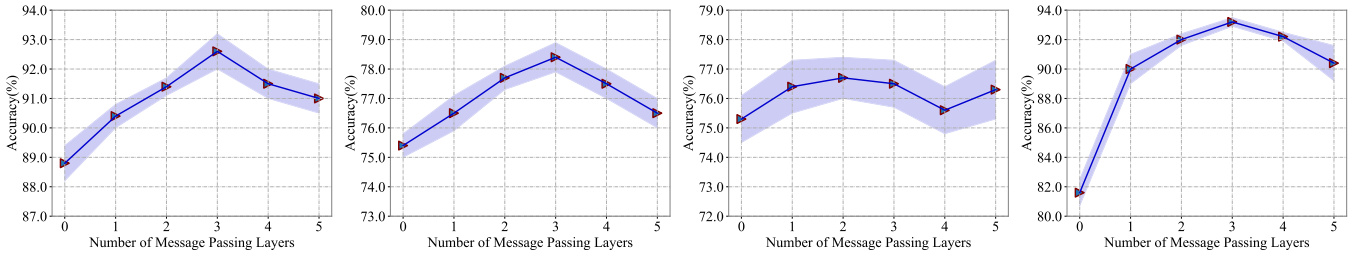


Fig. 5. Performance w.r.t. different numbers of message-passing layers in four datasets. (a) MUTAG. (b) PROTEINS. (c) IMDB-BINARY. (d) REDDIT-BINARY.

estimation in our DisenSemi, we investigate its impact on the performance with the following three variants.

1) *w/o Intra-MI:* It removes the intrafactor MI maximization on global–local pairs for unsupervised objective function (i.e., $L_U = L_{\text{inter}}$).
2) *w/o Inter-MI:* It removes the interfactor MI minimization between every two factor graphs for unsupervised objective function (i.e., $L_U = \sum_{k=1}^{K} L_{\text{intra}}^{k}$).
3) *w/o MI:* It removes the whole MI-based constraint for unsupervised objective function (i.e., w/o L_U).

Table III presents the results of different model variants on all ten datasets and we can observe that the best results have been attained by considering both intra- and inter-MI estimation, which indicates that both intrafactor MI maximization on global–local pairs and interfactor MI minimization between every two factor graphs play a key role to learn disentangled representation for semi-supervised graph classification.

3) Effect of Consistency Regularization: Disentangled consistency regularization helps to transfer the learned disentangled graph representations from the unsupervised encoder to the supervised encoder. To further investigate how the consistency regularization facilitates the model performance, we evaluate DisenSemi against the following two variants.

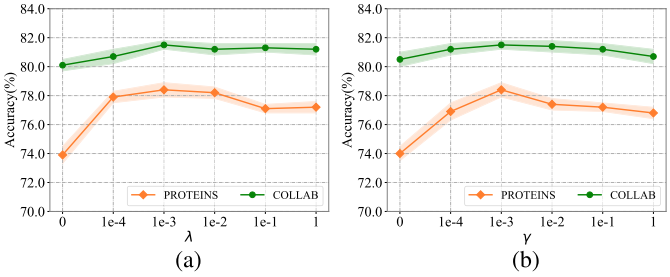
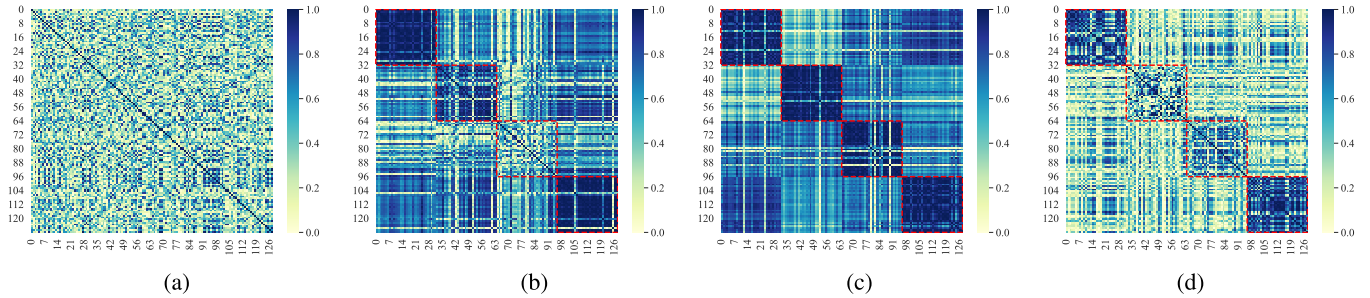


Fig. 6. Performance w.r.t. different relative weight between losses in PROTEINS and COLLAB. (a) Hyperparameter λ. (b) Hyperparameter γ.

