Attributed Graph Clustering with Dual Redundancy Reduction

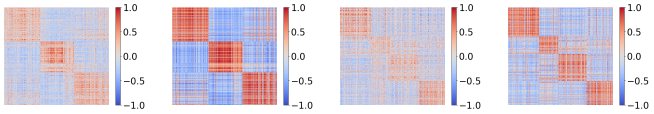
Lei Gong, **Sihang Zhou**, **Wenxuan Tu** and **Xinwang Liu***
National University of Defense Technology, Changsha, China
glnudt@163.com, xinwangliu@nudt.edu.cn

Abstract

Attributed graph clustering is a basic yet essential method for graph data exploration. Recent efforts over graph contrastive learning have achieved impressive clustering performance. However, we observe that the commonly adopted InfoMax operation tends to capture redundant information, limiting the downstream clustering performance. To this end, we develop a novel method termed attributed graph clustering with dual redundancy reduction (AGC-DRR) to reduce the information redundancy in both input space and latent feature space. Specifically, for the input space redundancy reduction, we introduce an adversarial learning mechanism to adaptively learn a redundant edge-dropping matrix to ensure the diversity of the compared sample pairs. To reduce the redundancy in the latent space, we force the correlation matrix of the cross-augmentation sample embedding to approximate an identity matrix. Consequently, the learned network is forced to be robust against perturbation while discriminative against different samples. Extensive experiments have demonstrated that AGC-DRR outperforms the state-of-the-art clustering methods on most of our benchmarks. The corresponding code is available at https://github.com/gongleii/AGC-DRR.

1 Introduction

Deep clustering methods, which improve the performance of complicated clustering problems with the help of deep network architecture, have achieved significant progress on applications like semantic segmentation [Caron *et al.*, 2018], social network analysis [Zhong *et al.*, 2016], and face recognition [Schroff *et al.*, 2015]. However, the successes of these methods are mostly rooted in the advances of autoencoder [Guo *et al.*, 2017; Xie *et al.*, 2016], convolutional neural networks, and generative adversarial networks [Xu *et al.*, 2019], which are not applicable to the non-Euclidean graph datasets. However, structural information is also found essential for data analysis [Zhang *et al.*, 2021]. As a consequence, graph convolutional network (GCN) [Kipf and Welling, 2016a] based clustering algorithms, for example, deep attentional embedded graph clustering (DAEGC) [Wang *et al.*, 2019] and deep fusion clustering network (DFCN) [Tu *et al.*, 2021], are recently attracting increasing attention.



(a) ACM-G (b) ACM-A (c) DBLP-G (d) DBLP-A

Figure 1: Illustration of the cosine similarity matrices in the latent space on the datasets ACM and DBLP with the corresponding methods, i.e., GAE and AGC-DRR. The samples are permuted to have those that belong to the same cluster located beside each other.

Nevertheless, the above GCN-based methods learn node representation by adopting the graph structure reconstruction principle, which may ignore the subtle but essential relationships and lead to indiscriminative representation among samples [Suresh *et al.*, 2021]. The problem is especially severe in the unsupervised learning scenario. Consequently, graph contrastive learning (GCL), which focuses on maximizing mutual information (InfoMax) between representations of the same instance with different augmentations, is proposed to alleviate the problem [Velickovic *et al.*, 2019; Zhao *et al.*, 2020]. It has achieved considerable improvement in node embedding and clustering. By utilizing two decoupled GCN-based encoders, multi-view graph representation learning (MVGRL) [Hassani and Khasahmadi, 2020] conducts cross-view contrastive learning to facilitate invariant node embedding learning. Although good performance has been achieved, the InfoMax principle is found to risk capturing task-unrelated information, which is also able to satisfy InfoMax [Tschannen *et al.*, 2019] and lead to sub-optimal node representation. To solve the problem, adversarial-GCL [Suresh *et al.*, 2021] integrates information bottleneck theory (IB) [Tishby *et al.*, 2000] with self-supervised GCNs by an adversarial learning mechanism to reduce redundant information in graph-structure data for graph-level tasks. However, it ignores the node-level redundancy reduction. Moti-

*Corresponding author

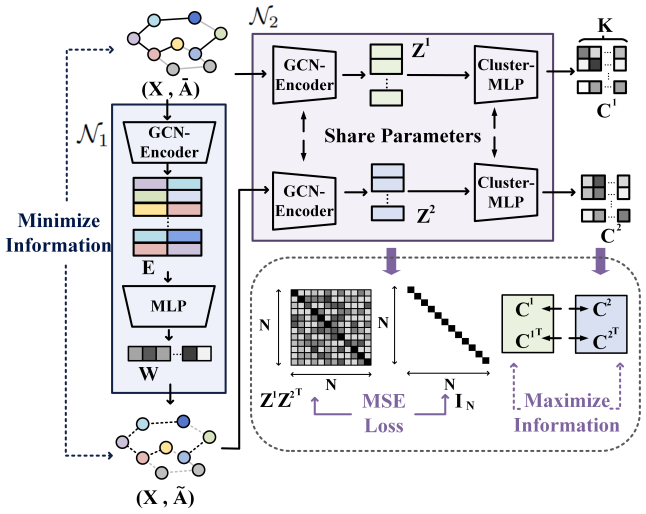


Figure 2: The proposed AGC-DRR consists of adversarial sub-networks pairs, $\mathcal{N}_1$ is a structure augmented sub-network, which adaptively learns a redundant edge-dropping matrix to obtain augmented graph by InfoMin principle to reduce redundant information in input space, and $\mathcal{N}_2$ is a clustering sub-network optimized by InfoMax principle and a $\mathcal{L}_{MSE}$ loss which decreases latent space redundancy. The clustering results are obtained by the average of $\mathbf{C}^1$ and $\mathbf{C}^2$.

