Multi-order Filter Fusion and Triple Contrastive Learning for Attribute Graph Clustering

Tianyou Yu[1,†], Kailin Zhang[1,†], Jianlin Zheng[1,†], Tianruo Liu[2], Tianxiang Zhao[1,*], Haopeng Yang[1], and Jipeng Guo[1,*]
[1]College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China
[2]Duke University, Durham, North Carolina, USA
{2023040091,2023040107,2023030200}@buct.edu.cn, tl411@duke.edu, {tianxiangzhao,2025200767,guojipeng}@buct.edu.cn

***Abstract*—Deep contrastive graph clustering has attracted growing attention due to its powerful self-supervised representation learning paradigm on attributed graphs. However, two challenges still hinder further improvements. First, most existing methods heavily rely on the observed adjacency in neighboring information propagation, which fails to capture attribute-induced global semantics and becomes fragile under sparse, incomplete, or noisy topology. Second, most existing methods employ a single contrastive strategy, resulting in shallow self-supervised representation learning. To this end, this paper proposes Multi-order Filtering Fusion (MFF) and Triple Contrastive Learning (TCL) for Attribute Graph Clustering (MT-AGC), a semantic feature-enhanced topology augmentation with comprehensive contrastive learning. Specifically, MFF constructs an attribute-similarity-induced global semantic graph and fuses it with the original topology, then performs multi-order low-pass filtering with self-attentive fusion to obtain discriminative multi-scale embedding representation. Further, TCL jointly optimizes representation contrastive consistency, structural consistency, and pseudo-label guided semantic matching, achieving thorough self-supervised training. More importantly, a two-stage strategy introduces high-confidence pseudo-labels to form a closed and reliable feedback loop in crucial semantic contrastive learning. Extensive experiments on six benchmark datasets demonstrate the effectiveness of MT-AGC, and ablation results and visualizations further verify the contribution of each component. The code could be available at https://github.com/Liu-yi-ian/GCNormer.**

***Index Terms*—Contrastive Graph Clustering, Topology Augmentation, Multi-order Filter Fusion, Triple Contrastive Learning, Pseudo-label Feedback.**

I. Introduction

With the rapid development of the Internet and the Internet of Things, massive real-world data naturally arise in the form of graphs, where nodes represent entities, edges describe mutual interactions, and nodes are associated with attribute features. As a fundamental unsupervised analysis technique, clustering aims to partition samples into disjoint groups according to intrinsic similarity, thereby revealing underlying complex data structures [1]–[4]. For attributed graphs, effective clustering should jointly exploit topological relations and semantic correlations from node attributes. Graph Neural Networks (GNNs) are widely used to learn discriminative embedding representation for graph [5]–[9].

Deep Graph Clustering (DGC) integrates deep graph embedding representation learning into clustering and has achieved remarkable progress on attributed graphs [10]–[13]. Existing DGC methods can be roughly categorized into generative-based, adversarial-based, and contrastive-based approaches. Generative-based methods usually learn embeddings by reconstructing attributes or structures with graph auto-encoders [14]–[17], while adversarial-based methods introduce adversarial training to regularize representation distributions and improve robustness [18]–[20]. However, these methods often rely on specific priors or initialization (e.g., cluster centers and prior distributions), making them sensitive and costly to tune in unsupervised settings.

Recently, contrastive learning has driven contrastive graph clustering, which constructs self-supervised augmentation signals and learns more discriminative representations by exploring multi-view consistency with less reliance on external priors [21]–[24]. Prior studies have explored multi-view self-supervision [25], [26], learnable or robust augmentation [24], [27], [28], and cluster-guided training strategies [29]–[31]. Despite these advances, most methods still suffer from two issues that limit clustering performance and optimization stability:

(1) Over-reliance on observed adjacency and insufficient attribute-induced global semantics. Many methods build information propagation on the observed adjacency [23], [24]. When original topology is sparse, incomplete, or noisy, semantically similar yet topologically distant nodes are poorly connected, resulting in embedding representation learning that over-emphasize local smoothness while missing global semantic structure. and inevitably reducing discriminability.

(2) Single and one-sided self-supervised training mechanism and unreliable clustering feedback. Contrastive clustering is sensitive to the construction of self-supervised mechanism [27], [28]. In fact, representation, structure, and semantic labels are the three key factors in graph clustering. However, existing methods only establish self-supervision on some of these factors, resulting in insufficient optimization. And, clustering feedback is hard to utilize reliably: introducing pseudo labels too early yields noisy supervision and oscillations, whereas ignoring clustering outputs makes it difficult to inject cluster-level structure into representation learning, limiting

†These authors contributed equally to this work.
*Corresponding authors.

This research was supported by 62403043; in part by the Interdisciplinary Research Center of Beijing University of Chemical Technology under Grant XK2025-06 and Key Project of Sichuan Science and Technology Innovation and Entrepreneurship Seeding Program under Grant 2024JDRC0084.

discriminability [29], [32], [33].

To address these issues, this study proposes Multi-order Filtering Fusion (MFF) and Triple Contrastive Learning (TCL) for Attribute Graph Clustering (MT-AGC), which achieves attribute semantics guided topology augmentation and comprehensive self-supervised training with reliable pseudo-label feedback. Concretely, MFF constructs an attribute-similarity-induced global graph and fuses it with the original topology, then performs multi-order low-pass filtering with self-attentive fusion to obtain multi-scale structural-semantic representation. Built upon these representations, TCL jointly optimizes contrastive representation consistency, structure consistency, and pseudo-label guided semantic matching, forming a collaborative and complementary self-supervised training. To fully utilize pseudo-label information to improve optimization stability, a two-stage strategy introduces high-confidence pseudo labels after pre-training epochs to provide a closed and reliable feedback loop.

The main contributions are summarized as follows:

- A feature-enhanced **MFF** module is proposed, which augments the observed topology with an attribute-similarity-induced global graph and self-attentively fuses multi-order low-pass filtered signals to obtain information-rich multi-scale graph embedding representations.
- A unified comprehensive **TCL** scheme is developed, which integrates representation alignment, structural consistency, and pseudo-label guided semantic matching among augmented contrastive views. And, TCL adopts a two-stage high-confidence pseudo-label strategy to provide reliable clustering feedback.
- Extensive experiments on six benchmark datasets demonstrate that MT-AGC consistently outperforms diverse baselines, and further analyses verify the effectiveness of each component with improved training stability and robust clustering results.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents the proposed framework and optimization objectives. Section IV reports experimental settings and results with extensive analyses. Finally, Section V concludes the paper.

II. Related Work

DGC is a fundamental yet challenging unsupervised task, aiming to learn semantic representations from graphs and partition nodes into clusters. Existing DGC methods are commonly grouped into three categories: generative-based, adversarial-based, and contrastive-based. Following this taxonomy, we briefly review generative/adversarial approaches and then focus on contrastive-based methods; more comprehensive discussions can be found in survey papers.