1) *Variant 1:* It set $p(k|G_i) = 1/K$, a uniform distribution over K latent factors, which means that each kth latent factors reflected in G_i is same.
2) *Variant 2:* It set $p(k|G_i)$ with a random distribution over K latent factors. We implement it by randomly choosing a latent factor reflected in G_i.

Table III displays the outcomes for DisenSemi and its variants. The observations are as follows.



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

Fig. 7. Representation correlation analysis, where representations are obtained from the test split in MUTAG and PROTEINS. We use the fine-tuned model of GraphCL and DGCL and the supervised model of our proposed DisenSemi to get the representation. (a) GraphCL (MUTAG). (b) DGCL (MUTAG). (c) DisenSemi (MUTAG). (d) DisenSemi (PROTEINS).

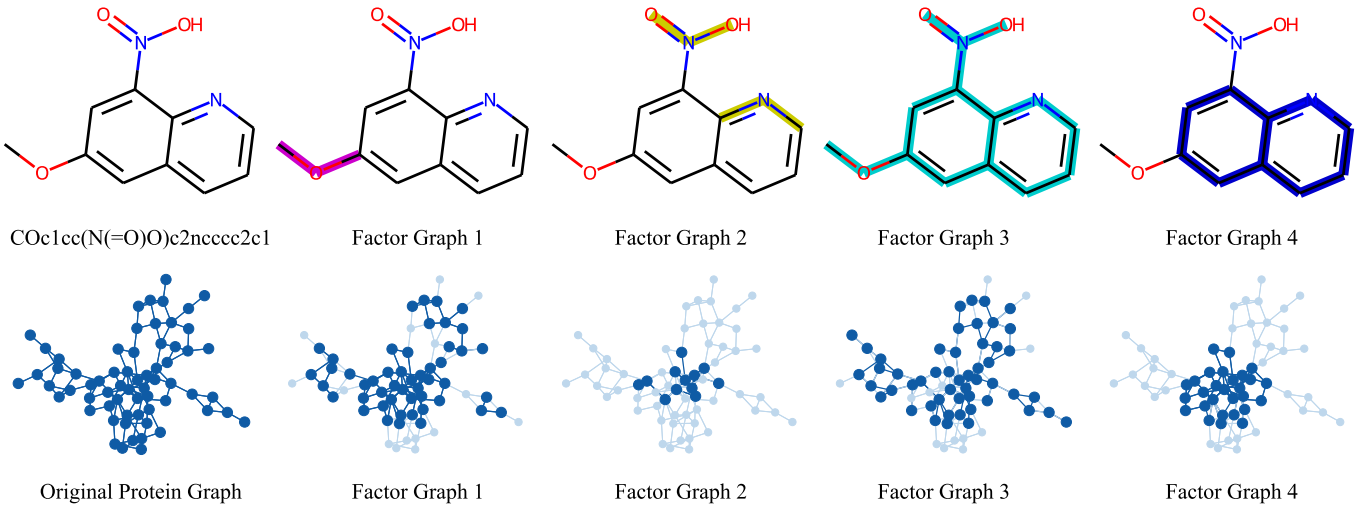


Fig. 8. Visualization of disentangled factor graph in MUTAG and PROTEINS. The first column presents the original graph and the second-to-fifth columns present those factor graphs highlighted with different colors.

1) The performance decreases when $p(k|G_i)$ of the model are set with uniform and random distribution in Variant 1 and Variant 2, respectively, which demonstrates that inferring the latent factor of a graph to explicitly identify and regularize the rationale between supervised and unsupervised learning model is important.
2) Variant 1 performs worse than variant 2 in most datasets. This may be that setting $p(k|G_i)$ with a random distribution can also specify a rationale between supervised and unsupervised learning models. Instead, inferring the latent factor of a graph uniformly might introduce rationale's complement to the representation learning.

D. Parameter Sensitivity (RQ3)

We also analyze how the proposed DisenSemi responds to different hyperparameter settings.