vated by decreasing redundant information in node clustering tasks, we propose attributed graph clustering with dual redundancy reduction (AGC-DRR) to reduce the information redundancy both in input space and latent feature space. Specifically, for the input space redundancy reduction, we introduce an adversarial learning mechanism to adaptively learn a redundant edge-dropping matrix to ensure the diversity of the compared sample pairs with the anchor of the original graph. Subsequently, to reduce the redundancy in latent space, we force the correlation matrix of the cross-augmentation sample embedding to approximate an identity matrix. Consequently, the learned network is more robust against perturbation while discriminative against different samples. In Fig. 1, we compare the discriminative capability of GAE and our proposed AGC-DRR by calculating the cosine similarity matrices in the learned latent space, respectively. As we can see, when adopting the same encoder structure, the discriminative capability of the sample embedding is largely improved by our proposed dual redundancy reduction mechanism. We list the contributions of this paper as follows:

- AGC-DRR is the first attributed graph clustering algorithm that adaptively learns the adjacent matrix with an adversarial learning mechanism to the best of our knowledge.
- We propose a dual redundancy reduction strategy that decreases the information redundancy in both the input space and latent feature space to improve clustering performance.
- AGC-DRR is free from pre-training, which makes the algorithm efficient and more stable.
- Extensive experimental results have demonstrated that AGC-DRR outperforms the state-of-the-art clustering methods on most of the compared datasets.

2 Related Work

2.1 Deep Graph Clustering

A proper self-supervised principle is essential for deep clustering methods. Following the auto-encoder framework, graph auto-encoder (GAE) and variational graph auto-encoder (VGAE) [Kipf and Welling, 2016b] learn node representations by reconstructing the adjacent matrix. Deep attentional embedded graph clustering (DAEGC) [Wang *et al.*, 2019] integrates GCNs with a self-attention mechanism to capture more informative relationships among nodes. Also, adversarially regularized graph auto-encoder (ARGA) [Pan *et al.*, 2020] introduces an adversarial learning mechanism into GAE to improve the quality of learned representation. The mentioned methods improve the clustering performance by carefully exploiting the structural information, other methods like SDCN [Bo *et al.*, 2020] and DFCN [Tu *et al.*, 2021] achieve the target through learning both the attribute and structural information. Specifically, they combine the auto-encoder and GAE with a carefully designed mechanism for more comprehensive information merging. Moreover, different from the above methods, deep graph infomax (DGI) [Velickovic *et al.*, 2019] learns node embedding with an InfoMax principle. After that, various GCL-based algorithms are proposed by maximizing mutual information between representations of the same instance with different augmentation methods [Zhao *et al.*, 2021; Hassani and Khasahmadi, 2020]. However, the graph structural augmentation strategy in most existing GCL-based methods is pre-defined edge perturbation, which can not be optimized and is separated from representation learning and clustering task. To have the augmentation better serve the task of graph clustering, we design a special sub-network to learn the structure augmented graph adaptively. The learning processes of graph structure and clustering are united into a common adversarial optimization framework.

2.2 Information Maximization Principle

Driven by the great success of contrastive learning (CL) of CNN-based methods in computer version scenarios, great progress has also been witnessed in the field of unsupervised graph learning. Specifically, GCL applies CL on graphs to capture subtle relationships for high-quality node representation and clustering performance improvement. The trailblazing work DGI [Velickovic *et al.*, 2019] proposes to obtain node representations by maximizing mutual information [Hjelm *et al.*, 2018] between the local patch and global summary of a graph. Further, mutual information is developed into graphs with contrastive augmented pairs generated by edge-dropping or edge perturbation [Zhao *et al.*, 2020]. However, researchers have found that the InfoMax principle could put the corresponding algorithm at the risk of collecting too much trivial and downstream task-irrelevant information for over-accurate node recognition [Suresh *et al.*, 2021]. As a consequence, information bottleneck (IB) theory [Tishby *et*

Notations	Meaning
$\mathcal{G}$	Original graph
$\mathcal{G}'$	Structure augmented graph
$\mathcal{N}_1$	Structure augmented sub-network
$\mathcal{N}_2$	Clustering sub-network
$\mathbf{I}_N \in \mathbb{R}^{N\times N}$	Identity matrix
$\mathbf{X} \in \mathbb{R}^{N\times d}$	Attribute matrix
$\mathbf{A} \in \mathbb{R}^{N\times N}$	Original adjacent matrix
$\bar{\mathbf{A}} \in \mathbb{R}^{N\times N}$	Normalized adjacent matrix
$\tilde{\mathbf{A}} \in \mathbb{R}^{N\times N}$	Augmented adjacent matrix
$\mathbf{Z} \in \mathbb{R}^{N\times d'}$	Graph embedding in $\mathcal{N}_1$
$\mathbf{Z}^v \in \mathbb{R}^{N\times d'}$	Graph embedding of the v-th view in $\mathcal{N}_2$
$\mathbf{C}^v \in \mathbb{R}^{N\times K}$	Clustering indicator matrix in the v-th view
$\mathbf{E} \in \mathbb{R}^{M\times 2d'}$	Edge embedding
$\mathbf{W} \in \mathbb{R}^{M\times 1}$	Edge-oriented weight vector
$\mathbf{W}' \in \mathbb{R}^{N\times N}$	Edge-oriented weight matrix
$\mathbf{D} \in \mathbb{R}^{N\times N}$	Degree matrix

Table 1: Basic notations for the proposed AGC-DRR

al., 2000], which aims to obtain minimal sufficient information for downstream tasks, is taken into consideration to avoid this issue. Graph information bottleneck (GIB) [Wu *et al.*, 2020] applies IB for graph representation learning and then GIB is used in [Yu *et al.*, 2020] to address subgraph recognition problem.