A. Generative and Adversarial Deep Graph Clustering

Most generative/adversarial DGC methods adopt GNNs as backbones and optimize self-supervised objectives built with certain priors. Kipf et al. propose Graph Auto-encoder (GAE) [14] with Graph Convolutional Network (GCN) [6] encoder and reconstruction loss. Deep Attention Embedded Graph Clustering (DAEGC) [15] combines Graph Attention network (GAT) [7] with GAE [14] and introduces a self-training clustering loss with KL-divergence to enhance discriminability. Adversarially Regularized Graph Autoencoder (ARGA) [18], [34] and Adversarial Graph AutoEncoders (AGAE) [19] further incorporate adversarial training [35], [36] to regularize embeddings towards a prior distribution. To alleviate over-smoothing and better fuse attributes with topology, Structural Deep Clustering Network (SDCN) [16] and Deep Fusion Clustering Network (DFCN) [17] design attribute–structure deep fusion modules to improve representation learning and clustering.

Despite their effectiveness, many of these methods rely on distribution alignment losses [37] that match embeddings to pre-learned cluster centers, making performance sensitive to initialization and often requiring non-trivial pre-training and tuning. In contrast, graph contrastive learning has become increasingly popular in DGC for its reduced dependence on prior distributions and generally stable and efficient training.

B. Contrastive Deep Graph Clustering

Driven by graph contrastive learning, contrastive-based DGC methods have achieved strong clustering performance on attribute graphs. Most of them follow the paradigm of first constructing augmented views and then performing contrastive representation learning [38], [39], where the effectiveness largely depends on (i) view generation and (ii) self-supervised objectives.

1) Data Augmentation. Many methods generate multiple views via handcrafted augmentations. Contrastive Multi-View Representation Learning on (MVGRL) [21] and Dual Correlation Reduction Network (DCRN) [40] adopt diffusion-based views, while Self-supervised Contrastive Attributed Graph Clustering (SCAGC) [41] perturbs topology by randomly adding/deleting edges; DCRN and SCAGC also augment node attributes via perturbation. Although effective, such augmentations can be complex and may suffer from coupling between aggregation and transformation, limiting efficiency. To address this, learnable augmenters are introduced, e.g., the perturb network in CONtrastiVe Graph ClustEring network with Reliable AugmenTation (CONVERT) [27] and the attribute/structure augmenters in Graph Node Clustering with Fully Learnable Augmentation (GraphLearner) [24], though they can still be complex and tightly coupled with time-consuming representation learning. Some preliminary explorations attempt to simplify this pipeline [23], [29]. Notably, Simple Contrastive Graph Clustering (SCGC) [23] uses a parameter-free high-order graph Laplacian filter for embeddings and employs MLP encoders to form flexible augmented views, achieving strong empirical performance. Inspired by these filtering-based and augmentation-light designs, our work further introduces feature-induced global semantic relations and fuses multi-order filtered signals via self-attention, providing a stronger multi-scale structural–semantic representation backbone.

2) Self-supervised Training Objective. Contrastive-based DGC also hinges on positive/negative pair construction and contrastive losses. For pair construction, MVGRL [21] treats different views of the same node as positives and forms negatives by feature shuffling; DCRN [40] aligns the same node across views while separating different nodes under a decorrelation constraint. Typical objectives include the MSE loss in DCRN [40], the InfoMAX loss in MVGRL [21], and the InfoNCE loss in SCAGC [41]. While InfoNCE is widely used, it relies on exponential similarity computations and may introduce false negatives in clustering scenarios. Moreover, reliably incorporating cluster-level feedback remains nontrivial: early pseudo labels are often noisy, while ignoring clustering outputs limits semantic discriminability. Motivated by these limitations, we adopt a unified training scheme with semantic contrastive consistency, structural consistency, and pseudo-label guided semantic matching, where high-confidence pseudo labels are introduced in a later stage to improve both discriminability and optimization stability.

III. Proposed Method

The method details of the proposed **MT-AGC** are elaborated.As illustrated in Fig. 1, MT-AGC mainly consists of two modules: MFF module and TCL module.

A. Notations

An attributed graph is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, where $\mathcal{V} = \{v_1, \dots, v_N\}$ is the node set, $\mathcal{E}$ is the edge set, and $\mathbf{X} \in \mathbb{R}^{N\times D}$ is the attribute feature matrix. The graph topology is represented by an adjacency matrix $\mathbf{A} \in \{0,1\}^{N\times N}$, where $A_{ij} = 1$ if $(v_i, v_j) \in \mathcal{E}$, and $A_{ij} = 0$ otherwise.

To incorporate feature-induced global semantics, a feature-based KNN graph $\mathbf{A}_k$ is constructed and fused with the original adjacency matrix to obtain the fused graph $\mathbf{A}_f$. Throughout this paper, a hat over a matrix (e.g., $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$) indicates the addition of self-connections, while a tilde denotes symmetric normalization, i.e., $\tilde{\mathbf{A}} = \hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}$, where $\hat{\mathbf{D}}$ is the corresponding degree matrix.

The main notations used in MT-AGC are summarized in Table I to improve readability.

TABLE I
THE EXPLANATION OF MAIN NOTATIONS.

Notation	Meaning
$\mathbf{X} \in \mathbb{R}^{N\times D}$	Attribute feature matrix
$\mathbf{X}_f \in \mathbb{R}^{N\times D}$	Multi-order low-pass filter fusion representation
$\mathbf{A} \in \{0,1\}^{N\times N}$	Adjacency matrix of the original graph
$\mathbf{A}_k \in \{0,1\}^{N\times N}$	Feature-induced KNN adjacency matrix
$\mathbf{A}_f \in \{0,1\}^{N\times N}$	Fused adjacency matrix
$\mathbf{I} \in \{0,1\}^{N\times N}$	Identity matrix
$\mathbf{D}, \hat{\mathbf{D}} \in \mathbb{R}^{N\times N}$	Degree matrices
$\mathbf{F}_L^t \in \mathbb{R}^{N\times N}$	t-order filtered node features on the original graph
$\mathbf{F}_J^k \in \mathbb{R}^{N\times N}$	k order filtered node features on the fused graph
$\mathbf{Z}^k \in \mathbb{R}^{N\times d_L}$	Node embedding in the k-th augmented view
$\mathbf{S} \in \mathbb{R}^{N\times N}$	Cross-view similarity matrix
$\mathbf{Z} \in \mathbb{R}^{N\times d_L}$	Final node embedding representation for clustering

B. Multi-order Filter Fusion Module

The MFF module enriches the original graph structure by integrating attribute-induced semantic relationships. It first constructs a feature-similarity-based K-nearest neighbor (KNN) graph, denoted as $\mathbf{A}_k$, to complement observed topology $\mathbf{A}$.

Given the attribute feature matrix $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times D}$, we compute the pairwise affinity via cosine similarity:

$$S_{ij}^1 = \cos(\mathbf{x}_i, \mathbf{x}_j), \quad i \neq j. \tag{1}$$

For each node i, its K-nearest neighbors are selected according to the Top-K similarity scores, yielding a feature-based KNN graph with adjacency matrix $\mathbf{A}_k$, which is further symmetrized to ensure undirectedness. The resulting $\mathbf{A}_k$ captures global semantic associations induced by the feature space. The fused adjacency matrix is then obtained by $\mathbf{A}_f = \mathbf{A} + \mathbf{A}_k$, followed by binarization in practice.