1) Effect of the Number of Factor Graphs: To assess the advantages of disentangled representation for DisenSemi, we evaluate the model's performance with different quantities of factor graphs. Specifically, we explore the number of factor graphs in the set $\{2^0, 2^1, \ldots, 2^4\}$. Fig. 4 presents a summary of the experimental outcomes across four datasets, revealing the following insights.

1) When the number of factor graphs $K = 1$, the model can be degraded into an entangled representation-based semi-supervised graph classification model with poor performance. This indicates that explicitly modeling multiple aspects of discriminant features for the graph can greatly facilitate the model performance.
2) Increasing the number of factor graphs can substantially enhance the model performance. DisenSemi achieves the best performance at $K = 4$ in MUTAG, PROTEINS, and REDDIT-B and $K = 8$ in IMDB-B, which represents the optimal aspects of discriminant features for the graph.
3) Nonetheless, if the number of factor graphs is excessively large (e.g., $K \geq 16$), the model's performance tends to decline gradually. This degradation may result from employing an overly complex semantic structure for the graph.

2) Effect of the Number of Message-Passing Layers: To examine the impact of depth on performance for each disentangled factor graph, we test the DisenSemi model with a range of message-passing layers from [0, 5]. Fig. 5 illustrates the results of these experiments across four datasets—MUTAG, PROTEINS, IMDB-BINARY, and REDDIT-BINARY. The findings are as follows.

1) When the number of message-passing layers $L = 0$, the model only takes a linear transformation into consideration and suffers from the degenerating issue. Hence, the result verifies the rationality and effectiveness of GNNs for learning graph-level representation.

2) Increasing the number of message-passing layers incorporates information from higher order neighbors and deepens the model. Evidently, a model with two layers performs better than one with only one layer. It illustrates the importance of messages passing between nodes to capture the structure of each factor graph.
3) Too many message-passing layers may hurt the model performance. When stacking more than three message-passing layers, the model may introduce noise and suffer from the over-smoothing issue.

3) Effect of Relative Weight Between Losses: We investigate the sensitivity of relative weight between losses, namely, λ and γ. In particular, we adjust the value of λ and γ from 0 to 1 on PROTEINS and COLLAB. Fig. 6 demonstrates that both the unsupervised module and disentangled consistency regularization play an important role in the objective, and there is an evident performance drop when $\lambda, \gamma = 0$. As λ and γ get larger, the performance is relatively stable, and we can get the best results when $\lambda, \gamma = 1e - 3$.

E. Visualization and Case Study (RQ4)

To further investigate how the disentangled representation facilitates the semi-supervised graph classification task, we conduct two qualitative assessments (visualization and case study) of the proposed DisenSemi and baselines.

1) Visualization of Representation Correlation: Besides the quantitative evaluation, we also visualize the absolute correlations between the components of the 128-D graph representations. Fig. 7 shows the correlations analysis of representation obtained from GraphCL, DGCL, and our proposed DisenSemi with four factor graphs in the MUTAG and PROTEINS datasets. We find that the following hold.

1) Compared with the more highly independent representation of DGCL, the learned representation of GraphCL is entangled, and the performance is degraded. This indicates that disentangled representation can achieve high performance in the graph classification task.
2) The representation produced by DisenSemi demonstrates a distinct blockwise correlation pattern, suggesting that the four factor graphs of DisenSemi are likely capturing mutually exclusive information. This allows for the extraction of discriminative features for the target task.

2) Case Study: To illustrate the disentanglement process more clearly, we present examples of the factor graphs generated by DisenSemi. Fig. 8 shows the original graph alongside the disentangled factor graphs. The edges of the visualized disentangled factor graphs are highlighted with different colors after we set the coefficient threshold in the original graph. For example, in the MUTAG dataset, the task is to predict the molecule's mutagenicity on Salmonella typhimurium for a collection of nitroaromatic compounds. We can see different parts of a molecule graph playing different roles in prediction. This also justifies the reliability of our generated factor graphs in getting the disentangled graph representation.