3 Method

In this section, we will introduce the proposed attributed graph clustering with dual redundancy reduction (AGC-DRR) algorithm, which unifies the graph structural augmentation and sample clustering into a common min-max optimization framework to reduce the information redundancy in both input and latent feature spaces. As shown in Fig. 2, AGC-DRR mainly consists of two components, i.e., structure augmented sub-network ($\mathcal{N}_1$) and clustering sub-network ($\mathcal{N}_2$). Next, we will first introduce the basic notations and preliminaries, and then introduce $\mathcal{N}_1$ and $\mathcal{N}_2$ in detail, respectively.

3.1 Basic Notations

Given an undirected graph $\mathcal{G} = \{\mathcal{E}, \mathcal{V}\}$ with K categories, where $\mathcal{E}$ and $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ are the corresponding edge set and node set, $M = |\mathcal{E}|$, $N = |\mathcal{V}|$. $\mathbf{X} \in \mathbb{R}^{N\times d}$ is the node attribute matrix and d is the raw attribute dimension. $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the original adjacent matrix, $\mathbf{A}_{ij} = 1$ denotes that there exists a connection between node v_i and node v_j, otherwise, $\mathbf{A}_{ij} = 0$. The notations are summarized in Table 1.

3.2 Preliminaries

Graph Encoder As shown in Fig. 2, the GCN-Encoder is a three-layer graph convolutional network (GCN) [Kipf and Welling, 2016a] that aggregates the first-order neighbor information to update the embedding of the central node for representation learning, which is formulated as below:

$$\mathbf{Z}^{(l)} = \sigma(\bar{\mathbf{A}}\mathbf{Z}^{(l-1)}\mathbf{H}^{(l)}), \tag{1}$$

$$\bar{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\mathbf{D}^{-\frac{1}{2}}, \tag{2}$$

where $\bar{\mathbf{A}} \in \mathbb{R}^{N\times N}$ denotes the normalized adjacent matrix, $\mathbf{D} = diag(d_1, d_2, \ldots, d_N) \in \mathbb{R}^{N\times N}$ is a degree matrix, $d_n = \sum_{j=1}^{N} \mathbf{A}_{nj}$, and $\mathbf{I} \in \mathbb{R}^{N\times N}$ is an identity matrix. $\mathbf{Z}^{(l)} \in \mathbb{R}^{N\times d^{(l)}}$ and $\mathbf{H}^{(l)} \in \mathbb{R}^{d^{(l-1)}\times d^{(l)}}$ denote the latent representations and network parameters of the l-th layer, respectively. σ denotes the Tanh activation function.

Graph Contrastive Learning Different from the auto-encoder-based methods [Kipf and Welling, 2016b; Wang *et al.*, 2019] that learn node representations by reconstructing the graph structure, graph contrastive learning (GCL) aims to maximize the agreement between positive sample pairs and minimize the agreement between negative sample pairs. To this end, mutual information maximization (InfoMax) [Hassani and Khasahmadi, 2020] is a commonly adopted measure to estimate the agreement between sample pairs. The graph contrastive learning objective is generally formulated as:

$$I(\mathbf{Z}^1, \mathbf{Z}^2) = \frac{1}{N}\sum_{i=1}^{N} log\frac{exp(sim(z_i^1, z_i^2))}{\sum_{i'=1, i'\neq i}^{N} exp(sim(z_i^1, z_{i'}^2))}, \tag{3}$$

where $I(\cdot,\cdot)$ and $sim(\cdot,\cdot)$ denote the mutual information and the cosine similarity, respectively. $\mathbf{Z}^1 \in \mathbb{R}^{N\times d'}$ and $\mathbf{Z}^2 \in \mathbb{R}^{N\times d'}$ are two-view graph embeddings learned from the corresponding views, z_i^1 and z_i^2 refer to the i-th rows in $\mathbf{Z}^1$ and $\mathbf{Z}^2$, respectively.

3.3 Clustering Sub-network

Most existing GCL-based clustering methods focus on learning node representations by maintaining the consistency of the latent feature spaces of different views as Eq. (3) and then utilizing classic clustering algorithms (e.g., K-means [Wong, 1979]) to obtain the clustering results over the learned representations. In this circumstance, the optimization processes of representation learning and node clustering are disconnected, thus leading to sub-optimal clustering performance. To solve this issue, we develop a one-step clustering sub-network that can directly predict the probabilities of cluster-ID for each sample. Specifically, after obtaining the graph embeddings of both views, we transform them into a K-dimension clustering space by feeding $\mathbf{Z}^1$ and $\mathbf{Z}^2$ into a 1-layer Multi-layer Perception (MLP) with a softmax activation function, where K refers to the number of clusters. The above learning process is formulated as:

$$\mathbf{C}^v = softmax(MLP(\mathbf{Z}^v)), \quad v \in \{1, 2\}, \tag{4}$$

where $\mathbf{C}^v \in \mathbb{R}^{N\times K}$ denotes the clustering indicator matrix in the v-th view. To maintain the consistency of two-view clustering spaces, we reformulate the Eq. (3) from the perspective of the sample level as below:

$$I(\mathbf{C}^1, \mathbf{C}^2) = \frac{1}{N}\sum_{i=1}^{N} log\frac{exp(sim(c_i^1, c_i^2))}{\sum_{i'=1, i'\neq i}^{N} exp(sim(c_i^1, c_{i'}^2))}, \tag{5}$$

