The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)

Dual Label-Guided Graph Refinement for Multi-View Graph Clustering

Yawen Ling[1]*, Jianpeng Chen[1]*, Yazhou Ren[1, 2]†, Xiaorong Pu[1, 2], Jie Xu[1], Xiaofeng Zhu[1, 2], Lifang He[3]

[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China
[2]Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China
[3]Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA, USA
{yawen.Ling, cjpcool, jiexuwork}@outlook.com, {yazhou.ren, puxiaor}@uestc.edu.cn,
seanzhuxf@gmail.com, lih319@lehigh.edu

Abstract

With the increase of multi-view graph data, multi-view graph clustering (MVGC) that can discover the hidden clusters without label supervision has attracted growing attention from researchers. Existing MVGC methods are often sensitive to the given graphs, especially influenced by the low quality graphs, *i.e.*, they tend to be limited by the homophily assumption. However, the widespread real-world data hardly satisfy the homophily assumption. This gap limits the performance of existing MVGC methods on low homophilous graphs. To mitigate this limitation, our motivation is to extract high-level view-common information which is used to refine each view's graph, and reduce the influence of non-homophilous edges. To this end, we propose dual label-guided graph refinement for multi-view graph clustering (DuaLGR), to alleviate the vulnerability in facing low homophilous graphs. Specifically, DuaLGR consists of two modules named dual label-guided graph refinement module and graph encoder module. The first module is designed to extract the soft label from node features and graphs, and then learn a refinement matrix. In cooperation with the pseudo label from the second module, these graphs are refined and aggregated adaptively with different orders. Subsequently, a consensus graph can be generated in the guidance of the pseudo label. Finally, the graph encoder module encodes the consensus graph along with node features to produce the high-level pseudo label for iteratively clustering. The experimental results show the superior performance on coping with low homophilous graph data. The source code for DuaLGR is available at https://github.com/YwL-zhufeng/DuaLGR.

Datasets	Graphs	Edges	Homo-edges	HR
ACM	$\mathcal{G}^1$	26256	21550	0.82
	$\mathcal{G}^2$	2207736	1411658	0.64
DBLP	$\mathcal{G}^1$	7056	5636	0.80
	$\mathcal{G}^2$	4996438	3346042	0.67
	$\mathcal{G}^3$	6772278	2183134	**0.32**
Texas	$\mathcal{G}^1$	574	50	**0.09**
Chameleon	$\mathcal{G}^1$	62792	14476	**0.23**

Table 1: The description about edges and homophily ratio (HR) of four graph datasets. $\mathcal{G}$ represents different views of each dataset; the third column represents the number of all edges in specific view without self-loop; the fourth column means the number of homophilous edges; HR denotes the homophily ratio defined as homo-edges / edges.

Introduction

As an important task in unsupervised learning, multi-view clustering (MVC) keeps attracting attention in the past decade. The goal of MVC is to partition the objects into different classes by exploiting on the consistency and complementarity across multiple views. Generally, the conventional MVC methods can be mainly divided into two categories: multi-view subspace clustering and multi-view graph-based clustering. The former is dedicated to learning a consensus subspace representation through the learning and comparison of different views (Yang et al. 2022; Huang et al. 2021; Tan et al. 2021; Xu et al. 2021). The latter usually focuses on mining graph structure information in features and obtaining assignment results via Laplace matrix eigendecomposition (Liang et al. 2022; Qiang et al. 2021; Zhong and Pun 2022). In recent years, MVC methods achieve impressive performance. However, they are not competent in dealing with graph data with both conventional features and graph structure information as they tend to only focus on the feature or graph information while ignoring the other one.

Graph data are rich in structural information, thanks to which graph neural networks (GNNs) have achieved impressive success (Kipf and Welling 2016a; Cotta, Morris, and Ribeiro 2021; Zhang et al. 2022; Tse et al. 2022). Considering that GNNs can simultaneously exploit the information implied in features and graphs, some works attempt to employ GNNs on multi-view graph clustering (MVGC), such as O2MAC (Fan et al. 2020), MAGCN (Cheng et al. 2020), MVGRL (Hassani and Khasahmadi 2020). The development of these methods has brought a breakthrough in MVGC. However, the GNNs on which these methods rely are based on the homophily assumption, *i.e.*, edges tend to connect similar nodes. The graph data in real-world

*These authors contributed equally.

†Corresponding author.

scenarios often cannot fully satisfy this assumption, and there may be a large number of non-homophilous edges that connect the nodes from different classes, violating the homophily assumption. For example, Table 1 lists the number of edges, homophilous edges (Homo-edges) and homophily ratio (HR) of four widely used graph datasets. As shown, these graphs are filled with great numbers of non-homophilous edges, especially the $\mathcal{G}^3$ of DBLP (HR 0.32), Texas (HR 0.09) and Chameleon (HR 0.23). Due to the aggregation operation of GNNs, these noisy edges keep aggregating feature representations from different classes, which will have cascading effects with iterations, and thus affect the distinguishability of clusters (Zhu et al. 2021). In such a scenario, these homophily assumption based MVGC approaches may be ineffective.

To alleviate the limitations of existing MVGC methods in low homophilous graphs, we propose a dual label-guided graph refinement framework for multi-view graph clustering (named DuaLGR). With the absence of labels, we try to make full use of the extracted high-level class information, *i.e.*, soft label and pseudo label. In specific, DuaLGR consists of two modules, *i.e.*, dual label-guided graph refinement module and graph encoder module. On basis of the soft label extracted from a pretrained autoencoder, a refinement matrix can be learned. In cooperation with the pseudo label from the followed graph encoder module, the graph of each view can be aggregated adaptively and then refined by the refinement matrix. With the guidance of the pseudo label which carries the global information, the graphs are fused to generate a consensus graph. Finally, the graph encoder module is responsible for encoding the graphs along with node features, and evaluating the predictions for each view. In this way, the dual label-guided graph refinement module helps the graph encoder module to encode node features along with the global consensus graph. Thus, more distinguishable representations can be generated for clustering, and the graph encoder module in turn facilitates the dual label-guided graph refinement module for learning a better refinement matrix and generating a better consensus graph. To summarize, our main contributions are listed as follows:

- We propose a dual label-guided graph refinement framework for multi-view graph clustering task in low homophilous graphs. In the framework, the extracted soft label and pseudo label are fully exploited to learn a refinement matrix to refine the graphs, so the negative influence of non-homophilous edges is mitigated.

