Dynamic Deep Graph Learning for Incomplete Multi-View Clustering with Masked Graph Reconstruction Loss

Zhenghao Zhang[1], Jun Xie[2], Xingchen Chen[3], Tao Yu[4], Hongzhu Yi[1], Kaixin Xu[5], Yuanxiang Wang[1], Tianyu Zong[1], Xinming Wang[4], Jiahuan Chen[6], Guoqing Chao[7*], Feng Chen[2], Zhepeng Wang[2], Jungang Xu[1*]

[1]School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing
[2]Lenovo Research, Beijing
[3]Faculty of Computing, Harbin Institute of Technology, Harbin
[4]Institute of Automation, Chinese Academy of Sciences, Beijing
[5]School of Software Technology, Zhejiang University, Ningbo
[6]School of Automation and Intelligent Sensing, Shanghai Jiao Tong University, Shanghai
[7]School of Computer Science and Technology, Harbin Institute of Technology, Weihai
zhangzhenghao25@mails.ucas.ac.cn, guoqingchao10@gmail.com, xujg@ucas.ac.cn

Abstract

The prevalence of real-world multi-view data makes incomplete multi-view clustering (IMVC) a crucial research. The rapid development of Graph Neural Networks (GNNs) has established them as one of the mainstream approaches for multi-view clustering. Despite significant progress in GNNs-based IMVC, some challenges remain: (1) Most methods rely on the K-Nearest Neighbors (KNN) algorithm to construct static graphs from raw data, which introduces noise and diminishes the robustness of the graph topology. (2) Existing methods typically utilize the Mean Squared Error (MSE) loss between the reconstructed graph and the sparse adjacency graph directly as the graph reconstruction loss, leading to substantial gradient noise during optimization. To address these issues, we propose a novel **D**ynamic Deep **G**raph Learning for **I**ncomplete **M**ulti-**V**iew **C**lustering with **M**asked Graph Reconstruction Loss (DGIMVCM). Firstly, we construct a missing-robust global graph from the raw data. A graph convolutional embedding layer is then designed to extract primary features and refined dynamic view-specific graph structures, leveraging the global graph for imputation of missing views. This process is complemented by graph structure contrastive learning, which identifies consistency among view-specific graph structures. Secondly, a graph self-attention encoder is introduced to extract high-level representations based on the imputed primary features and view-specific graphs, and is optimized with a masked graph reconstruction loss to mitigate gradient noise during optimization. Finally, a clustering module is constructed and optimized through a pseudo-label self-supervised training mechanism. Extensive experiments on multiple datasets validate the effectiveness and superiority of DGIMVCM.

Code — https://github.com/PaddiHunter/DGIMVCM

Introduction

Multi-view data refers to data that describes the same object from multiple perspectives (Fu et al. 2020). Multi-view learning aims to leverage information from multiple views to enhance generalization performance (Yu et al. 2024; Xu, Tao, and Xu 2013). Multi-view clustering (MVC) is a prominent unsupervised learning branch within multi-view learning, which groups data by exploiting the consistency and complementarity across views (Fang et al. 2023; Yang and Wang 2018; Zhou et al. 2024; Wu et al. 2024). Most existing multi-view clustering methods (Xu et al. 2021) operate under a data completeness assumption. However, due to various unavoidable factors, samples often have only partial views available (Tang et al. 2024; Yu et al. 2025). Consequently, incomplete multi-view clustering has garnered increasing attention.

Numerous studies (Ren et al. 2025, 2024) leverage Graph Neural Networks (GNNs) to address multi-view clustering problems. Most of these methods construct adjacency graphs from raw data using the K-nearest neighbors algorithm, and these graphs remain fixed during model training (Chao et al. 2025a; Chao, Jiang, and Chu 2024). However, such pre-defined graphs often suffer from noise inherent in the raw data, and their immutability during training can limit performance improvements. To mitigate this, some research endeavors first denoise the data using autoencoders and subsequently extract graph structures from the refined features (Huang et al. 2023; Wang et al. 2024). Nevertheless, these approaches are not well-suited to effectively handle data incompleteness. Furthermore, many methods (Chao et al. 2025b; Wang et al. 2024; Cheng et al. 2021) directly compute the MSE loss between the reconstructed graph and the adjacency graph as graph reconstruction loss. Due to the sparsity of adjacency graphs, this often leads to the loss optimization gradient direction containing substantial noise, which impedes further performance enhancement.

To address these limitations, we propose the Dynamic Deep Graph Learning for Incomplete Multi-View Clustering with Masked Graph Reconstruction Loss. Our approach first utilizes a Graph Convolutional Network(GCN)-based embedding layer with a global graph to impute features for

*Corresponding authors.

missing views. Subsequently, view-specific graphs are dynamically extracted from these features. The structures of these view-specific graphs are then refined by global graph, and consistency among them is learned through graph structure contrastive learning. We then design a Graph Attention Network (GAT) encoder, enabling the model to dynamically learn edge weights, and optimize it using the masked graph reconstruction loss to reduce gradient noise during optimization. Finally, a clustering module is integrated, leveraging pseudo-labels to guide the training process.