Algorithm 1 The training procedure of AGC-DRR

Input: Graph data $\{\mathbf{A}, \mathbf{X}\}$; Number of clusters K; Maximum iterations T; Hyper-parameter λ
Output: Clustering results

```
for t = 1 : T do
    Calculate W and Ã to obtain the structure augmented graph by Eq. (9) and Eq. (10), respectively;
    /* Fix N_2 and optimize N_1 */
    Calculate C^1 and C^2 by Eq. (4);
    Update N_1 by minimizing the objective in Eq. (11).
    /* Fix N_1 and optimize N_2 */
    Calculate Z^1 and Z^2 by Eq. (1);
    Calculate C^1 and C^2 by Eq. (4);
    Update N_2 by maximizing the objective in Eq. (12).
end for
Obtain clustering results over the average of C^1 and C^2
return Clustering results
```

where c_i^1 and c_i^2 are the i-th rows in $\mathbf{C}^1$ and $\mathbf{C}^2$, respectively. To improve the robustness of the clustering sub-network against the perturbation derived from other components, we further consider the consistency from the perspective of cluster level via Eq. (6):

$$I(\mathbf{C}^{1\mathbf{T}}, \mathbf{C}^{2\mathbf{T}}) = \frac{1}{K}\sum_{j=1}^{K} log\frac{exp(sim(c_j^{1\mathbf{T}}, c_j^{2\mathbf{T}}))}{\sum_{j'=1, j'\neq j}^{K} exp(sim(c_j^{1\mathbf{T}}, c_{j'}^{2\mathbf{T}}))}, \tag{6}$$

where $c_j^{1\mathbf{T}}$ and $c_j^{2\mathbf{T}}$ are the j-th columns in $\mathbf{C}^1$ and $\mathbf{C}^2$, respectively. In this way, both the similarities of clustering assignments for the same instance and node distributions in the two views are considered to optimize this sub-network to further improve the clustering performance. The total mutual information between two views is formulated as:

$$I(\mathcal{G}, \mathcal{G}') = I(\mathbf{C}^1, \mathbf{C}^2) + I(\mathbf{C}^{1\mathbf{T}}, \mathbf{C}^{2\mathbf{T}}). \tag{7}$$

Latent Space Redundancy Reduction Although the InfoMax principle plays a crucial role in GCL-based methods for performance improvement, it risks enabling the encoder to capture redundant information when estimating the agreement of sample pairs. To alleviate this issue, we introduce a redundancy reduction strategy into the latent space by forcing the cross-view correlation matrix to approximate an identity matrix:

$$\mathcal{L}_{MSE} = \frac{1}{N}\left\|\mathbf{Z}^1\mathbf{Z}^{2\mathbf{T}} - \mathbf{I}_N\right\|_F^2. \tag{8}$$

This term reduces the redundancy across two-view embeddings within the corresponding graphs. By this means, redundant information in the embedding could be minimized and more discriminative features could be well preserved. Hence, it makes the learned representations be affected less by irrelevant information, thus guaranteeing the quality of latent space for subsequent clustering tasks.

3.4 Structure Augmented Sub-network

In existing GCL-based methods, edge perturbation is a commonly adopted graph augmentation before network learning, and the augmented graph is generally viewed as the ground truth information in a fixed pattern. Since the augmented graph is stemmed from the original graph, it would contain some incorrect or redundant connections. If these noisy structures are not eliminated in the network learning process, the learned clustering space would inherit it in the final step. To tackle this problem, we design a structure augmented sub-network to learn a clustering-oriented structural graph, which would adaptively learn a redundant edge-dropping matrix to ensure the diversity of the compared sample pairs.

Edge Weight Learner As shown in Fig. 2, there exists an additional GCN-Encoder in $\mathcal{N}_1$ that has an identical architecture as the one in $\mathcal{N}_2$. Similarly, this graph encoder accepts the normalized adjacent matrix $\bar{\mathbf{A}}$ and the node attribute matrix $\mathbf{X}$ as input and outputs the graph embedding $\mathbf{Z} \in \mathbb{R}^{N\times d'}$. The learned graph embedding $\mathbf{Z}$ is first adopted to generate the edge embedding $\mathbf{E} = \{\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_M\} \in \mathbb{R}^{M\times 2d'}$, where $\mathbf{E}_m = C(z_i, z_j)$, $C(\cdot,\cdot)$ is a concatenation operation, z_i and z_j are the embeddings of central node v_i and neighbor node v_j. Then the resultant edge embedding $\mathbf{E}$ is fed into a 1-layer MLP with sigmoid activation function to obtain the edge-oriented weight vector $\mathbf{W} = [w_1, w_2, \ldots, w_M]^{\mathbf{T}} \in \mathbb{R}^{M\times 1}$ as Eq. (9):

$$w_i = sigmoid(MLP(\mathbf{E}_i)), \tag{9}$$

where w_i means the probability of the corresponding original edge e_i being preserved in the augmented graph. Accordingly, $1 - w_i$ denotes the edge-dropping probability.