Based on the feature-enhanced fused structure, we perform multi-order graph Laplacian low-pass filtering to obtain multi-scale representations. We first add self-connections and apply symmetric normalization to $\mathbf{A}$ and $\mathbf{A}_f$, yielding normalized adjacency matrices $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{A}}_f$. Then, multi-order filtered features are generated by powers of the propagation operator. On the original graph, the t-th order filtered features are defined as:

$$\mathbf{F}_L^t = \big(\tilde{\mathbf{A}}\big)^t\mathbf{X}, \quad t = 0, 1, \cdots, T, \tag{2}$$

where $t = 0$ corresponds to the raw features. Similarly, on the fused graph, the k-th order filtered features are:

$$\mathbf{F}_J^k = \big(\tilde{\mathbf{A}}_f\big)^k\mathbf{X}, \quad k = 0, 1, \cdots, K. \tag{3}$$

This dual-stream filtering strategy captures representations at different propagation depths from two structural perspectives: low-order filtering encodes local neighborhood consistency, while high-order filtering facilitates information exchange among semantically similar yet topologically distant nodes via feature-induced global connections.

To adaptively fuse features from different orders and avoid redundancy or over-smoothing, we introduce a self-attention weighting mechanism. The final representation is defined as:

$$\mathbf{X}_f = \sum_{t=0}^{T} \alpha_t\, \mathbf{F}_L^t + \sum_{k=1}^{K} \alpha_{k+T}\, \mathbf{F}_J^k, \tag{4}$$

where $\{\alpha_0, \cdots, \alpha_T, \alpha_{T+1}, \cdots, \alpha_{K+T}\}$ are learnable weights that automatically emphasize the most relevant orders for clustering. This allows MT-AGC to balance local discriminability (low-order) with broader semantic dependencies (high-order), without requiring manual order selection.

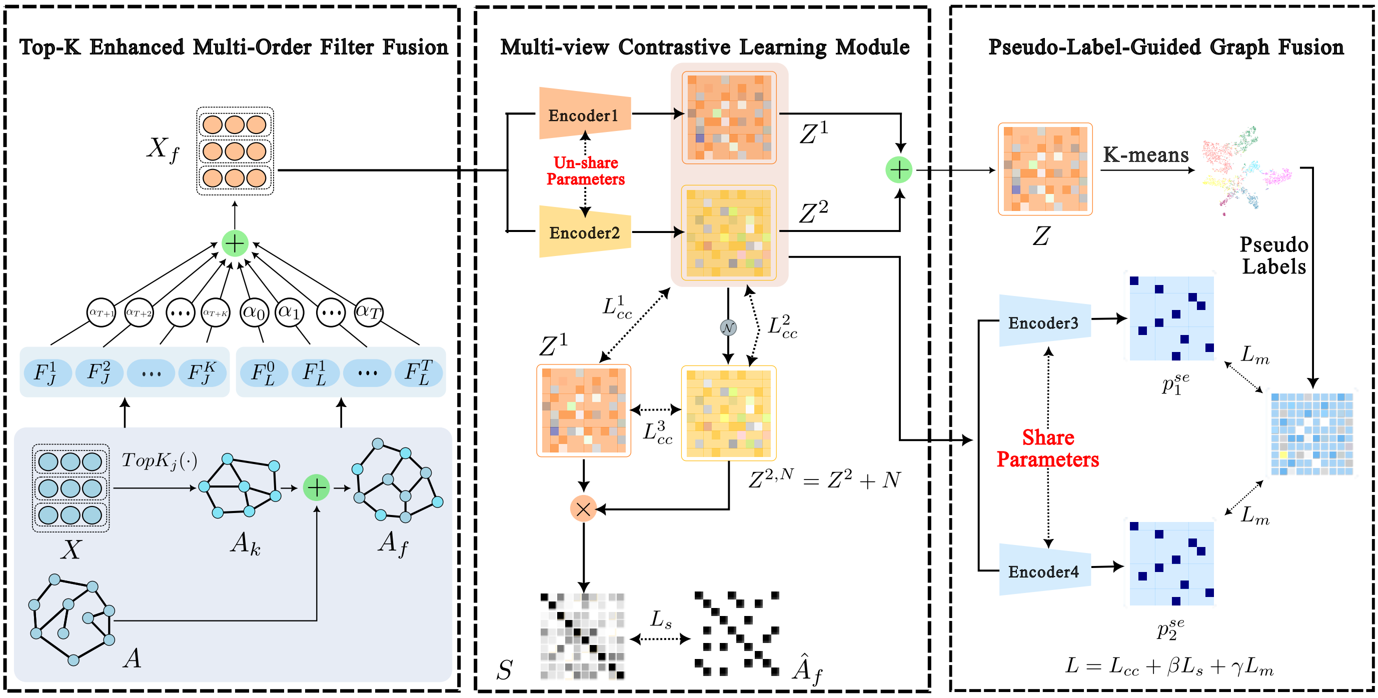


Fig. 1. The overall framework of MT-AGC consists of two main modules: (1) the Multi-order Filter Fusion (MFF) module, which produces a comprehensive node representation X_f and (2) the Triple-contrastive Learning (TCL) module, which refines the embeddings via structural, semantic, and clustering contrastive objectives. The final clustering results are obtained by performing K-means on the refined node embeddings.

C. Triple Contrastive Learning Module

Representation Augmentation: After obtaining the multi-order filter fused representation $\mathbf{X}_f$, we formulate attribute graph clustering as a unified triple contrastive learning problem, where representations are aligned at three complementary semantic levels, namely semantic-level, structural-level, and clustering-level. Specifically, we first perform representation augmentation by feeding $\mathbf{X}_f$ into two MLP projection heads with identical architectures but independent parameters, yielding two low-dimensional embedding views:

$$\mathbf{Z}^1 = \mathbf{MLP}_1(\mathbf{X}_f), \qquad \mathbf{Z}^2 = \mathbf{MLP}_2(\mathbf{X}_f). \tag{5}$$

Since the two MLPs have different parameter initializations and optimization trajectories, their outputs can be regarded as the representations of the same nodes from two embedding views.To ensure that $\mathbf{Z}^1$ and $\mathbf{Z}^2$ share the same scale and range, we normalize them as follows:

$$\mathbf{Z}^1 = \frac{\mathbf{Z}^1}{\|\mathbf{Z}^1\|_2} \in \mathbb{R}^{N\times d_L}, \qquad \mathbf{Z}^2 = \frac{\mathbf{Z}^2}{\|\mathbf{Z}^2\|_2} \in \mathbb{R}^{N\times d_L}. \tag{6}$$

To improve robustness and avoid representation collapse, we inject Gaussian noise into the second view to generate a perturbed embedding:

$$\mathbf{Z}^{2,N} = \mathbf{Z}^2 + \mathbf{N}, \tag{7}$$

where $\mathbf{N} \in \mathbb{R}^{N\times d_L}$ is randomly sampled from the Gaussian distribution $\mathcal{N}(0, \sigma)$.