- We propose to use the pseudo label that carries global high-level class information to adjust the aggregation of graphs adaptively and to evaluate the weight of each view, and thus a consensus graph is generated for clustering.

- Experiments on widely used homophilous and low homophilous graph datasets demonstrate that DuaLGR has competitive performance with SOTAs, and it can better deal with the low homophilous data.

Related Works

In order to explore the feature information and graph structural information simultaneously, many methods seek help from GNNs to solve the MVGC tasks. O2MGC is the first work that attempts to employ the novel GNN for MVGC, which encodes the attributed multi-view graphs to a low-dimensional space by utilizing single-view graph convolutional encoder and multi-view graph structure decoder (Fan et al. 2020). Subsequently, Cheng et al. design two-pathway graph encoders to map graph embedding features and learn view-consistency information (Cheng et al. 2020). Hassani and Khasahmadi introduce a GNN solution to learn node and graph level representations for multi-view self-supervised learning (Hassani and Khasahmadi 2020). Lin and Kang apply a graph filter to smooth the features and learn a consensus graph for clustering (Lin and Kang 2021). Pan and Kang utilize contrastive learning to dig out the shared geometry and semantics for learning a consensus graph (Pan and Kang 2021). Despite the attractive performance of these methods, they are often sensitive to the quality of graph structure. In other words, low homophilous graphs may lead to unsatisfactory performance when they are ignored in model design.

In recent years, some efforts have been extended to deal with the low homophilous graphs (Lim et al. 2021; Suresh et al. 2021). Chien et al. address heterophily and over-smoothing by propagating with special learnable weight in GNN (Chien et al. 2021). Wang et al. introduce two measurements of homophily degree to change the propagation and aggregation process adaptively (Wang et al. 2022). He et al. introduce block modeling to implement automatically learning of the corresponding aggregation rules for neighbors of different classes (He et al. 2022). These improvements, to some extent, have alleviated the problem of low homophilous graphs. However, all these methods require label information when training, which is not applicable to unsupervised tasks. In this work, we propose DuaLGR to solve the difficulty of MVGC in low homophilous graphs. DuaLGR could extract dual label information in unsupervised settings to learn a refinement matrix to refine the graphs and generate a consensus graph for clustering.

Methodology

Notations. Given a multi-view graph data $G(\mathbf{X}, \mathbf{A}^v)$ with shared feature $\mathbf{X} \in \mathbb{R}^{n \times d}$, the undirected adjacent matrix $\mathbf{A}^v = \{a_{ij}^v | a_{ij}^v \in \{0, 1\}\}$ from v-th view represents the relationships between nodes. Specifically, $a_{ij} = 1$ means there exists an edge between node i and node j. For an MVGC task, its objective is to divide the n nodes into c classes. In this work, $\mathbf{A}^v$ is normalized as $\widetilde{\mathbf{A}}^v = (\mathbf{D}^v)^{-1}\mathbf{A}^v$, where diagonal matrix $\mathbf{D}_{ii}^v = \sum_j a_{ij}^v$ represents the degree matrix.

Motivation and Overview

Facing graph data with numerous noisy edges, *i.e.*, non-homophilous edges, an intuitive idea is to refine the graphs so as to highlight the role of homophilous edges and reduce the influence of non-homophilous edges. However, in MVGC task, how to refine the graphs without the label information is challenging.

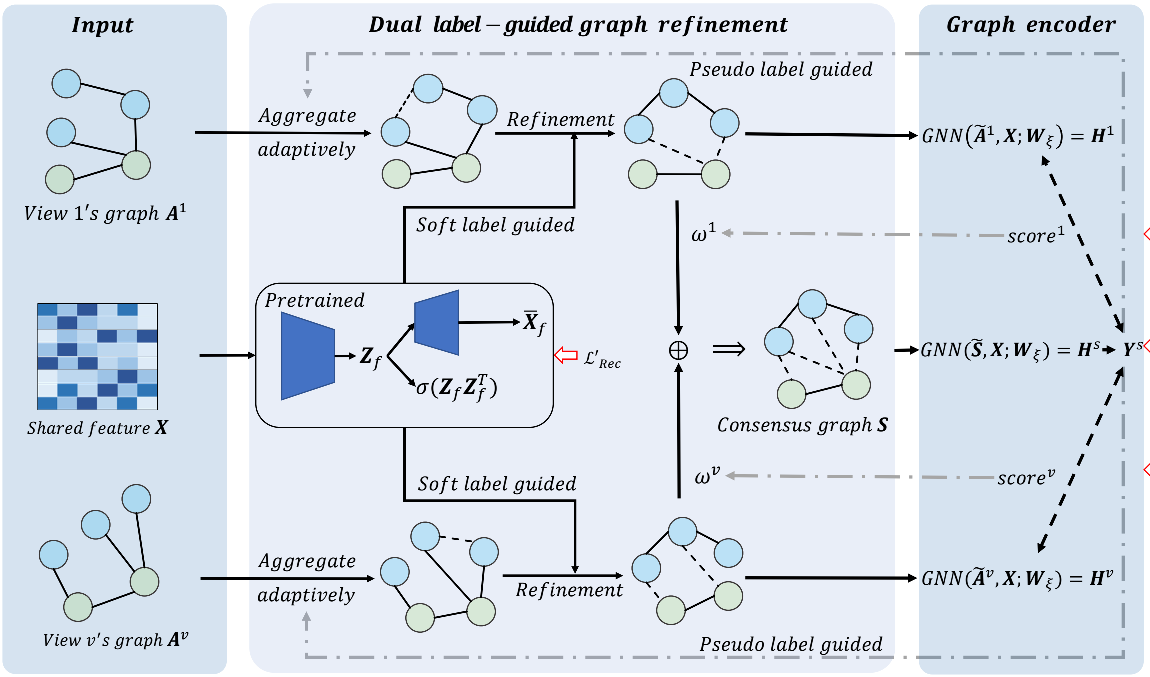

Figure 1: The illustration of DuaLGR. It consists of two main modules: dual label-guided graph refinement and graph encoder. With the guidance of dual labels, the first module refines the multiple high-order $\overline{\mathbf{A}}^v$ of each view by learning a refinement matrix, and then generates a global graph $\mathbf{S}$ through a weighted summation operation. The second module encodes the graph and features together to produce pseudo label. After finishing training, the second module also obtains the final predictions.