After that, we transform the weight vector $\mathbf{W} \in \mathbb{R}^{M\times 1}$ into an edge-oriented weight matrix $\mathbf{W}' \in \mathbb{R}^{N\times N}$ to generate the structure augmented graph $\mathcal{G}'$. Specifically, we fill $\mathbf{W}'_{ij}$ with w_m if node v_i and node v_j are connected via the edge e_m in the original graph, otherwise, $\mathbf{W}'_{ij}$ is set to zero value. The construction of augmented adjacent matrix $\tilde{\mathbf{A}}$ is formulated as follows:

$$\tilde{\mathbf{A}} = \mathbf{W}' \odot \bar{\mathbf{A}}, \tag{10}$$

where $\odot$ is the Hadamard product, i.e., $\tilde{\mathbf{A}}_{ij} = \mathbf{W}'_{ij} \times \bar{\mathbf{A}}_{ij}$. Thus, the final augmented graph $\mathcal{G}'$ consists of the augmented adjacent matrix $\tilde{\mathbf{A}}$ and the node attribute matrix $\mathbf{X}$.

By minimizing Eq. (7), the structure augmented sub-network is enabled to reduce the redundant information in the input space to ensure the diversity of the compared sample pairs. In this way, the learning processes of graph structure and clustering are united into a common adversarial optimization framework, which could make both sub-networks benefit each other to alleviate the risk of information redundancy for better clustering.

Regularization Term To filter the redundant information as much as possible, we introduce a regularization term $\frac{1}{M}\sum_{i=1}^{M} w_i$ into Eq. (7) to control the ratio of structure information preservation and reduction, where w_i refers the probability that i-th edge gets preserved.

In summary, the final objective function for $\mathcal{N}_1$ and $\mathcal{N}_2$ can be formulated as Eq. (11) and Eq. (12), respectively.

$$min \;\; I(\mathcal{G}, \mathcal{G}') + \frac{\lambda}{M}\sum_{i=1}^{M} w_i, \tag{11}$$

Dataset	#Samples	#Dimensions	#Edges	#Clusters
ACM	3025	1870	13128	3
DBLP	4057	334	3528	4
CITE	3327	3703	4552	6
AMAP	7650	745	119081	8

Table 2: Summary of datasets.

$$max \;\; I(\mathcal{G}, \mathcal{G}') - \mathcal{L}_{MSE}, \tag{12}$$

where λ is a pre-defined hyper-parameter. The training procedure of AGC-DRR is presented in Algorithm 1.

4 Experiment

4.1 Experiments Setup

Benchmark Datasets We evaluate the proposed AGC-DRR on four public benchmark datasets including ACM[1], DBLP[2], CITE[3], and AMAP [Shchur *et al.*, 2018]. The brief descriptions of these datasets are summarized in Table 2.

Training Details We conduct experiments to evaluate the proposed AGC-DRR on the PyTorch platform with the NVIDIA GeForce RTX 3080. Within an adversarial learning framework, $\mathcal{N}_1$ and $\mathcal{N}_2$ are optimized by minimizing Eq. (11) and maximizing Eq. (12), respectively, and they are trained alternately. We train AGC-DRR on all benchmark datasets for at least 100 iterations until convergence. In the testing phase, we calculate the average of both clustering indicator matrices, i.e., $\mathbf{C}^1$ and $\mathbf{C}^2$, and directly predict the cluster-ID for each sample using a softmax function. To avoid the adverse effect of randomness, we run each experiment 10 times and report the average values with standard deviations.

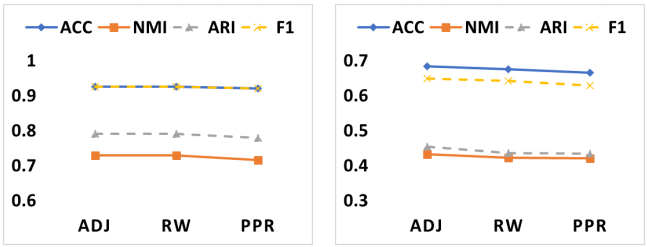
Evaluation Metrics Here we adopt four public metrics to evaluate clustering performance for all compared methods, including Clustering Accuracy (C-ACC), Average Rand Index (ARI), Normalized Mutual Information (NMI), and macro F1-score (F1).

Parameter Settings For MVGRL and ARGA/ARVGA methods, we reproduce the official source code by following the parameter settings of their original papers and report the corresponding clustering results. For other compared methods, we directly record the clustering results reported in DFCN [Tu *et al.*, 2021]. For our proposed AGC-DRR, we optimize it with the Adam optimizer, the learning rates for $\mathcal{N}_1$ and $\mathcal{N}_2$ are set to 1e-3 and 1e-4 on CITE, and 1e-4, 5e-4 on others, respectively. The regularized hyper-parameter λ is set as 1 for all datasets.

4.2 Clustering Results and Analysis

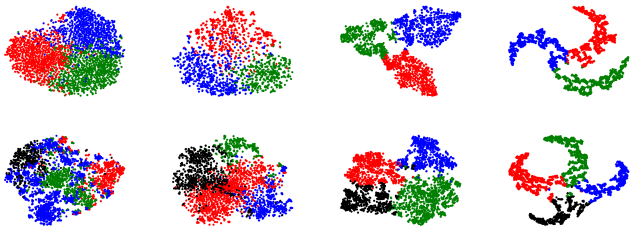
To verify the superiority of AGC-DRR, we compare it with 13 clustering methods, including 1) one classic clustering method: K-means [Wong, 1979]; 2) three deep neural network (DNN)-based clustering methods: AE [Hinton and Salakhutdinov, 2006], DEC [Xie *et al.*, 2016], and IDEC [Guo *et al.*, 2017]; and 3) nine graph neural network (GNN)-based clustering methods: GAE/VGAE [Kipf and Welling, 2016b], ARGA/ARVGA [Pan *et al.*, 2020], DAEGC [Wang *et al.*, 2019], MVGRL [Hassani and Khasahmadi, 2020], DFCN [Tu *et al.*, 2021], and SDCN/SDCN$_Q$ [Bo *et al.*, 2020].