VI. Conclusion

This article explores the problem of semi-supervised graph classification, a core problem in the field of graph-structured data analysis. For transferring suitable knowledge from the unsupervised model to the supervised model, we propose a novel framework termed DisenSemi, which learns disentangled representation to capture multiple graph characteristics stemming from different aspects. Specifically, our DisenSemi consists of a supervised model and an unsupervised model. For both models, we design a disentangled graph encoder to extract factorwise graph representation and train two models with supervised objective and MI-based constraints, respectively. Then, we propose an MI-based disentangled consistency regularization to identify the rationale that aligns well between two models for the current graph classification task and transfer corresponding knowledge semantically. Extensive experiments on ten benchmark graph classification datasets demonstrate the efficacy of our DisenSemi. However, the number of factor graphs varies among datasets and needs to be searched to find the best value. In future research, we plan to extend our DisenSemi to more complex semi-supervised classification scenarios and automatically select factor graph numbers in a bilevel optimization framework.

Acknowledgement

The authors are grateful to the anonymous reviewers for critically reading the manuscript and for giving important suggestions to improve their paper.

References

[1] K. M. Borgwardt, C. S. Ong, S. Schonauer, S. V. N. Vishwanathan, A. J. Smola, and H.-P. Kriegel, "Protein function prediction via graph kernels," *Bioinformatics*, vol. 21, no. 1, pp. i47–i56, Jun. 2005.
[2] Z. Hao et al., "ASGN: An active semi-supervised graph neural network for molecular property prediction," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 731–752.
[3] Y. Wang et al., "DisenCite: Graph-based disentangled representation learning for context-specific citation generation," in *Proc. 36th AAAI Conf. Artif. Intell.*, 2022, pp. 11449–11458.
[4] L. Backstrom and J. Leskovec, "Supervised random walks: Predicting and recommending links in social networks," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, Feb. 2011, pp. 635–644.
[5] Y. Wang, S. Tang, Y. Lei, W. Song, S. Wang, and M. Zhang, "DisenHAN: Disentangled heterogeneous graph attention network for recommendation," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2020, pp. 1605–1614.
[6] H. Linmei, T. Yang, C. Shi, H. Ji, and X. Li, "Heterogeneous graph attention networks for semi-supervised short text classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 4821–4830.
[7] K. M. Borgwardt and H. Kriegel, "Shortest-path kernels on graphs," in *Proc. 5th IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2005, pp. 1–8.
[8] N. Shervashidze, P. Schweitzer, E. J. Van Leeuwen, K. Mehlhorn, and K. M. Borgwardt, "Weisfeiler–Lehman graph kernels," *JMLR*, vol. 12, no. 9, pp. 2539–2561, 2011.
[9] R. Kondor and H. Pan, "The multiscale Laplacian graph kernel," in *Proc. NeurIPS*, 2016, pp. 2990–2998.
[10] N. Shervashidze, S. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt, "Efficient graphlet kernels for large graph comparison," in *Proc. AISTATS*, 2009, pp. 488–495.
[11] W. Ju, X. Luo, Z. Ma, J. Yang, M. Deng, and M. Zhang, "GHNN: Graph harmonic neural networks for semi-supervised graph-level classification," *Neural Netw.*, vol. 151, pp. 70–79, Jul. 2022.
[12] W. Ju, J. Yang, M. Qu, W. Song, J. Shen, and M. Zhang, "KGNN: Harnessing kernel-based networks for semi-supervised graph classification," in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 421–429.