Specifically, since no true label is provided, it is difficult to get guidance information from given unreliable graphs to refine the graphs themselves. To address this problem, firstly, we try to extract the high-level soft label containing more accurate and reliable consensus information from shared features and multiple graphs. The intuition comes from that the view-common information must be more reliable. Therefore, the soft label can be used to guide the process of each specific graph refinement.

On the other hand, the forward propagation of our graph encoding module can well integrate feature with multiple graphs together, and extract label information from them. So, the final predictions can be regarded as pseudo label, and they can be used to guide the update and the fusion of each refined graph. To sum up, the proposed refinement process is guided by dual labels (soft label and pseudo label), generating a better consensus graph, and this in turn leads to better label information. In this way, the model is optimized alternatively. The overall process of proposed DuaLGR is shown in Figure 1.

Soft Label Guided Graph Refinement

Soft label learning. There are abundant class information and view-common information implied in node features and graphs, and these information can be mined to generate a refinement matrix. Given the excellent information extraction capability of autoencoder (Hinton and Salakhutdinov 2006), we first pretrain an autoencoder to extract the high-level semantic information, *i.e.*, soft label, from the shared node features $\mathbf{X}$ and graphs $\mathbf{A}^v$ of different views. Specially, the encoder $\mathbf{Z}_f = f(\sigma(\mathbf{X}; \mathbf{W}_\theta))$ and decoder $\overline{\mathbf{X}}_f = g(\sigma(\mathbf{Z}_f; \mathbf{W}_\phi))$ are used to encode and reconstruct the shared feature $\mathbf{X}$, where $\mathbf{Z}_f \in \mathbb{R}^{n\times c}$, c is the number of classes, $\mathbf{W}_\theta$ and $\mathbf{W}_\phi$ are learnable parameters for $f(\cdot)$ and $g(\cdot)$, and $\sigma(\cdot)$ is activation function.

In order to enforce the encoder to mine more common semantics across all views simultaneously, we define the autoencoder based reconstruction loss as:

$$\mathcal{L}'_{Rec} = l(\overline{\mathbf{X}}_f; \mathbf{X}) + \sum_{v=1}^{V} l(\sigma(\mathbf{Z}_f \mathbf{Z}_f^T); \mathbf{A}^v), \qquad (1)$$

where $l(\cdot;\cdot)$ is loss function, and $\sigma(\cdot)$ is activation function, instantiated as cross entropy and sigmoid here, respectively.

Soft label guided graph refinement. We consciously compress the shared feature $\mathbf{X}$ from d-dimension to c-dimension, thus the learned soft label $\mathbf{Z}_f$ summarizes both the class information of node features and common information from all views with the constraint of $\mathcal{L}'_{Rec}$. Considering the $\mathbf{Z}_f$ cannot directly guide the refinement of graphs, we calculate a refinement matrix by it as follows:

$$\mathbf{\Omega} = \mathbf{Z}_f \mathbf{Z}_f^T. \qquad (2)$$

The obtained $\mathbf{\Omega}$ is a matrix that describes the similarity of nodes. In other words, it indicates the intensity of the existence of an edge between two nodes. Since the soft label $\mathbf{Z}_f$ contains the feature information from node features and the first-order structural consensus information from different graphs, the obtained $\mathbf{\Omega}$ can provide guidance for the refinement of low homophilous graphs. In this work, we first quantize the dense $\mathbf{\Omega}$ and then define the refinement of graphs under the guidance of soft label as:

$$\overline{\mathbf{A}}^v = \alpha \mathbf{A}^v + \mathbf{\Omega}, \tag{3}$$

where α is a hyperparameter that controls the influence of the homophily of the graph acorss different views.

Intuitively, since $\mathbf{\Omega}$ contains class information and global common information, it can bring global complementary information to each graph. When a particular graph is a low homophilous graph, $\mathbf{\Omega}$ can reinforce or generate homophilous edges; when this particular graph is a homophilous graph, $\mathbf{\Omega}$ naturally reinforces its homophily.

Pseudo Label Guided Graph Refinement and Graph Fusion

Pseudo label guided graph refinement. $\mathbf{A}^v$ in Eq. (3) is defined as the first-order neighborhood relationship. Moreover, high-order graph structure can capture more relationships (Chen et al. 2020), as different orders of $\mathbf{A}^v$ contain different structural information. Therefore, we consider the multiple neighborhood relationship as:

$$\mathbf{A}_k^v = \frac{1}{k}\sum_{i=1}^{k}(\mathbf{A}^v)^i = \frac{1}{k}(\mathbf{A}^v + (\mathbf{A}^v)^2 + ... + (\mathbf{A}^v)^k), \tag{4}$$

where k denotes the order of neighborhood relationship aggregation.

However, the HR of different graphs is different, aggregation of the homophilous graph with high HR can discover similar nodes, while high-order aggregation of low homophilous graph with low HR tends to deteriorate the whole network. Therefore, each graph aggregated with the same order may not be an effective way. Naturally, the homophilous graphs shall be assigned a high order to exploit as many as neighborhoods as possible, while low homophilous graphs shall be assigned a low order to reduce the influence of non-homophilous edges. Faced with the challenge of unsupervised learning, the quality of each graph cannot be evaluated precisely. So, we seek help from the final pseudo label in the current iteration to evaluate the HR of each view's graph. Specifically, we calculate the HR of each graph $\mathbf{A}^v$ by the pseudo label $\mathbf{Y}^s$ as follows:

$$\mathrm{HR}^v = \frac{\mathrm{sum}(\mathbf{A}^v \odot \mathbf{B}\mathbf{B}^T - \mathbf{I})}{\mathrm{sum}(\mathbf{A}^v - \mathbf{I})}, \tag{5}$$

where, $\mathrm{sum}(\cdot)$ represents the summation operation, $\odot$ represents the Hadamard product, $\mathbf{B} \in \{0, 1\}^{n \times c}$ is the one-hot encoding of the pseudo label $\mathbf{Y}^s$, and $\mathbf{I}$ denotes the identity matrix.