[1] https://dl.acm.org/
[2] https://dblp.uni-trier.de
[3] http://citeseerx.ist.psu.edu/index



(a) ACM (b) CITE

Figure 3: Effect of Different Graph Structure.



(a) AE (b) GAE (c) DFCN (d) OURS

Figure 4: t-SNE visualization of different methods on ACM and DBLP, respectively.

Table 3 presents the clustering performance comparison of all compared clustering algorithms. From these results, we can observe that 1) compared with the early clustering methods that exploit the node attributes but ignore the structure information, i.e., K-means, AE, DEC, and IDEC, AGC-DRR outperforms them by a large margin, these results indicate that the structure information among data samples is significant to learn more discriminative node representations; 2) AGC-DRR also achieves competitive performance on most datasets compared with GNN-based methods that consider both attribute and structure information for clustering. Taking the metric F1 for instance, it exceeds DFCN by 1.75%, 4.20%, 0.52%, and 1.14% performance increments, respectively. This is because that AGC-DRR considers reducing the redundant information in both input and latent feature spaces, thus the network is enabled to learn a clustering-friendly graph augmentation as well as discriminative latent representations. In addition, for the less satisfying performance on the CITE dataset, we think the reason is that the attribute information of CITE plays a more important role than its structural information in the performance of clustering which may be caused by a higher attribute dimension. Since DFCN and MVGRL have extra architectures to extract attribute information, the attribute embeddings of these methods are better learned. However, AGC-DRR is more balanced, it provides more stable performance overall for four compared datasets.

Method	ACM				DBLP			
	C-ACC(%)	NMI(%)	ARI(%)	FI(%)	C-ACC(%)	NMI(%)	ARI(%)	FI(%)
K-means	67.31 ± 0.71	32.44 ± 0.46	30.60 ± 0.69	67.57 ± 0.74	38.65 ± 0.65	11.45 ± 0.38	6.97 ± 0.39	31.92 ± 0.27
AE	81.83 ± 0.08	49.30 ± 0.16	54.64 ± 0.16	82.01 ± 0.08	51.43 ± 0.35	25.40 ± 0.16	12.21 ± 0.43	52.53 ± 0.36
DEC	84.33 ± 0.76	54.54 ± 1.51	60.64 ± 1.87	84.51 ± 0.74	58.16 ± 0.56	29.51 ± 0.28	23.92 ± 0.39	59.38 ± 0.51
IDEC	85.12 ± 0.52	56.61 ± 1.16	62.16 ± 1.50	85.11 ± 0.48	60.31 ± 0.62	31.17 ± 0.50	25.37 ± 0.60	61.33 ± 0.56
GAE	84.52 ± 1.44	55.38 ± 1.92	59.46 ± 3.10	84.65 ± 1.33	61.21 ± 1.22	30.80 ± 0.91	22.02 ± 1.40	61.41 ± 2.23
VGAE	84.13 ± 0.22	53.20 ± 0.52	57.72 ± 0.67	84.17 ± 0.23	58.59 ± 0.06	26.92 ± 0.06	17.92 ± 0.07	58.69 ± 0.07
DAEGC	86.94 ± 2.83	56.18 ± 4.15	59.35 ± 3.89	87.07 ± 2.79	62.05 ± 0.48	32.49 ± 0.45	21.03 ± 0.52	61.75 ± 0.67
ARGA	86.29 ± 0.36	56.21 ± 0.82	63.37 ± 0.86	86.31 ± 0.35	64.83 ± 0.59	29.42 ± 0.92	27.99 ± 0.91	64.97 ± 0.66
ARVGA	83.89 ± 0.54	51.88 ± 1.04	57.77 ± 1.17	83.87 ± 0.55	54.41 ± 0.42	25.90 ± 0.33	19.81 ± 0.42	55.37 ± 0.40
$SDCN_Q$	86.95 ± 0.08	58.90 ± 0.17	65.25 ± 0.19	86.84 ± 0.09	65.74 ± 1.34	35.11 ± 1.05	34.00 ± 1.76	65.78 ± 1.22
SDCN	90.45 ± 0.18	68.31 ± 0.25	73.91 ± 0.40	90.42 ± 0.19	68.05 ± 1.81	39.50 ± 1.34	39.15 ± 2.01	67.71 ± 1.51
MVGRL	86.73 ± 0.76	60.87 ± 1.40	65.07 ± 1.76	86.85 ± 0.72	42.73 ± 1.02	15.41 ± 0.63	8.22 ± 0.50	40.52 ± 1.51
DFCN	90.90 ± 0.20	69.40 ± 0.40	74.90 ± 0.40	90.80 ± 0.20	76.00 ± 0.80	43.70 ± 1.00	47.00 ± 1.50	75.70 ± 0.80
Ours	**92.55 ± 0.09**	**72.89 ± 0.24**	**79.08 ± 0.24**	**92.55 ± 0.09**	**80.41 ± 0.47**	**49.77 ± 0.65**	**55.39 ± 0.88**	**79.90 ± 0.45**
Method	CITE				AMAP			
	C-ACC(%)	NMI(%)	ARI(%)	FI(%)	C-ACC(%)	NMI(%)	ARI(%)	FI(%)
K-means	39.32 ± 3.17	16.94 ± 3.22	13.43 ± 3.02	36.08 ± 3.53	27.22 ± 0.76	13.23 ± 1.33	5.50 ± 0.44	23.96 ± 0.51
AE	57.08 ± 0.13	27.64 ± 0.08	29.31 ± 0.14	53.80 ± 0.11	48.25 ± 0.08	38.76 ± 0.30	20.80 ± 0.47	47.87 ± 0.20
DEC	55.89 ± 0.20	28.34 ± 0.30	28.12 ± 0.36	52.62 ± 0.17	47.22 ± 0.08	37.35 ± 0.05	18.59 ± 0.04	46.71 ± 0.12
IDEC	60.49 ± 1.42	27.17 ± 2.40	25.70 ± 2.65	61.62 ± 1.39	47.62 ± 0.08	37.83 ± 0.08	19.24 ± 0.07	47.20 ± 0.11
GAE	61.35 ± 0.80	34.63 ± 0.65	33.55 ± 1.18	57.36 ± 0.82	71.57 ± 2.48	62.13 ± 2.79	48.82 ± 4.57	68.08 ± 1.76
VGAE	60.97 ± 0.36	32.69 ± 0.27	33.13 ± 0.53	57.70 ± 0.49	74.26 ± 3.63	66.01 ± 3.40	56.24 ± 4.66	70.38 ± 2.98
DAEGC	64.54 ± 1.39	36.41 ± 0.86	37.78 ± 1.24	62.20 ± 1.32	76.44 ± 0.01	65.57 ± 0.03	59.39 ± 0.02	69.97 ± 0.02
ARGA	61.07 ± 0.49	34.40 ± 0.71	34.32 ± 0.70	58.23 ± 0.31	69.28 ± 2.30	58.36 ± 2.76	44.18 ± 4.41	64.30 ± 1.95
ARVGA	59.31 ± 1.38	31.80 ± 0.81	31.28 ± 1.22	56.05 ± 1.13	61.46 ± 2.71	53.25 ± 1.91	38.44 ± 4.69	58.50 ± 1.70
$SDCN_Q$	61.67 ± 1.05	34.39 ± 1.22	35.50 ± 1.49	57.82 ± 0.98	35.53 ± 0.39	27.90 ± 0.40	15.27 ± 0.37	34.25 ± 0.44
SDCN	65.96 ± 0.31	38.71 ± 0.32	40.17 ± 0.43	63.62 ± 0.24	53.44 ± 0.81	44.85 ± 0.83	31.21 ± 1.23	50.66 ± 1.49
MVGRL	68.66 ± 0.36	43.66 ± 0.40	44.27 ± 0.73	63.71 ± 0.39	45.19 ± 1.79	36.89 ± 1.31	18.79 ± 0.47	39.65 ± 2.39
DFCN	**69.50 ± 0.20**	**43.90 ± 0.20**	**45.50 ± 0.30**	64.30 ± 0.20	76.88 ± 0.80	69.21 ± 1.00	58.98 ± 0.84	71.58 ± 0.31
Ours	68.32 ± 1.83	43.28 ± 1.41	45.34 ± 2.33	**64.82 ± 1.60**	**78.11 ± 1.69**	**72.21 ± 1.63**	**61.15 ± 1.65**	**72.72 ± 0.97**