Based on three representations $\mathbf{Z}^1$, $\mathbf{Z}^2$, and $\mathbf{Z}^{2,N}$, MT-AGC performs triple contrastive learning as detailed below:

Representation-level Contrastive Learning: The first contrastive objective operates at the representation level, aiming to enhance instance-wise discriminability and robustness against noise. Conventional graph contrastive learning typically employs an InfoNCE-style objective where all other nodes are treated as negatives. However, in clustering scenarios, this introduces false negative bias, as semantically similar nodes (i.e., those in the same cluster) are inadvertently pushed apart.

To mitigate this, we discard inter-node negative sampling and instead construct an instance-specific triplet objective. For each node, the clean view from the other branch serves as the positive sample, while its own noise-perturbed variant serves as the exclusive negative sample. The loss is defined as:

$$\mathcal{L}_{sem} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\mathbf{z}_i^1,\mathbf{z}_i^2)}}{e^{s(\mathbf{z}_i^1,\mathbf{z}_i^2)} + e^{s(\mathbf{z}_i^1,\mathbf{z}_i^{2,N})} + e^{s(\mathbf{z}_i^2,\mathbf{z}_i^{2,N})}}, \tag{8}$$

where $s(\mathbf{a}, \mathbf{b}) = \cos(\mathbf{a}, \mathbf{b})/\tau$ denotes the temperature-scaled cosine similarity.

In Eq. (8), the numerator aligns the positive pair $(\mathbf{z}_i^1, \mathbf{z}_i^2)$, while the denominator suppresses the representation similarity with the noisy variants $(\mathbf{z}_i^1, \mathbf{z}_i^{2,N})$ and $(\mathbf{z}_i^2, \mathbf{z}_i^{2,N})$. This mechanism encourages the model to distinguish intrinsic semantic features from random noise distributions without penalizing intra-cluster similarities, thereby learning more robust and instance-discriminative representations.

Structural-level Contrastive Learning: The second contrastive objective operates at the structural level, aiming to align the similarity geometry in embedding space with the feature-enhanced fused graph structure.We compute the cross-view similarity matrix:

$$S_{ij}^2 = \cos\left(\mathbf{z}_i^1, \mathbf{z}_j^{2,N}\right). \tag{9}$$

Based on this, the structural loss enforces the pairwise similarity relations learned in the embedding space to be consistent with the fused graph structure. It is formulated as follows:

$$\mathcal{L}_{str} = \frac{1}{N^2}\left\|\mathbf{S}^2 - \hat{\mathbf{A}}_f\right\|_F^2, \tag{10}$$

where S_{ij}^2 denotes the cosine similarity between the *clean* embedding $\mathbf{z}_i^1$ and the *perturbed* embedding $\mathbf{z}_j^{2,N}$.

Crucially, this design functions as a denoising structural constraint: by minimizing the discrepancy between the noise-corrupted similarity $\mathbf{S}^2$ and the fixed adjacency $\hat{\mathbf{A}}_f$, the model is compelled to recover authentic topological patterns even from noisy representations. Furthermore, since the subsequent contrastive loss lacks inter-node negative pairs, this structural alignment strictly constrains the global geometric distribution of nodes, effectively preventing the trivial solution (mode collapse) where all nodes map to a single point.

Semantic-level Contrastive Learning: To inject explicit clustering structure into the representation learning, we design a delayed pseudo label guidance strategy that operates only after the model has reached a stable training stage. This avoids the noise amplification that would occur if unreliable pseudo-labels were introduced too early.

Once the training epoch exceeds a preset threshold T_{stage}, we first fuse the two embedding views by averaging:

$$\mathbf{Z} = \frac{1}{2}\left(\mathbf{Z}^1 + \mathbf{Z}^2\right). \tag{11}$$

On this fused embedding $\mathbf{Z}$, we perform K-means to obtain an initial set of pseudo labels $\mathbf{p} \in \{1, \ldots, K\}^N$.

To reduce the influence of uncertain samples, we retain only high confidence pseudo-labels. For each cluster c, we compute the Euclidean distance between every node assigned to c-th cluster centroid μ_c. From each cluster, the top η proportion of samples with the smallest distances are selected, forming a high confidence label set $\overline{\mathbf{Y}}$. This process can be summarized as follows:

$$\overline{\mathbf{Y}} = \mathrm{top}(\mathbf{p}), \tag{12}$$

where $\mathrm{top}(\cdot)$ denotes the selection of the most confident η fraction of samples per cluster based on centroid proximity.

Simultaneously, each view's embeddings are passed through a lightweight MLP followed by a softmax layer to produce semantic class-distribution predictions:

$$\mathbf{p}_i^{se} = \mathrm{Softmax}(\mathbf{MLP}(\mathbf{Z}^i)), \quad i \in \{1, 2\}. \tag{13}$$

Finally, the clustering-level contrastive loss is defined as the cross-entropy between the predicted distributions and the high-confidence pseudo-labels:

$$\mathcal{L}_{clu} = \mathrm{CE}\big(\mathbf{p}_i^{se}, \overline{\mathbf{Y}}\big), \quad i \in \{1, 2\}, \tag{14}$$

where $\mathrm{CE}(\cdot,\cdot)$ denotes the cross entropy loss.

TABLE II
DATASET STATISTICS AND HYPER-PARAMETER SETTINGS.

Dataset	Node	Dim	Edge	Cls	Hom.	LR	Top-K	d_L	T_{stage}
CORA	2708	1433	5429	7	0.81	$1e$-3	10	1200	200
CITESEER	3327	3703	4732	6	0.74	$2e$-4	5	1000	200
AMAP	7650	745	119081	8	0.82	$1e$-3	12	800	300
BAT	131	81	1038	4	0.43	$4e$-3	3	800	200
EAT	399	203	5994	4	0.41	$2e$-3	20	800	200
UAT	1190	239	13599	4	0.70	$5e$-4	15	600	300

This two-stage design ensures that pseudo-label supervision is activated only when the embeddings are sufficiently discriminative, thereby providing reliable clustering feedback that refines the semantic boundaries without being misled by early-training noise.

D. Objective Function and Clustering

By integrating the three contrastive objectives, the overall training loss of MT-AGC is formulated as:

$$\mathcal{L} = \mathcal{L}_{sem} + \beta\,\mathcal{L}_{str} + \gamma\,\mathcal{L}_{clu}, \tag{15}$$

where β and γ are non-negative trade-off hyperparameters balancing the importance of the three terms in Eq. (15). Through joint optimization, MT-AGC progressively learns representations that are semantically discriminative, structurally consistent, and clustering-aligned, enabling reliable attribute graph clustering without external supervision. After training, for computational efficiency and fair comparison, we conduct the K-means algorithm on the fused node embedding $\mathbf{Z}$ to obtain the final clustering assignments.

IV. EXPERIMENTAL RESULTS

In this section, extensive experiments are conducted to comprehensively verify the effectiveness of the proposed MT-AGC. This section first introduces some fundamental experimental settings, including benchmark datasets, parameter setting, and clustering performance metrics. Then, the fine-grained analyses of the experimental results are provided. Finally, several auxiliary experiments are provided to intuitively understand performance improvement, including ablation study and visualization analysis. The experimental environment is a laptop equipped with an Intel(R) Core(TM) i5-13500HX CPU, an NVIDIA GeForce RTX 4060 GPU, 16 GB of RAM, and the PyTorch deep learning framework.