With the guidance of the pseudo label $\mathbf{Y}^s$, the obtained HR^v reflects the homophily of $\mathbf{A}^v$ relative to $\mathbf{Y}^s$, *i.e.*, the confidence of the graph quality about $\mathbf{Y}^s$. On basis of the confidence, the order of the specific graph $\mathbf{A}^v$ can thus be determined by:

$$od^v = \begin{cases} 0, & if \quad \mathrm{HR}^v \leq \varepsilon, \\ \left\lfloor \frac{1}{1-\mathrm{HR}^v} \right\rfloor, & if \quad \mathrm{HR}^v > \varepsilon, \end{cases} \tag{6}$$

where $\lfloor \cdot \rfloor$ is floor operation, and ε is a cut-off to filter out low homophilous graphs.

Via Eq. 6, the low homophilous graph will be assigned a small order od^v, and even it will be discarded when HR is less than the cut-off ε. In contrast, the homophilous graph with high HR will be assigned a higher order od^v accordingly. Finally, each view's graph $\mathbf{A}^v$ is refined with the guidance of the dual labels:

$$\overline{\mathbf{A}}^v = \alpha \mathbf{A}_{od^v}^v + \mathbf{\Omega}. \tag{7}$$

Here, since od^v is derived from the global pseudo label $\mathbf{Y}^s$, it will be adjusted adaptively with $\mathbf{Y}^s$ updating during the iteration.

Pseudo label guided graph fusion. $\mathbf{A}^v$ in different views have complementary information, and it is important to integrate them for a consensus graph $\mathbf{S}$. Intuitively, weighting and summing up each refined $\overline{\mathbf{A}}^v$ is a simple yet effective way. However, clustering is an unsupervised task without label information for guidance, how to weight each $\overline{\mathbf{A}}^v$ is challenging. To address this problem, we utilize the pseudo label $\mathbf{Y}^s$ to score each view with $score^v$, which is obtained from the graph encoding module, to determine the weights of each view. With the $\{score^v\}_{v=1}^V$, we can calculate the weight ω^v of each view $\overline{\mathbf{A}}^v$ by:

$$\omega^v = (\frac{socre^v}{\max{(score^1, score^2, ..., score^V)}})^p, \tag{8}$$

where p is a smooth-sharp parameter. When $p \in [0, 1]$, it mainly plays a smoothing role. When $p > 1$, it will sharpen the differences between views.

Finally, the global consensus graph $\mathbf{S}$ can be obtained:

$$\mathbf{S} = \sum_{v=1}^{V} \omega^v \overline{\mathbf{A}}^v = \sum_{v=1}^{V} \alpha \omega^v \mathbf{A}_{od^v}^v + \omega^v \mathbf{\Omega}, \tag{9}$$

where, V denotes the number of views.

Graph Encoding and View Evaluation

As shown in Figure 1, the input of the graph encoding module is shared feature $\mathbf{X}$, refined multiple high-order neighborhood relationship $\overline{\mathbf{A}}^v$ and the consensus graph $\mathbf{S}$. The aim of this module is to encode the shared feature $\mathbf{X}$ along with the structural information to generate the latent embedding for training or clustering. Specifically, we train a parameter-shared GNN to encode shared feature $\mathbf{X}$ with the refined $\overline{\mathbf{A}}^v$ and $\mathbf{S}$:

$$\mathbf{H}^v = GNN(\widetilde{\mathbf{A}}^v, \mathbf{X}; \mathbf{W}_\xi), \quad \mathbf{H}^s = GNN(\widetilde{\mathbf{S}}, \mathbf{X}; \mathbf{W}_\xi), \tag{10}$$

where $GNN(\cdot)$ represents a graph convolution operation which contains two linear layers, $\widetilde{\mathbf{A}}^v$ and $\widetilde{\mathbf{S}}$ denote the normalization of $\overline{\mathbf{A}}^v$ and $\mathbf{S}$ respectively, $\mathbf{W}_\xi$ are the trainable parameters for the shared GNN.

Since $\widetilde{\mathbf{A}}^v$ and $\widetilde{\mathbf{S}}$ have contained the high-order structural information, the shared feature $\mathbf{X}$ is only aggregated once followed by an activation function, and no more aggregation or nonlinear layers are stacked behind.

In addition, without the guidance of the ground truth, the quality of each view can not be completely determined so as to guide the graph fusion process. Since the latent embedding $\mathbf{H}^s$ of the consensus graph $\mathbf{S}$ contains the global common information from all views, we naturally treat the pseudo label $\mathbf{Y}^s$ obtained from it as ground truth to evaluate the clustering score ($score^v$) of the predictions from each view $\mathbf{H}^v$. Specifically, the $score^v$ can be calculated by:

$$score^v = metric(kmeans(\mathbf{H}^v), \mathbf{Y}^s), \tag{11}$$

where, $metric(\cdot)$ denotes the calculation of metrics such as accuracy, and $kmeans(\cdot)$ represents the k-means algorithm.

Model Optimization

In DuaLGR, in order to preserve information as much as possible, we stack a multilayer perceptron (MLP) decoder to reconstruct $\mathbf{X}$ and $\mathbf{A}^v$ from $\mathbf{H}^s$ by $\sigma(\mathbf{H}^s\mathbf{H}^{sT})$. Specifically, the reconstruction loss can be defined as:

$$\mathcal{L}_{Rec} = l(MLP(\mathbf{H}^s); \mathbf{X}) + \sum_{v=1}^{V} l(\sigma(\mathbf{H}^s\mathbf{H}^{sT}); \mathbf{A}^v), \tag{12}$$

where $l(\cdot;\cdot)$ is loss function, and $\sigma(\cdot)$ is activation function (cross entropy loss and sigmoid function used in this work).