[13] X. Luo et al., "DualGraph: Improving semi-supervised graph classification via dual contrastive learning," in *Proc. IEEE 38th Int. Conf. Data Eng. (ICDE)*, May 2022, pp. 699–712.
[14] H. Yue, C. Zhang, C. Zhang, and H. Liu, "Label-invariant augmentation for semi-supervised graph classification," in *Proc. NeurIPS*, 2022, pp. 29350–29361.
[15] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen, "Graph contrastive learning with augmentations," in *Proc. NeurIPS*, vol. 33, 2020, pp. 5812–5823.
[16] F.-Y. Sun, J. Hoffmann, V. Verma, and J. Tang, "InfoGraph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization," in *Proc. ICLR*, 2020, pp. 1–16.
[17] Y. You, T. Chen, Y. Shen, and Z. Wang, "Graph contrastive learning automated," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 12121–12132.
[18] W. Ju et al., "TGNN: A joint semi-supervised framework for graph-level classification," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 2122–2128.
[19] W. Ju et al., "Focus on informative graphs! Semi-supervised active learning for graph-level classification," *Pattern Recognit.*, vol. 153, Sep. 2024, Art. no. 110567.
[20] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 4805–4815.
[21] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[22] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[23] Y. Wang et al., "DisenCTR: Dynamic graph-based disentangled representation for click-through rate prediction," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2022, pp. 2314–2318.
[24] J. Ma, P. Cui, K. Kuang, X. Wang, and W. Zhu, "Disentangled graph convolutional networks," in *Proc. ICML*, 2019, pp. 4212–4221.
[25] Y. Yang, Z. Feng, M. Song, and X. Wang, "Factorizable graph convolutional networks," in *Proc. NeurIPS*, 2020, pp. 20286–20296.
[26] H. Li, X. Wang, Z. Zhang, Z. Yuan, H. Li, and W. Zhu, "Disentangled contrastive learning on graphs," in *Proc. NeurIPS*, 2021, pp. 21872–21884.
[27] Y. Mo, Y. Lei, J. Shen, X. Shi, H. T. Shen, and X. Zhu, "Disentangled multiplex graph representation learning," in *ICML*, 2023, pp. 24983–25005.
[28] M. Sajjadi, M. Javanmardi, and T. Tasdizen, "Regularization with stochastic transformations and perturbations for deep semi-supervised learning," in *Proc. NeurIPS*, 2016, pp. 1163–1171.
[29] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *Proc. ICLR*, 2017, pp. 1–13.
[30] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. NeurIPS*, 2017, pp. 1195–1204.
[31] K. Sohn et al., "FixMatch: Simplifying semi-supervised learning with consistency and confidence," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2020, pp. 596–608.
[32] B. Zhang et al., "FlexMatch: Boosting semi-supervised learning with curriculum pseudo labeling," in *Proc. NeurIPS*, 2021, pp. 18408–18419.
[33] J. Li, Y. Rong, H. Cheng, H. Meng, W. Huang, and J. Huang, "Semi-supervised graph classification: A hierarchical graph perspective," in *Proc. World Wide Web Conf.*, May 2019, pp. 972–982.
[34] X. Luo, Y. Zhao, Y. Qin, W. Ju, and M. Zhang, "Towards semi-supervised universal graph classification," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 1, pp. 416–428, Jan. 2024.
[35] H. Li, Y. Li, and H. Lu, "Semi-supervised learning with Gaussian processes," in *Proc. Chin. Conf. Pattern Recognit.*, Oct. 2008, pp. 1690–1701.
[36] I. Higgins et al., "β-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. ICLR*, 2016, pp. 1–22.
[37] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Dec. 2016, pp. 2172–2180.
[38] P. Cheng et al., "Improving disentangled text representation learning with information-theoretic guidance," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 7530–7541.
[39] Y. Qin et al., "DisenPOI: Disentangling sequential and geographical influence for point-of-interest recommendation," in *Proc. 16th ACM Int. Conf. Web Search Data Mining*, Feb. 2023, pp. 508–516.
[40] P. Velickovic, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, "Deep graph infomax," in *Proc. ICLR*, 2019, pp. 1–17.
[41] M. I. Belghazi et al., "Mutual information neural estimation," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 531–540.
[42] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.
[43] M. Gutmann and A. Hyvarinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 297–304.
[44] C. Morris et al., "Weisfeiler and Leman go neural: Higher-order graph neural networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 4602–4609.
[45] 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.
[46] P. Cheng, W. Hao, S. Dai, J. Liu, Z. Gan, and L. Carin, "CLUB: A contrastive log-ratio upper bound of mutual information," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1779–1788.
[47] X. Liang, D. Li, and A. Madden, "Attributed network embedding based on mutual information estimation," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 835–844.
[48] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, vol. 119, 2020, pp. 1597–1607.
[49] P. Yanardag and S. V. N. Vishwanathan, "Deep graph kernels," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2015, pp. 1365–1374.
[50] B. Adhikari, Y. Zhang, N. Ramakrishnan, and B. A. Prakash, "Sub2Vec: Feature learning for subgraphs," in *Proc. PAKDD*, 2018, pp. 170–182.
[51] A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu, and S. Jaiswal, "Graph2Vec: Learning distributed representations of graphs," 2017, *arXiv:1707.05005*.
[52] K. Hassani and A. H. Khasahmadi, "Contrastive multi-view representation learning on graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 4116–4126.
[53] 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.
[54] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 1025–1035.
[55] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. ICLR*, 2018, pp. 1–12.
[56] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *Proc. ICLR*, 2018, pp. 1–17.
[57] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. ICLR*, 2017, pp. 1–14.