Table 3: Node clustering performance on four datasets (mean ± std). Best results are **bold** values and the second best values are underlined.

Dataset	Model	C-ACC(%)	NMI(%)	ARI(%)	F1(%)
ACM	w/o min-max	92.0±0.1	71.5±0.3	77.6±0.3	92.0±0.1
	w/o $\mathcal{L}_{MSE}$	85.7±4.0	59.2±5.3	63.8±7.0	85.6±4.2
	Ours	**92.6±0.1**	**72.9±0.2**	**79.1±0.2**	**92.6±0.1**
DBLP	w/o min-max	64.8±5.6	33.7±2.7	31.6±3.5	64.8±5.6
	w/o $\mathcal{L}_{MSE}$	58.0±6.9	29.5±5.9	30.1±7.5	51.8±7.8
	Ours	**80.4±0.5**	**49.8±0.7**	**55.4±0.9**	**79.9±0.5**
CITE	w/o minmax	63.9±4.5	39.9±2.5	40.2±3.6	60.8±4.2
	w/o $\mathcal{L}_{MSE}$	61.0±6.4	37.5±3.7	36.7±5.7	57.8±5.6
	Ours	**68.3±1.8**	**43.3±1.4**	**45.3±2.3**	**64.8±1.6**
AMAP	w/o minmax	73.4±4.0	66.8±3.8	56.2±3.5	68.1±5.9
	w/o $\mathcal{L}_{MSE}$	72.2±6.7	69.2±3.3	56.3±4.9	65.3±8.1
	Ours	**78.1±1.7**	**72.2±1.6**	**61.2±1.7**	**72.7±1.0**

Table 4: Ablation for each component. w/o min-max and w/o $\mathcal{L}_{MSE}$ indicate that the method with the min-max optimization mechanism and the MSE objective being removed, respectively.

4.3 Ablation Studies

In this section, we conduct ablation studies to verify the superiority of each component in our method. The w/o min-max method and the w/o $\mathcal{L}_{MSE}$ method indicate the network with the proposed min-max optimization mechanism and the MSE objective being removed, respectively. From the results in Table 4, some observations can be summarized. AGC-DRR exceeds the method w/o minmax by 0.6%, 15.6%, 4.4%, and 4.7% accuracy increments, and exceeds the method w/o $\mathcal{L}_{MSE}$ by 6.9%, 22.4%, 7.3%, and 5.9% accuracy increments on ACM, DBLP, CITE, and AMAP, respectively. These results demonstrate that both components play an essential role in our method for improving the clustering performance. On the one hand, the min-max optimization mechanism could reduce the redundant information in the input space to ensure the diversity of the compared views for a more reliable structure augmented graph. On the other hand, $\mathcal{L}_{MSE}$ could achieve latent space redundancy reduction to obtain more discriminative node representations. Both components contribute to better clustering performance.