For multi-view clustering, the KL divergence loss is adopted following the previous related work (Zhao et al. 2021; Ren et al. 2022). We first measure $\mathbf{H}^s$ to calculate the distribution $\mathbf{Q}^s$ of pseudo label by Student's t-distribution:

$$q_{ij}^s = \frac{(1 + ||\mathbf{H}_i^s - \boldsymbol{\mu}_j^s||^2)^{-1}}{\sum_j (1 + ||\mathbf{H}_i^s - \boldsymbol{\mu}_j^s||^2)^{-1}}, \tag{13}$$

where $\boldsymbol{\mu}^s$ are learnable parameters, which are adopted as the centroids in K clusters and are initialized by performing k-means on $\mathbf{H}^s$.

And then, the target distribution $\mathbf{P}^s$ can be derived from the $\mathbf{Q}^s$:

$$p_{ij}^s = \frac{(q_{ij}^s)^2 / \sum_i q_{ij}^s}{\sum_j ((q_{ij}^s)^2 / \sum_i q_{ij}^s)}. \tag{14}$$

Thus, the KL divergence loss is specified as follows:

$$\mathcal{L}_{KL} = \sum_{v=1}^{V} KL(\mathbf{P}^s||\mathbf{Q}^v) + KL(\mathbf{P}^s||\mathbf{Q}^s). \tag{15}$$

The $\mathcal{L}_{KL}$ on the one hand encourages the framework to learn more discriminative latent embeddings $\mathbf{H}^v$, and on the other hand, enforces the soft distribution of each view to fit the global graph's latent embeddings $\mathbf{H}^s$.

Finally, the loss function is formulated as:

$$\mathcal{L} = \mathcal{L}_{Rec} + \gamma\mathcal{L}_{KL}, \tag{16}$$

where, γ denotes the trade-off parameter.

Experiments

Experimental Setup

Datasets. The datasets consist of three categaries *i.e.*, two raw high homophilous datasets, two raw low homophilous datasets and six synthetic datasets. The specific statistics information of these datasets is summarized in Table 2.

Datasets	Clusters	Nodes	Features	Graphs	HR
ACM	3	3025	1830	$\mathcal{G}^1$ $\mathcal{G}^2$	0.82 0.64
DBLP	4	4057	334	$\mathcal{G}^1$ $\mathcal{G}^2$ $\mathcal{G}^3$	0.80 0.67 0.32
Texas	5	183	1703	$\mathcal{G}^1$	0.09
Chameleon	5	2277	2325	$\mathcal{G}^1$	0.23

Table 2: The statistics information of the four graph datasets.

- Homophilous graph datasets: two widely used homophilous multi-graph data, including ACM and DBLP. ACM is a paper network from the ACM database[1] and consists of two graphs, *i.e.*, co-paper and co-subject, whose HR are 0.82 and 0.64 respectively. DBLP is an author network from DBLP database[2]. Three graphs, co-author (HR 0.80), co-conference (HR 0.67) and co-term (HR 0.32), compose the dataset.
- Low homophilous graph datasets: Texas and Chameleon are adopted to test the performance of DuaLGR in low homophilous graphs. Texas is a webpage graph from WebKB[3] with an HR of 0.09. Chameleon (HR 0.23) is a subset of the Wikipedia network (Rozemberczki, Allen, and Sarkar 2021). Since Texas and Chameleon are single-view graph data, we copy the graph as the second view.
- Synthetic graph datasets: in order to observe the performance of DuaLGR on low homophilous graph data more intuitively, we take ACM as an example, and generate low HR graphs with the same number of edges of each view's graph for the test. The HR of these graphs are $[0.00, 0.10, 0.20, 0.30, 0.40, 0.50]$ respectively.

Baselines. Several baselines are reproduced for comparison with DuaLGR.

- Single view clustering: Line (Tang et al. 2015) and VGAE (Kipf and Welling 2016b) are two classical single-view clustering methods.
- Graph besed multi-view clustering: PMNE (Liu et al. 2017), SwMC (Nie et al. 2017) and MNE (Zhan et al. 2018) are traditional MVGC methods based on graph embedding. RMSC (Xia et al. 2014) is a multi-view spectral clustering method.
- Multi-view graph clustering: O2MAC (Fan et al. 2020) is the method that learns from both node features and graphs. MvAGC (Lin and Kang 2021) and MCGC (Pan and Kang 2021) are two recent methods based on graph filter to learn a consensus graph for clustering.

For ACM and DBLP, the results of baselines are drawn from the best in literature. For othter raw data, including

[1] https://dl.acm.org/

[2] https://dblp.uni-trier.de/

[3] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-11/www/wwkb

Methods / Datasets	ACM (HR 0.82 & 0.64)				DBLP (HR 0.87 & 0.67 & 0.32)			
	NMI%	ARI%	ACC%	F1%	NMI%	ARI%	ACC%	F1%
RMSC (2014)	39.7	33.1	63.2	57.5	71.1	76.5	89.9	82.5
LINE (2015)	39.4	34.3	64.8	65.9	66.8	69.9	86.9	85.5
VGAE (2016b)	49.1	54.4	82.2	82.3	69.3	74.1	88.6	87.4
PMNE (2017)	46.5	43.0	69.4	69.6	59.1	52.7	79.3	79.7
SwMC (2017)	8.4	4.0	41.6	47.1	37.6	38.0	65.4	56.0
MNE (2018)	30.0	24.9	63.7	64.8	–	–	–	–
O2MAC (2020)	69.2	73.9	90.4	90.5	72.9	77.8	90.7	90.1
MvAGC (2021)	67.4	72.1	89.8	89.9	77.2	**82.8**	92.8	92.3
MCGC (2021)	71.3	76.3	91.5	91.6	**83.0**	77.5	**93.0**	**92.5**
DuaLGR (ours)	**73.2**	**79.4**	**92.7**	**92.7**	75.5	81.7	92.4	91.8
Methods / Datasets	Texas (HR 0.09 & 0.09)				Chameleon (HR 0.23 & 0.23)			
VGAE (2016b)	12.7 ± 4.4	21.7 ± 8.4	**55.3** ± 1.8	29.5 ± 3.1	15.1 ± 0.7	12.4 ± 0.6	35.4 ± 1.0	29.6 ± 1.7
O2MAC (2020)	8.7 ± 0.8	14.6 ± 1.8	46.7 ± 2.4	29.1 ± 2.4	12.3 ± 0.7	8.9 ± 1.2	33.5 ± 0.3	28.6 ± 0.2
MvAGC (2021)	5.4 ± 2.8	1.1 ± 4.1	54.3 ± 2.6	19.8 ± 5.1	10.8 ± 0.8	3.3 ± 1.7	29.2 ± 0.9	24.3 ± 0.5
MCGC (2021)	12.7 ± 2.9	12.9 ± 3.8	51.9 ± 0.9	32.5 ± 1.8	9.5 ± 1.3	5.9 ± 2.7	30.0 ± 2.0	19.1 ± 0.8
DuaLGR (ours)	**32.6 ± 0.5**	**26.0 ± 0.6**	54.3 ± 0.3	**46.8 ± 0.3**	**19.5 ± 1.0**	**16.0 ± 0.6**	**41.1 ± 0.8**	**37.7 ± 1.5**