Yifan Wang received the B.S. and M.S. degrees in software engineering from Northeastern University, Shenyang, China, in 2014 and 2017, respectively, and the Ph.D. degree in computer science from Peking University, Beijing, China, in 2023.

He is currently an Assistant Professor with the School of Information Technology and Management, University of International Business and Economics, Beijing. His research interests include graph representation learning, graph neural networks (GNNs), disentangled representation learning, and corresponding applications, such as drug discovery and recommender systems.

Xiao Luo received the B.S. degree in mathematics from Nanjing University, Nanjing, China, in 2017, and the Ph.D. degree from the School of Mathematical Sciences, Peking University, Beijing, China, in 2022.

He is currently a Post-Doctoral Researcher with the Department of Computer Science, University of California at Los Angeles, Los Angeles, CA, USA. His research interests include machine learning on graphs, image retrieval, statistical models, and bioinformatics.

Chong Chen (Member, IEEE) received the B.S. degree in mathematics and the Ph.D. degree in statistics, under the supervision of Prof. Ruibin Xi, from Peking University, Beijing, China, in 2013 and 2019, respectively.

He is currently a Research Scientist with the Terminus Group, Beijing. His research interests include image understanding, self-supervised learning, and data mining.

Xian-Sheng Hua (Fellow, IEEE) received the B.S. and Ph.D. degrees in applied mathematics from Peking University, Beijing, China, in 1996 and 2001, respectively.

In 2001, he joined Microsoft Research Asia, Beijing, as a Researcher. He became a Researcher and the Senior Director of the Alibaba Group, Hangzhou, China, in 2015. He has been a Senior Researcher at Microsoft Research Redmond, Redmond, WA, USA, since 2013. He has authored or co-authored over 250 research articles and has filed over 90 patents. His research interests include multimedia search, advertising, understanding, mining, pattern recognition, and machine learning.

Dr. Hua was honored as one of the recipients of MIT Technology Review Innovators Under 35 Asia Pacific (MIT35). He served as a Program Co-Chair for the IEEE International Conference on Multimedia and Expo (ICME) 2013, the Association for Computing Machinery (ACM) Multimedia 2012, and the IEEE ICME 2012, and on the Technical Directions Board for the IEEE Signal Processing Society. He is a Distinguished Scientist of ACM.

Ming Zhang (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science from Peking University, Beijing, China, in 1988, 1991, and 2005, respectively.

She is currently a Full Professor at the School of Computer Science, Peking University. She has authored or co-authored more than 200 research articles on text mining and machine learning in the top journals and conferences.

Prof. Zhang is a member of the Advisory Committee of Ministry of Education (MOE) in China and the Chair of Association for Computing Machinery (ACM) Special Interest Group on Computer Science Education (SIGCSE) China. She is one of the 15 members of the ACM/IEEE CC2020 Steering Committee. She won the Best Paper of International Conference on Machine Learning (ICML) 2014 and Best Paper Nominee of the Web Conference (WWW) 2016. She is the leading author of several textbooks on data structures and algorithms in Chinese, and the corresponding course has been awarded as the National Elaborate Course, National Boutique Resource Sharing Course, National Fine-designed Online Course, and National First-Class Undergraduate Course by MOE China.

Wei Ju (Member, IEEE) received the B.S. degree in mathematics from Sichuan University, Chengdu, China, in 2017, and the Ph.D. degree from the School of Computer Science, Peking University, Beijing, China, in 2022.

He worked as a Post-Doctoral Research Fellow with Peking University. He is currently an Associate Professor with the College of Computer Science, Sichuan University. He has authored or co-authored more than 40 articles in top-tier venues. His current research interests lie primarily in the area of machine learning on graphs including graph representation learning and graph neural networks (GNNs), and interdisciplinary applications, such as recommender systems, bioinformatics, drug discovery, and knowledge graphs.

Dr. Ju has won the Best Paper Finalist in IEEE International Conference on Data Mining (ICDM) 2022.