A. Experimental Setting

*1) **Benchmark Datasets**:* The proposed MT-AGC is evaluated on six available graph benchmark datasets, including CORA [42], CITESEER [42], AMAP [40], BAT [23], EAT [23], and UAT [23]. The detailed statistics of these datasets are briefly summarized in Table II.

*2) **Baselines**:* The comparison baselines mainly cover three categories of mainstream graph clustering methods: three generative-based graph clustering approaches (including classic attributed graph clustering: DAEGC [15], SDCN [16], and DFCN [17]), one adversarial-based graph clustering method

TABLE III

PERFORMANCE COMPARISON ON SIX DATASETS. RESULTS ARE REPORTED AS MEAN ± STD OVER TEN RUNS. RED AND BLUE VALUES INDICATE THE BEST AND SUB-OPTIMAL RESULTS. OOM INDICATED OUT-OF-MEMORY DURING TRAINING.

Dataset	Metric	DAEGC	SDCN	DFCN	ARGA	AGE	MVGRL	GDCL	AutoSSL	MVGC	DCRN	AFGRL	NCLA	SCCGC	SCGC	CCGC	GL	MCGC	Ours
CORA	ACC	70.43±0.36	35.60±2.83	36.33±0.49	71.04±0.25	73.50±1.83	70.47±3.70	70.83±0.47	63.81±0.57	42.85±1.13	61.93±0.47	26.25±1.24	51.09±1.25	74.45±0.02	73.88±0.88	73.88±1.20	74.91±1.78	74.15±0.37	78.27±1.83
	NMI	52.89±0.69	14.28±1.91	19.36±0.87	51.06±0.52	57.58±1.42	55.57±1.54	56.60±0.36	47.62±0.45	24.11±1.00	45.13±1.57	12.36±1.54	31.80±0.78	57.80±0.03	56.10±0.72	56.45±1.04	58.16±0.83	56.47±1.64	59.68±1.03
	ARI	49.63±0.43	07.78±3.24	04.67±2.10	47.71±0.33	50.60±2.14	48.70±3.94	48.05±0.72	38.92±0.77	14.33±1.26	33.15±0.14	14.32±1.87	36.66±1.65	55.36±0.01	51.79±1.59	52.51±1.89	53.82±2.25	51.87±0.90	58.82±2.06
	F1	68.27±0.57	24.37±1.04	26.16±0.50	69.27±0.39	69.68±1.59	67.15±1.86	52.88±0.97	56.42±0.21	35.16±0.91	49.50±0.42	30.20±1.15	51.12±1.12	65.52±0.02	70.81±1.96	70.98±2.79	73.33±1.86	70.57±1.61	76.31±2.11
CITESEER	ACC	64.54±1.39	65.96±0.31	69.50±0.20	61.07±0.49	69.73±0.24	62.83±1.59	66.39±0.65	66.76±0.67	64.76±0.07	69.86±0.18	31.45±0.54	59.23±2.32	60.56±0.76	71.02±0.77	69.84±0.94	70.12±0.36	71.77±0.42	72.11±0.86
	NMI	36.41±0.86	38.71±0.32	43.90±0.20	34.40±0.71	44.93±0.53	40.69±0.93	39.52±0.38	40.67±0.84	39.11±0.06	44.86±0.35	15.17±0.47	36.68±0.89	33.48±0.62	45.25±0.45	44.33±0.79	43.56±0.35	45.65±0.64	46.94±0.7
	ARI	37.78±1.24	40.17±0.43	45.50±0.30	34.32±0.70	45.31±0.41	34.18±1.73	41.07±0.96	38.73±0.55	37.54±0.12	45.64±0.30	14.32±0.78	33.37±0.53	32.61±0.81	46.29±1.13	45.68±1.80	44.85±0.69	47.21±0.73	48.08±1.37
	F1	62.20±1.32	63.62±0.24	64.30±0.20	58.23±0.31	64.45±0.27	59.54±2.17	61.12±0.70	58.22±0.68	59.64±0.05	64.83±0.21	30.20±0.71	52.67±0.64	57.08±0.92	64.80±1.01	62.71±2.06	65.01±0.39	62.53±0.40	63.69±1.38
AMAP	ACC	75.96±0.23	53.44±0.81	76.82±0.23	69.28±2.30	75.98±0.68	41.07±3.12	43.75±0.78	54.55±0.97	OOM	OOM	75.51±0.77	67.18±0.75	51.94±0.47	77.48±0.37	77.25±0.41	77.24±0.87	77.66±0.22	78.97±0.62
	NMI	65.25±0.45	44.85±0.83	66.23±1.21	58.36±2.76	65.38±0.61	30.28±3.94	37.32±0.28	48.56±0.71	OOM	OOM	64.05±0.15	63.63±1.07	41.27±0.75	67.67±0.88	67.44±0.48	67.12±0.92	67.48±0.57	69.55±0.52
	ARI	58.12±0.24	31.21±1.23	58.28±0.74	44.18±4.41	55.89±1.34	18.77±2.34	21.57±0.51	26.87±0.34	OOM	OOM	54.45±0.48	46.30±1.59	29.66±0.81	58.48±0.72	57.99±0.66	58.14±0.82	58.71±0.55	60.8±1.15
	F1	69.87±0.54	50.66±1.49	71.25±0.31	64.30±1.95	71.74±0.93	32.88±5.50	38.37±0.29	54.47±0.83	OOM	OOM	69.99±0.34	73.04±1.08	50.70±1.37	72.22±0.97	72.18±0.57	73.02±2.34	72.05±0.20	72.94±0.38
BAT	ACC	52.67±0.00	53.05±4.63	55.73±0.06	67.86±0.80	56.68±0.76	37.56±0.32	45.42±0.54	42.43±0.47	38.93±0.23	67.94±1.45	50.92±0.44	47.48±0.64	48.47±4.31	77.97±0.99	75.04±1.78	75.50±0.87	78.63±1.08	78.86±0.49
	NMI	21.43±0.35	25.74±5.71	48.77±0.51	49.09±0.54	36.04±1.54	29.33±0.70	31.70±0.42	17.84±0.98	23.11±0.56	47.23±0.74	27.55±0.62	24.36±1.54	22.20±4.59	52.91±0.68	50.23±2.43	50.58±0.90	54.64±1.17	54.70±0.81
	ARI	18.18±0.29	21.04±4.97	37.76±0.23	42.02±1.21	26.59±1.83	13.45±0.03	19.33±0.57	13.11±0.81	08.41±0.32	39.76±0.87	21.89±0.74	24.14±0.98	16.13±4.54	50.64±1.85	46.95±3.09	47.45±1.53	52.84±1.49	52.62±1.14
	F1	52.23±0.03	46.45±5.90	50.90±0.12	67.02±1.15	55.07±0.80	29.64±0.49	39.94±0.57	34.84±0.15	32.92±0.25	67.40±0.35	46.53±0.57	42.25±0.34	45.14±5.15	78.03±0.96	74.90±1.80	75.40±0.88	78.33±1.31	78.88±0.40
UAT	ACC	52.29±0.49	52.25±1.91	33.61±0.09	49.31±0.15	52.37±0.42	44.16±1.38	48.70±0.06	42.52±0.64	41.93±0.56	49.92±1.25	37.42±1.24	45.38±1.15	46.60±3.47	56.58±1.62	56.34±1.11	54.76±1.42	56.66±0.90	59.45±0.67
	NMI	21.33±0.44	21.61±1.26	26.49±0.41	25.44±0.31	23.64±0.66	21.53±0.94	25.10±0.01	17.86±0.22	16.64±0.41	24.09±0.53	17.33±0.54	24.49±0.57	14.92±2.84	28.07±0.71	28.15±1.92	25.23±0.96	27.29±0.87	29.92±0.47
	ARI	20.50±0.51	21.63±1.49	11.87±0.23	16.57±0.31	20.39±0.70	17.12±1.46	21.76±0.01	13.13±0.71	12.21±0.14	17.17±0.69	13.62±0.57	21.34±0.78	14.15±2.64	24.80±1.85	25.52±2.09	19.44±1.69	24.49±1.32	28.53±0.79
	F1	50.33±0.64	45.59±3.54	25.79±0.29	50.26±0.16	50.15±0.73	39.44±2.19	45.69±0.08	52.94±0.87	35.78±0.38	44.81±0.87	36.52±0.89	30.56±0.25	44.79±4.48	55.52±0.87	55.24±1.69	53.61±2.61	53.90±1.69	58.45±1.18
EAT	ACC	36.89±0.15	39.07±1.51	49.37±0.19	52.13±0.00	47.26±0.32	32.88±0.71	33.46±0.18	31.33±0.52	32.58±0.29	50.88±0.55	37.42±1.24	36.06±1.24	41.27±0.93	57.94±0.42	57.19±0.66	57.22±0.73	58.68±0.18	58.72±0.25
	NMI	05.57±0.06	08.83±2.54	32.90±0.41	22.48±1.21	23.74±0.90	11.72±1.08	13.22±0.33	07.63±0.85	07.04±0.56	22.01±1.23	11.44±1.41	21.46±1.80	14.68±1.52	33.91±0.49	33.85±0.87	33.47±0.34	34.80±0.19	34.63±0.22
	ARI	05.03±0.08	06.31±1.95	23.25±0.18	17.29±0.50	16.57±0.46	04.68±1.30	04.31±0.29	02.13±0.67	01.33±0.14	18.13±0.85	06.57±1.73	21.48±0.64	10.83±1.45	27.51±0.59	27.71±0.41	26.21±0.81	28.57±0.24	28.38±0.22
	F1	34.72±0.16	33.42±3.10	42.95±0.04	52.75±0.07	45.54±0.40	25.35±0.75	25.02±0.21	21.82±0.98	27.03±0.16	47.06±0.66	30.53±1.47	31.25±0.96	36.06±1.06	57.96±0.46	57.09±0.94	57.53±0.67	58.71±0.21	58.76±0.27