4.4 Effect of Different Graph Structure

To verify the robustness of AGC-DRR, we utilize three types of adjacent matrices as input. In our settings, ADJ indicates the normalized adjacent matrix $\bar{\mathbf{A}}$. RW indicates the adjacent matrix constructed by random walk [Grover and Leskovec, 2016]. PPR indicates the adjacent matrix constructed by Personalized PageRank algorithm [Hassani and Khasahmadi, 2020]. As illustrated in Fig. 3, these clustering results on two benchmark datasets have clearly verified that AGC-DRR could achieve robust performance when adopting different adjacent matrices.

4.5 t-SNE Visualization

As illustrated in Fig. 4, we present the clustering results of different clustering methods on ACM and DBLP by t-SNE algorithm [Van der Maaten and Hinton, 2008]. From these figures, we observe that the proposed AGC-DRR can clearly reveal the intrinsic clustering structure among samples.

5 Conclusion

In this paper, we design a novel attributed graph clustering with dual redundancy reduction (AGC-DRR), which can reduce the redundant information in both input and latent feature spaces. In the proposed method, the learning processes of structure augmented graph and clustering are united into a common min-max optimization framework. In this way, the learned network is robust against perturbation while discriminative against inter-class samples. The proposed AGC-DRR has been evaluated on four benchmark datasets. Extensive experimental results verify that our proposed method outperforms state-of-the-art counterparts.

Acknowledgments

This work is supported by the National Key R&D Program of China (2020AAA0107100), the National Natural Science Foundation of China (Grant No. 61922088, 62006237).

References

[Bo *et al.*, 2020] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. Structural deep clustering network. In *Proc. of WWW*, 2020.

[Caron *et al.*, 2018] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proc. of ECCV*, 2018.

[Grover and Leskovec, 2016] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016.

[Guo *et al.*, 2017] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. Improved deep embedded clustering with local structure preservation. In *IJCAI*, 2017.

[Hassani and Khasahmadi, 2020] Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *Proc. of ICML*, 2020.

[Hinton and Salakhutdinov, 2006] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 2006.

[Hjelm *et al.*, 2018] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *Proc. of ICLR*, 2018.

[Kipf and Welling, 2016a] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Kipf and Welling, 2016b] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[Pan *et al.*, 2020] S. Pan, R. Hu, S. F. Fung, G. Long, J. Jiang, and C. Zhang. Learning graph embedding with adversarial training methods. *IEEE Transactions on Cybernetics*, 2020.

[Schroff *et al.*, 2015] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proc. of CVPR*, 2015.

[Shchur *et al.*, 2018] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

[Suresh *et al.*, 2021] Susheel Suresh, Pan Li, Cong Hao, and Jennifer Neville. Adversarial graph augmentation to improve graph contrastive learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[Tishby *et al.*, 2000] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[Tschannen *et al.*, 2019] Michael Tschannen, Josip Djolonga, Paul K Rubenstein, Sylvain Gelly, and Mario Lucic. On mutual information maximization for representation learning. In *Proc. of ICLR*, 2019.

[Tu *et al.*, 2021] Wenxuan Tu, Sihang Zhou, Xinwang Liu, Xifeng Guo, Zhiping Cai, En Zhu, and Jieren Cheng. Deep fusion clustering network. In *Proc. of AAAI*, 2021.

[Van der Maaten and Hinton, 2008] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 2008.

[Velickovic *et al.*, 2019] Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *ICLR (Poster)*, 2019.

[Wang *et al.*, 2019] C Wang, S Pan, R Hu, G Long, J Jiang, and C Zhang. Attributed graph clustering: A deep attentional embedding approach. In *Proc. of IJCAI*, 2019.

[Wong, 1979] J. A. Hartiganm. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society*, 1979.

[Wu *et al.*, 2020] Tailin Wu, Hongyu Ren, Pan Li, and Jure Leskovec. Graph information bottleneck. *Neural Information Processing Systems (NeurIPS)*, 2020.

[Xie *et al.*, 2016] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, 2016.

[Xu *et al.*, 2019] Cai Xu, Ziyu Guan, Wei Zhao, Hongchang Wu, Yunfei Niu, and Beilei Ling. Adversarial incomplete multi-view clustering. In *IJCAI*, 2019.

[Yu *et al.*, 2020] Junchi Yu, Tingyang Xu, Yu Rong, Yatao Bian, Junzhou Huang, and Ran He. Graph information bottleneck for subgraph recognition. In *Proc. of ICLR*, 2020.

[Zhang *et al.*, 2021] Junning Zhang, Qunxing Su, Bo Tang, Cheng Wang, and Yining Li. Dpsnet: Multitask learning using geometry reasoning for scene depth and semantics. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[Zhao *et al.*, 2020] Tong Zhao, Yozen Liu, Leonardo Neves, Oliver Woodford, Meng Jiang, and Neil Shah. Data augmentation for graph neural networks. *arXiv preprint arXiv:2006.06830*, 2020.

[Zhao *et al.*, 2021] Han Zhao, Xu Yang, Zhenru Wang, Erkun Yang, and Cheng Deng. Graph debiased contrastive learning with joint representation clustering. In *Proc. IJCAI*, 2021.

[Zhong *et al.*, 2016] Haoti Zhong, Hao Li, Anna Cinzia Squicciarini, Sarah Michele Rajtmajer, Christopher Griffin, David J Miller, and Cornelia Caragea. Content-driven detection of cyberbullying on the instagram social network. In *IJCAI*, 2016.