Table 3: The clustering results on two raw homophilous graph datasets and two raw low homophilous graph datasets. The best results are shown in bold, and the results which are not reported in the original papers are denoted by '–'. For ACM and DBLP, there are no 'std' values as the results of compared methods are drawn from their original papers where the 'std' is absence.

Methods / Datasets	ACM (HR 0.00 & 0.00)				ACM (HR 0.10 & 0.10)			
	NMI%	ARI%	ACC%	F1%	NMI%	ARI%	ACC%	F1%
VGAE (2016b)	0.5 ± 0.3	0.5 ± 0.4	37.4 ± 1.4	37.1 ± 1.6	0.5 ± 0.3	0.5 ± 0.3	37.1 ± 1.0	35.6 ± 2.3
O2MAC (2020)	25.0 ± 15.6	24.7 ± 15.2	55.0 ± 11.6	54.6 ± 11.5	17.6 ± 14.7	17.1 ± 14.3	49.9 ± 12.1	49.7 ± 12.0
MvAGC (2021)	0.9 ± 0.4	0.9 ± 0.4	37.1 ± 0.7	35.5 ± 3.8	1.9 ± 0.4	2.0 ± 0.4	40.9 ± 1.6	39.1 ± 4.5
MCGC (2021)	49.8 ± 22.7	42.9 ± 19.6	63.0 ± 10.4	53.5 ± 5.1	52.9 ± 20.8	44.7 ± 17.9	63.9 ± 8.5	54.6 ± 3.1
DuaLGR (ours)	**55.1 ± 0.0**	**60.7 ± 0.0**	**84.8 ± 0.0**	**84.5 ± 0.0**	**55.9 ± 0.0**	**61.7 ± 0.0**	**85.3 ± 0.0**	**85.0 ± 0.0**
Methods / Datasets	ACM (HR 0.20 & 0.20)				ACM (HR 0.30 & 0.30)			
VGAE (2016b)	0.4 ± 0.4	0.4 ± 0.4	36.9 ± 1.2	34.9 ± 2.3	0.7 ± 0.5	0.7 ± 0.5	38.0 ± 1.2	37.6 ± 1.5
O2MAC (2020)	9.6 ± 13.9	9.4 ± 13.5	42.9 ± 11.8	42.8 ± 11.7	6.7 ± 12.3	6.5 ± 12.0	40.7 ± 10.3	40.5 ± 10.2
MvAGC (2021)	5.3 ± 3.1	5.6 ± 3.4	45.7 ± 5.1	45.4 ± 5.3	15.4 ± 5.7	16.5 ± 5.9	57.7 ± 5.4	57.7 ± 5.5
MCGC (2021)	29.1 ± 5.4	31.7 ± 9.5	67.7 ± 8.2	67.2 ± 7.8	51.8 ± 9.0	57.2 ± 12.7	83.0 ± 6.7	82.9 ± 6.6
DuaLGR (ours)	**59.2 ± 0.0**	**66.0 ± 0.0**	**87.3 ± 0.0**	**87.1 ± 0.0**	**60.2 ± 0.1**	**67.6 ± 0.0**	**88.0 ± 0.0**	**88.0 ± 0.0**
Methods / Datasets	ACM (HR 0.40 & 0.40)				ACM (HR 0.50 & 0.50)			
VGAE (2016b)	9.7 ± 2.5	8.1 ± 1.9	48.4 ± 3.0	49.0 ± 2.8	26.2 ± 6.1	27.0 ± 10.1	65.9 ± 7.8	66.4 ± 7.3
O2MAC (2020)	5.5 ± 11.0	5.4 ± 10.7	40.3 ± 9.3	40.2 ± 9.2	6.6 ± 11.1	6.7 ± 11.4	42.7 ± 10.8	42.6 ± 10.8
MvAGC (2021)	36.9 ± 4.2	39.5 ± 5.3	74.0 ± 3.1	74.2 ± 3.0	64.6 ± 3.7	71.1 ± 3.6	89.4 ± 1.5	89.4 ± 1.5
MCGC (2021)	83.9 ± 3.1	88.8 ± 2.9	96.2 ± 1.0	96.2 ± 1.0	91.0 ± 2.0	94.4 ± 1.4	98.1 ± 0.5	98.1 ± 0.5
DuaLGR (ours)	**85.1 ± 0.6**	**90.1 ± 0.7**	**96.6 ± 0.2**	**96.6 ± 0.2**	**97.8 ± 0.1**	**98.9 ± 0.1**	**99.6 ± 0.0**	**99.6 ± 0.0**

Table 4: The clustering results on six synthetic ACM graph datasets with different HR. The best results are shown in bold.

Texas and Chameleon, and synthetic data, all baselines are conducted for 5 times and report the averages with standard deviations to have a fair comparison.

Metrics. Following previous works, four commonly used metrics, *i.e.*, normalized mutual information (NMI), adjusted rand index (ARI), accuracy (ACC) and F1-score (F1), are adopted to evaluate the clustering performance.