(ARGA [18]), and thirteen contrastive-based graph clustering models (including Adaptive Graph Embedding (AGE) [42], MVGRL [21], Multi-view Graph Clustering (MVGC) [25], DCRN [40], Graph Debiased Contrastive Learning (GDCL) [22], Automated Self-Supervised Learning (AutoSSL) [43], Augmentation-Free Graph Representation Learning (AFGRL) [44], Neighbor Contrastive Learnable Augmentation (NCLA) [28], Self-Cumulative Contrastive Graph Clustering (SC-CGC) [30], SCGC [23], Cluster-guided Contrastive deep Graph Clustering network (CCGC) [29], Motif-based Contrastive Graph Clustering approach with Clustering-Oriented Prompt (MCGC) [31], GraphLearner (marked as GL) [24]).

*3) **Parameter Setting**:* MT-AGC is trained for 400 epochs using the Adam optimizer. For the trade-off parameters β and γ are both 0.5 on all datasets. To ensure the reproducibility of the proposed method, the dataset statistics and dataset-specific hyper-parameter settings (learning rate, Top-K, embedding dimension, and T_{stage}) are summarized in Table II.

*4) **Metrics**:* To comprehensively evaluate the clustering performance of all methods, four common metrics are utilized, including Accuracy (ACC), Normalized Mutual Information (NMI), Average Rand Index (ARI), and macro F1-score (F1). For all indicators, higher values imply better clustering performance. Since K-means is used to obtain discrete clustering labels, to minimize the impact of randomness, each method is executed 10 times and the experimental results report the average values and corresponding standard deviations of all clustering metrics.

B. Performance Comparison

To demonstrate the superiority of our proposed MT-AGC algorithm and ensure the comprehensiveness and representativeness of the evaluation, we compare MT-AGC with 17 baselines. Please refer to the original papers for the detailed descriptions of these methods. For fair comparison, the original clustering results of all comparison methods are taken directly from the original papers or replicated using their published codes with recommended configurations.

Experimental results of all compared methods on the used benchmark datasets are presented in Table III. The red and blue values indicate optimal and sub-optimal results, respectively. From these results, some interesting conclusions can be summarized and analyzed as follows.

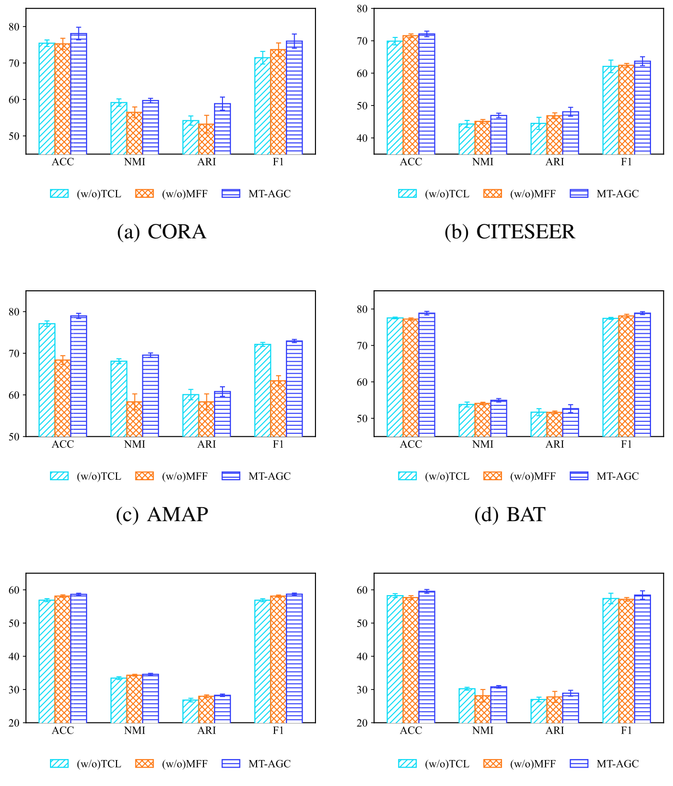
1) Across various application scenarios, including citation network (CORA and CITESEER), co-purchase network (AMAP), and air-traffic data (BAT, EAT, and UAT), the proposed MT-AGC invariably achieves the optimal clustering performance over all competitors in most cases.

2) The performance of MT-AGC consistently surpasses classical generative-based and adversarial-based graph clustering methods on all datasets. For instance, MT-AGC achieves average improvements of 17.50%, 9.63%, 15.98%, and 21.28% in ACC, NMI, ARI, and F1 over DFCN [17], respectively. This advantage mainly stems from the fact that generative and adversarial methods rely on prior assumptions (e.g., cluster number or initialization) and are sensitive to them, whereas MT-AGC learns representations through self-supervised contrastive signals inherent in the data, leading to better adaptability and optimization stability.

3) Among SCGC [23], CCGC [29], and GraphLearner [24], SCGC serves as a representative and strong contrastive baseline. MT-AGC still outperforms SCGC with consistent improvements of 1.92% in ACC, 1.92% in NMI, 2.95% in ARI, and 1.62% in F1 on average, indicating its superior clustering discriminability. This performance gain mainly stems from the joint effect of the proposed MFF and TCL. MFF constructs a feature-enhanced global graph and performs multi-order low-pass filtering with self-attentive fusion, providing rich and robust multi-scale representations, while triple contrastive learning further enforces semantic consistency at semantic, structural, and clustering levels to align instance features, relational geometry, and cluster semantics.

C. Ablation Study

To validate the effectiveness and necessity of three crucial components (Multi-order filter fusion module, Triple Con-

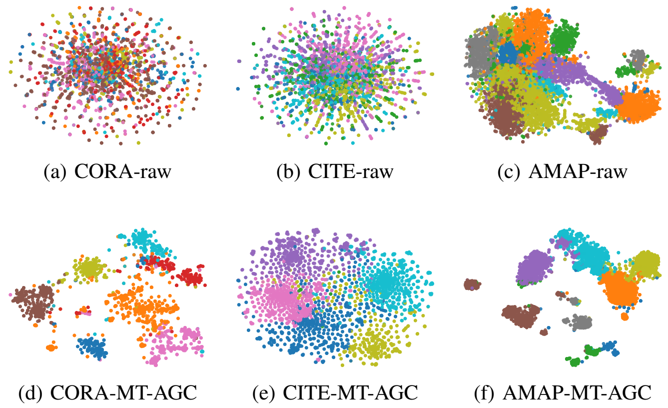


(a) CORA (b) CITESEER

(c) AMAP (d) BAT

(e) EAT (f) UAT

Fig. 2. The clustering results of three ablation models and MT-AGC on the six used datasets.



(a) CORA-raw (b) CITE-raw (c) AMAP-raw

(d) CORA-MT-AGC (e) CITE-MT-AGC (f) AMAP-MT-AGC

Fig. 3. The t-SNE visualization of raw attribute features and embedding representations learned by MT-AGC.

trastive Learning module) in our MT-AGC model, we compare the clustering results of the whole MT-AGC with its two ablation variants on six datasets. Specifically, we compare the complete MT-AGC model with the following two ablation variants:

- **(w/o) MFF:** This ablation variant is MT-AGC without MFF module and only utilizes t-order low-pass filter (i.e., $\mathbf{X}_f = \mathbf{F}_L^t$), which mainly verifies the effectiveness of feature-enhanced multi-order low-pass filtering information for comprehensive graph embedding representation.
- **(w/o) TCL:** This ablation variant is MT-AGC without TCL module and directly performs K-means on multi-order low-pass filter fusion representation $\mathbf{X}_f$, which aims to demonstrate the importance of representation alignment and structural consistency induced contrastive learning.

From these ablation results shown in Fig. 2, the following observations can be concluded.

*1) **Effect of the Multi-order filter fusion module**:* Across all datasets, MT-AGC consistently outperforms the variant without MFF, indicating that multi-order filtering fusion plays a crucial role in learning discriminative graph representations. Without multi-order fusion, relying solely on a single-order low-pass filter leads to over-smoothing and loss of structural details, which inevitably degrades clustering performance. The self-attentive fusion mechanism enables the model to automatically select informative propagation scales and effectively balances local consistency and global semantic aggregation.

*2) **Effect of the Triple Contrastive Learning module**:* Across all datasets, MT-AGC consistently outperforms the variant without TCL, indicating that the triple-level contrastive learning could effectively improve the discriminability of the graph embedding representation in a self-supervised way and enhance the stability and robustness of model. Specifically, the intra-class tightness of the embedding representation can be improved by forcing the distributions of the two augmented views to converge. Meanwhile, aligning the cross-view similarity with the prior adjacency information preserves the local topological semantics and enhances the inter-class separability.In addition, encouraging the semantic prediction distributions to remain consistent with the evolving representation geometry enables the model to progressively sharpen global semantic organization and suppress ambiguous assignments, thereby facilitating more coherent and stable partitioning of the embedding space.

D. Visualization Analysis

To intuitively demonstrate the superiority of MT-AGC in learning discriminative embedding representations, we employ the t-SNE technique [45] to visualize both the original attribute features and the embeddings learned by the proposed MT-AGC. As shown in Fig. 3, the visualization results indicate that MT-AGC can effectively uncover the discriminative structures in the data and enhance class separability among samples, thereby validating its advantage in discriminative representation learning.

V. Conclusion

This study proposed MT-AGC, a feature-enhanced framework for attribute graph clustering based on Multi-order Filter Fusion and Triple Contrastive Learning. In the representation learning stage, the MFF module strengthened the observed topology by constructing an attribute-similarity-induced global graph, which alleviates sparsity/noise issues. Then, multi-order low-pass filtering were integrated to mitigate the over-smoothing issue and obtain comprehensive embedding for contrastive augmentation. In the training stage, the TCL scheme integrated representation consistency, structural consistency, and pseudo-label guided semantic matching with reliable clustering feedback to improve semantic discriminability and optimization stability. Extensive experiments on six benchmark datasets demonstrated that the proposed MT-AGC method consistently outperforms competitive baselines in terms of ACC, NMI, ARI, and F1 metrics.

REFERENCES

[1] J. Guo, Y. Sun, J. Gao, Y. Hu, and B. Yin, "Low rank representation on product grassmann manifolds for multi-view subspace clustering," in *ICPR*. IEEE, 2021, pp. 907–914.

[2] Y. Tian, Y. Feng, X. Zhang, and C. Sun, "A fast clustering based evolutionary algorithm for super-large-scale sparse multi-objective optimization," *IEEE/CAA Journal of Automatica Sinica*, pp. 1048–1063, 2022.

[3] Y. Qin, H. Wu, J. Zhao, and G. Feng, "Enforced block diagonal subspace clustering with closed form solution," *Pattern Recognition*, p. 108791, 2022.

[4] J. Guo, Y. Sun, J. Gao, Y. Hu, and B. Yin, "Robust adaptive linear discriminant analysis with bidirectional reconstruction constraint," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 14, no. 6, pp. 1–20, 2020.

[5] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 4–24, 2021.

[6] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017.

[7] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *ICLR*, 2018.

[8] S. Guo, W. Chen, J. Gao, S. Xu, Y. Li, M. Liu, J. Guo, and Y. Wang, "Decouple then fusion: Flexible graph representation learning with cross-frequency diversity," in *IJCNN*. IEEE, 2025, pp. 1–7.

[9] T. Zhao, Y. Wang, S. Xu, T. Yang, J. Gao, and J. Guo, "Dual-level noise augmentation for graph clustering with triplet-wise contrastive learning," *Pattern Recognition*, p. 112463, 2026.