Overall Results

Raw data. The overall clustering results in the four real-world graph datasets are shown in Table 3. The results on homophilous graph datasets, *i.e.*, ACM and DBLP, demonstrate that the proposed DuaLGR has competitive performance with SOTAs, especially with comparable clustering results on ACM. When faced with the low homophilous graphs, *i.e.*, Texas and Chameleon, these SOTAs become worse dramatically, while DuaLGR has outstandingly superior performance. Specifically, DuaLGR improves NMI by about 19.9% on Texas and 4.4% on Chameleon. In addition, although VGAE also performs well, it has significant performance fluctuations (standard deviation of NMI on Texas is 4.4), while DuaLGR is more stable (standard deviation of NMI on Texas is 0.5). These results imply that the proposed DuaLGR has competitive performance on homophilous graphs and excellent performance on low ho-

Compenents / Datasets	Raw ACM (HR 0.82 & 0.64)				Synthetic ACM (HR 0.20 & 0.20)			
	NMI%	ARI%	ACC%	F1%	NMI%	ARI%	ACC%	F1%
DuaLGR (w/o $\mathcal{L}_{Rec}$)	1.7	1.8	41.1	41.0	0.07	0.06	37.0	36.5
DuaLGR (w/o $\mathcal{L}_{KL}$)	70.8	76.4	91.5	91.5	59.2	65.8	87.1	87.0
DuaLGR (w/o $\mathbf{\Omega}$)	61.8	65.8	86.9	86.9	44.5	48.6	79.0	78.6
DuaLGR (w/o od^v)	65.2	66.6	86.9	87.0	52.7	58.8	84.0	83.7
DuaLGR-original	**72.0**	**78.0**	**92.1**	**92.1**	**59.2**	**66.0**	**87.3**	**87.1**

Table 5: The ablation study results of DuaLGR on raw ACM and synthetic ACM. The original results are shown in bold.

mophilous graphs, which alleviates the dilemma of MVGC on low homophilous graphs.

Synthetic data. Table 4 reports the performance of four SOTAs and DuaLGR on the six synthetic ACM datasets. In general, DuaLGR outperforms all SOTAs in the low homophilous graph, especially HR in the range of 0.00-0.20. As the HR decreases, all methods show varying degrees of performance degradation. Taking MvAGC as an example, the ACC dropped from 89.4% (HR 0.50) to 37.1% (HR 0.00), which indicates that the low homophilous graph severely constrains the performance of these methods. However, the proposed DuaLGR still has an ACC of 84.8% when HR is 0.00, because the refined graph can reduce the influence of non-homophilous edges with the guidance of dual labels.

Ablation Studies and Analysis

The ablation study results of DuaLGR are presented in this subsection, including the effect of each component, convergence analysis, and parameter sensitive analysis. For simplicity, the ablation studies are conducted on the multi-graph dataset ACM and synthetic ACM with HR 0.20 which has enough homophilous edges and non-homophilous edges.

Effect of each loss. In order to understand the significance of components of the proposed DuaLGR, we remove each loss individually to observe the change in performance. The performance without a specific component is listed in Table 5, as can be seen, the performance of DuaLGR will decrease with the absence of each component. Specifically, $\mathcal{L}_{Rec}$ plays a significant role in DuaLGR, while $\mathcal{L}_{KL}$ seems to be faint.

Effect of each label's guidance. In Table 5, we remove each component, *i.e.*, $\mathbf{\Omega}$ and od^v, to see the influences from the guidance of the two labels respectively. As we can see, the contributions of $\mathbf{\Omega}$ and od^v are similar on homophilous graphs, while on low homophilous graphs it is $\mathbf{\Omega}$ that contributes more, implying that the influence of non-homophilous edges is attenuated with the refinement of $\mathbf{\Omega}$.

Convergence analysis. The left side of Figure 2 depicts the trend of the loss with epoch in the training process. On homophilous graphs, DuaLGR can converge more smoothly and quickly. Despite the constraints of low homophilous graphs, DuaLGR can still converge through the graph refinement process with the guidance of dual labels.

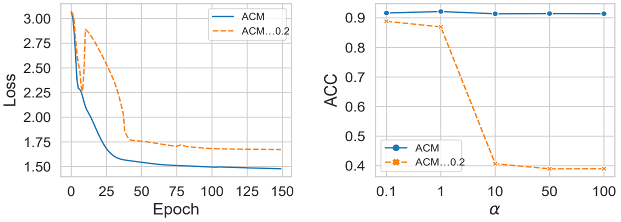

Figure 2: Convergence analysis (left) and parameter sensitive analysis (right) on ACM and synthetic ACM (HR 0.20).

Parameter sensitive analysis. The right side of Figure 2 shows the parameter sensitive analysis about α. α controls the influence of given graphs, which means the larger the α is, the lower the refining extent is, and vice versa. From the right side of Figure 2, on ACM dataset, the highest ACC is 92.1% when $\alpha = 1$, compared to 91.6% and 91.4% when $\alpha = 0.1$ and $\alpha = 10$ respectively. This demonstrates that on the homophilous dataset, we can also obtain a satisfactory result when the extent of refinement is small. In contrast, on the synthetic low homophilous dataset ACM (HR 0.20), it is clear that the highest location is $\alpha = 0.1$ where the extent of refinement is largest. This shows that on the low homophilous graphs, the refinement process contributes largely to the results.

Conclusion and Future Work

In this work, we try to mitigate the reliance of most existing MVGC methods on homophily assumption, and focus on the idea of graph refinement, proposing dual label-guided graph refinement framework for multi-view graph clustering (DuaLGR). Specifically, DuaLGR extracts high-level semantic and view-common information to generate soft-label and pseudo label. The dual labels are then conducted to guide the graph refinement and fusion process. Our experimental results demonstrate the necessity of refining the low homophilous graphs, and the effectiveness of the proposed dual label guidance. We also show superior clustering performance on both real-world and synthetic low homophilous multi-view graphs datasets. In future works, we will try to extent the proposed dual label-guided graph refinement idea to more unsupervised tasks, such as traditional multi-view clustering and single-view graph clustering tasks.

Acknowledgments

This work was supported in part by Sichuan Science and Technology Program (Nos. 2021YFS0172, 2022YFS0047, and 2022YFS0055), Medico-Engineering Cooperation Funds from University of Electronic Science and Technology of China (No. ZYGX2021YGLH022), National Science Foundation (MRI 2215789), and Lehigh's grants (S00010293 and 001250).

References

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and Deep Graph Convolutional Networks. In *ICML*, 1725–1735.

Cheng, J.; Wang, Q.; Tao, Z.; Xie, D.; and Gao, Q. 2020. Multi-View Attribute Graph Convolution Networks for Clustering. In *IJCAI*, 2973–2979.

Chien, E.; Peng, J.; Li, P.; and Milenkovic, O. 2021. Adaptive Universal Generalized PageRank Graph Neural Network. In *ICLR*.

Cotta, L.; Morris, C.; and Ribeiro, B. 2021. Reconstruction for Powerful Graph Representations. In *NeurIPS*, 1713–1726.

Fan, S.; Wang, X.; Shi, C.; Lu, E.; Lin, K.; and Wang, B. 2020. One2Multi Graph Autoencoder for Multi-view Graph Clustering. In *WWW*, 3070–3076.

Hassani, K.; and Khasahmadi, A. H. 2020. Contrastive multi-view representation learning on graphs. In *ICML*, 4116–4126.

He, D.; Liang, C.; Liu, H.; Wen, M.; Jiao, P.; and Feng, Z. 2022. Block modeling-guided graph convolutional neural networks. In *AAAI*, 4022–4029.

Hinton, G. E.; and Salakhutdinov, R. R. 2006. Reducing the dimensionality of data with neural networks. *Science*, 313(5786): 504–507.

Huang, Z.; Ren, Y.; Pu, X.; and He, L. 2021. Non-Linear Fusion for Self-Paced Multi-View Clustering. In *ACM MM*, 3211–3219.

Kipf, T. N.; and Welling, M. 2016a. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*.

Kipf, T. N.; and Welling, M. 2016b. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

Liang, W.; Liu, X.; Zhou, S.; Liu, J.; Wang, S.; and Zhu, E. 2022. Robust Graph-Based Multi-View Clustering. In *AAAI*, 7462–7469.

Lim, D.; Hohne, F.; Li, X.; Huang, S. L.; Gupta, V.; Bhalerao, O.; and Lim, S. N. 2021. Large Scale Learning on Non-Homophilous Graphs: New Benchmarks and Strong Simple Methods. In *NeurIPS*, 20887–20902.

Lin, Z.; and Kang, Z. 2021. Graph Filter-based Multi-view Attributed Graph Clustering. In *IJCAI*, 2723–2729.

Liu, W.; Chen, P.-Y.; Yeung, S.; Suzumura, T.; and Chen, L. 2017. Principled multilayer network embedding. In *ICDMW*, 134–141.

Nie, F.; Li, J.; Li, X.; et al. 2017. Self-weighted Multiview Clustering with Multiple Graphs. In *IJCAI*, 2564–2570.

Pan, E.; and Kang, Z. 2021. Multi-view Contrastive Graph Clustering. In *NeurIPS*, 2148–2159.

Qiang, Q.; Zhang, B.; Wang, F.; and Nie, F. 2021. Fast multi-view discrete clustering with anchor graphs. In *AAAI*, 9360–9367.

Ren, Y.; Pu, J.; Yang, Z.; Xu, J.; Li, G.; Pu, X.; Yu, P. S.; and He, L. 2022. Deep clustering: A comprehensive survey. *arXiv preprint arXiv:2210.04142*.

Rozemberczki, B.; Allen, C.; and Sarkar, R. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2): cnab014.

Suresh, S.; Budde, V.; Neville, J.; Li, P.; and Ma, J. 2021. Breaking the limit of graph neural networks by improving the assortativity of graphs with local mixing patterns. *arXiv preprint arXiv:2106.06586*.

Tan, J.; Shi, Y.; Yang, Z.; Wen, C.; and Lin, L. 2021. Unsupervised Multi-View Clustering by Squeezing Hybrid Knowledge From Cross View and Each View. *TMM*, 23: 2943–2956.

Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line: Large-scale information network embedding. In *WWW*, 1067–1077.

Tse, T. H. E.; Kim, K. I.; Leonardis, A.; and Chang, H. J. 2022. Collaborative Learning for Hand and Object Reconstruction With Attention-Guided Graph Convolution. In *CVPR*, 1664–1674.

Wang, T.; Jin, D.; Wang, R.; He, D.; and Huang, Y. 2022. Powerful Graph Convolutional Networks with Adaptive Propagation Mechanism for Homophily and Heterophily. In *AAAI*, 4210–4218.

Xia, R.; Pan, Y.; Du, L.; and Yin, J. 2014. Robust Multi-View Spectral Clustering via Low-Rank and Sparse Decomposition. In *AAAI*, 2149–2155.

Xu, J.; Ren, Y.; Tang, H.; Pu, X.; Zhu, X.; Zeng, M.; and He, L. 2021. Multi-VAE: Learning disentangled view-common and view-peculiar visual representations for multi-view clustering. In *CVPR*, 9234–9243.

Yang, M.; Li, Y.; Hu, P.; Bai, J.; Lv, J. C.; and Peng, X. 2022. Robust multi-view clustering with incomplete information. *TPAMI*, 45(1): 1055–1069.

Zhan, K.; Nie, F.; Wang, J.; and Yang, Y. 2018. Multiview consensus graph clustering. *TIP*, 28(3): 1261–1270.

Zhang, Z.; Liu, Q.; Wang, H.; Lu, C.; and Lee, C. 2022. Protgnn: Towards self-explaining graph neural networks. In *AAAI*, 9127–9135.

Zhao, H.; Yang, X.; Wang, Z.; Yang, E.; and Deng, C. 2021. Graph Debiased Contrastive Learning with Joint Representation Clustering. In *IJCAI*, 3434–3440.

Zhong, G.; and Pun, C.-M. 2022. Improved Normalized Cut for Multi-View Clustering. *TPAMI*, 44(12): 10244–10251.

Zhu, Y.; Xu, W.; Zhang, J.; Du, Y.; Zhang, J.; Liu, Q.; Yang, C.; and Wu, S. 2021. A Survey on Graph Structure Learning: Progress and Opportunities. *arXiv preprint arXiv:2103.03036*.