[10] S. Wang, J. Yang, J. Yao, Y. Bai, and W. Zhu, "An overview of advanced deep graph node clustering," *IEEE Transactions on Computational Social Systems*, pp. 1302–1314, 2023.

[11] Y. Ren, J. Pu, Z. Yang, J. Xu, G. Li, X. Pu, P. S. Yu, and L. He, "Deep clustering: A comprehensive survey," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 5858–5878, 2025.

[12] J. Zhao, Y. Sun, J. Guo, J. Gao, and B. Yin, "Robust graph convolutional clustering with adaptive graph learning," in *IJCNN*. IEEE, 2022, pp. 1–8.

[13] J. Guo, T. Yin, T. Zhao, J. Zhao, Y. Sun, J. Gao, and Y. Wang, "Improved attributed graph clustering with representation and structure augmentation," in *IJCNN*. IEEE, 2024, pp. 1–8.

[14] T. N. Kipf and M. Welling, "Variational graph auto-encoders," in *NIPS*, 2016.

[15] C. Wang, S. Pan, R. Hu, G. Long, J. Jiang, and C. Zhang, "Attributed graph clustering: A deep attentional embedding approach," in *IJCAI*, 2019, pp. 3670–3676.

[16] D. Bo, X. Wang, C. Shi, M. Zhu, E. Lu, and P. Cui, "Structural deep clustering network," in *WWW*, 2020, p. 1400–1410.

[17] W. Tu, S. Zhou, X. Liu, X. Guo, Z. Cai, E. Zhu, and J. Cheng, "Deep fusion clustering network," in *AAAI*, 2021, pp. 9978–9987.

[18] S. Pan, R. Hu, G. Long, J. Jiang, L. Yao, and C. Zhang, "Adversarially regularized graph autoencoder for graph embedding," in *IJCAI*, 2018, pp. 2609–2615.

[19] Z. Tao, H. Liu, J. Li, Z. Wang, and Y. Fu, "Adversarial graph embedding for ensemble clustering," in *IJCAI*, 2019, pp. 3562–3568.

[20] Y. Yang, F. Ju, Y. Sun, J. Gao, and B. Yin, "Adversarially regularized joint structured clustering network," *Information Sciences*, pp. 136–151, 2022.

[21] K. Hassani and A. H. Khasahmadi, "Contrastive multi-view representation learning on graphs," in *ICML*, 2020, pp. 4116–4126.

[22] H. Zhao, X. Yang, Z. Wang, E. Yang, and C. Deng, "Graph debiased contrastive learning with joint representation clustering." in *IJCAI*, 2021, pp. 3434–3440.

[23] Y. Liu, X. Yang, S. Zhou, X. Liu, S. Wang, K. Liang, W. Tu, and L. Li, "Simple contrastive graph clustering," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 13 789–13 800, 2023.

[24] X. Yang, E. Min, K. Liang, Y. Liu, S. Wang, S. Zhou, H. Wu, X. Liu, and E. Zhu, "Graphlearner: Graph node clustering with fully learnable augmentation," in *ACM MM*, 2024, pp. 5517–5526.

[25] W. Xia, S. Wang, M. Yang, Q. Gao, J. Han, and X. Gao, "Multi-view graph embedding clustering network: Joint self-supervision and block diagonal representation," *Neural Networks*, pp. 1–9, 2022.

[26] J. Guo, Y. Sun, X. Ma, J. Gao, Y. Hu, Y. Wang, and B. Yin, "Globality meets locality: An anchor graph collaborative learning framework for fast multiview subspace clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 6, pp. 10 213 – 10 227, 2025.

[27] X. Yang, C. Tan, Y. Liu, K. Liang, S. Wang, S. Zhou, J. Xia, S. Z. Li, X. Liu, and E. Zhu, "Convert: Contrastive graph clustering with reliable augmentation," in *ACM MM*, 2023, pp. 319–327.

[28] X. Shen, D. Sun, S. Pan, X. Zhou, and L. T. Yang, "Neighbor contrastive learning on learnable graph augmentation," in *AAAI*, 2023, pp. 9782–9791.

[29] X. Yang, Y. Liu, S. Zhou, S. Wang, W. Tu, Q. Zheng, X. Liu, L. Fang, and E. Zhu, "Cluster-guided contrastive graph clustering network," in *AAAI*, 2023, pp. 10 834–10 842.

[30] X. Yan, K. Deng, Q. Zou, Z. Tian, and H. Yu, "Self-cumulative contrastive graph clustering," *IEEE/CAA Journal of Automatica Sinica*, pp. 1194–1208, 2025.

[31] X. Wu, J. Hu, Y. Quan, Q. Miao, and P. G. Sun, "Motif-based contrastive graph clustering with clustering-oriented prompt," *Information Processing & Management*, p. 104208, 2025.

[32] W. Xia, Q. Wang, Q. Gao, M. Yang, and X. Gao, "Self-consistent contrastive attributed graph clustering with pseudo-label prompt," *IEEE Transactions on Multimedia*, pp. 6665–6677, 2022.

[33] J. Wang, Y. Sun, J. Wang, J. Gao, S. Wang, B. Yin, and J. Guo, "Comgrl: Comprehensive graph representation learning from local to global bridged by mixup," *IEEE Transactions on Computational Social Systems*, 2026.

[34] 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*, pp. 2475–2487, 2020.

[35] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014.

[36] H. Wang, J. Wang, J. Wang, M. Zhao, W. Zhang, F. Zhang, X. Xie, and M. Guo, "Graphgan: Graph representation learning with generative adversarial nets," in *AAAI*, 2018.

[37] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *ICML*, 2016, pp. 478–487.

[38] T. Zhao, Y. Wang, J. Wang, J. Wang, M. Cui, J. Gao, and J. Guo, "Hybrid-collaborative augmentation and contrastive sample adaptive-differential awareness for robust attributed graph clustering," in *NeurIPS*, 2025.

[39] Y. Wang, T. Zhao, M. Cui, J. Gao, L. Liang, and J. Guo, "Representation then augmentation: Wide graph clustering network with multi-order filter fusion and double-level contrastive learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 13, no. 2, pp. 421–435, 2026.

[40] Y. Liu, W. Tu, S. Zhou, X. Liu, L. Song, X. Yang, and E. Zhu, "Deep graph clustering via dual correlation reduction," in *AAAI*, 2022, pp. 7603–7611.

[41] W. Xia, Q. Wang, Q. Gao, M. Yang, and X. Gao, "Self-consistent contrastive attributed graph clustering with pseudo-label prompt," *IEEE Transactions on Multimedia*, pp. 6665–6677, 2023.

[42] G. Cui, J. Zhou, C. Yang, and Z. Liu, "Adaptive graph encoder for attributed graph embedding," in *SIGKDD*, 2020, p. 976–985.

[43] W. Jin, X. Liu, X. Zhao, Y. Ma, N. Shah, and J. Tang, "Automated self-supervised learning for graphs," in *ICLR*, 2022.

[44] N. Lee, J. Lee, and C. Park, "Augmentation-free self-supervised learning on graphs," in *AAAI*, 2022, pp. 7372–7380.

[45] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, pp. 2579–2605, 2008.