The main contributions of this paper are summarized as follows:

1. We propose a novel incomplete multi-view clustering method that dynamically learns graph structures and edge weights. Extensive experimental results on four widely used datasets validate the superiority of it.
2. We introduce an innovative embedding layer and a graph structure contrastive learning method, which are designed to enable the learning of inter-view consistent graph structures and handle data incompleteness.
3. We present a generic masked graph reconstruction loss that effectively mitigates gradient noise during the optimization process. This is demonstrated through both theoretical analysis and extensive experimental evaluation.

Related Works

Deep Incomplete Multi-view Clustering

Deep incomplete multi-view clustering (DIMVC) methods are broadly categorized into imputation-based and imputation-free approaches. Imputation-based methods typically initiate by recovering missing views, followed by clustering on the reconstructed complete multi-view dataset. For example, AGDIMC (Pu et al. 2024) imputes missing features adaptively by leveraging cross-view soft cluster assignments and global cluster centroids. Similarly, Tang and Liu (2022) propose a bi-level optimization framework that dynamically imputes missing views from learned semantic neighbors. Conversely, imputation-free methods directly perform clustering based on available views. For instance, Xu et al. (2024) utilizes multiple view-specific encoders to extract information from each view and employs the Product-of-Experts (PoE) approach to obtain a common representation.

Graph Neural Networks for Multi-view Clustering

Graph Neural Networks (GNNs) are widely applied in multi-view clustering due to their ability to learn both node attributes and graph structures (Wen et al. 2021a; Shao et al. 2022; Wen et al. 2024). For instance, SERIES (Wang and Feng 2024) leverages GNNs through deep graph autoencoders to extract latent representations from multi-view data, which are then fused to form a consistent structural graph for clustering. SURER (Wang et al. 2024) enhances multi-view clustering by adaptively refining view-specific graphs and unifying them into a heterogeneous graph, subsequently learning a consensus representation via graph neural network. SGDMC (Huang et al. 2023) performs multi-view clustering using self-supervised graph attention networks, where attention allocation considers both node attributes and pseudo-labels. GHICMC (Chao et al. 2025a) performs multi-view clustering by learning view-specific and consensus representations via GCNs, imputing missing data through hierarchical global graph propagation. ICMVC (Chao, Jiang, and Chu 2024) leverages multi-view consistency transfer with GCN for missing data, fuses view-specific representations via instance-level attention, and applies contrastive learning and high-confidence guiding.

The Proposed Method

In this section, we introduce the proposed method consisting of four modules. Figure 1 illustrates the overall architecture of the method.

Notations Let $\mathcal{X} = \{\mathcal{X}^1, ..., \mathcal{X}^v, ..., \mathcal{X}^V\}$ denotes a complete multi-view dataset with V views. $\mathcal{X}^v \in \mathbb{R}^{N \times d^v}$ represents the dataset for the v-th view, where N is the number of samples and d^v is the feature dimension of the v-th view. To construct an incomplete multi-view dataset X, a missing indicator matrix $M \in \{0,1\}^{N \times V}$ is defined. Specifically, $M_{iv} = 1$ indicates the presence of the v-th view for the i-th sample, while $M_{iv} = 0$ denotes the absence of it. Each sample has at least one available view. The incomplete dataset is then defined as $X = \{X^1, ..., X^v, ...X^V\}$, where $X^v = \mathcal{M}^v \odot \mathcal{X}^v$. Here, $\mathcal{M} \in \{0,1\}^{N \times d^v}$ is a matrix obtained by broadcasting the v-th column of M, and $\odot$ denotes the Hadamard product. The objective of the algorithm is to cluster X into K categories.

Global Graph Fusion with Missing Robustness

To exploit consistency across views and leverage complementary information from other views to alleviate missing data issues, we construct a global graph for feature imputation in the embedding layer and optimization of the encoder.

The global graph is aggregated from the graphs of individual views. Specifically, we first compute the similarity matrix $S^v \in (0,1]^{N \times N}$ for each view using the radial basis function: $S^v_{ij} = \exp\left(-\frac{\|x^v_i - x^v_j\|_2^2}{t}\right)$, where S^v_{ij} quantifies the similarity between samples x^v_i and x^v_j in the v-th view, and t is a scale parameter. A challenge arises because missing samples are represented by zero vectors, leading to abnormally high similarities between them, which significantly impact the edges of the global graph during aggregation. Therefore, prior to aggregation, each S^v is pre-processed by pruning edges associated with missing samples. The global graph $\overline{A} \in \{0,1\}^{N \times N}$ is then computed as follows:

$$\overline{A} = \text{TopK}\left(\sum_{v=1}^{V} f_P(S^v)\right), \tag{1}$$

where the $f_P(\cdot)$ function re-calibrates the similarity matrix by zeroing out rows and columns associated with missing samples. Subsequently, the function TopK$(\cdot)$ constructs an adjacency matrix by setting the K largest elements in each row to 1 and the remainder to 0. This approach effectively reduces the influence of missing samples on the global graph.

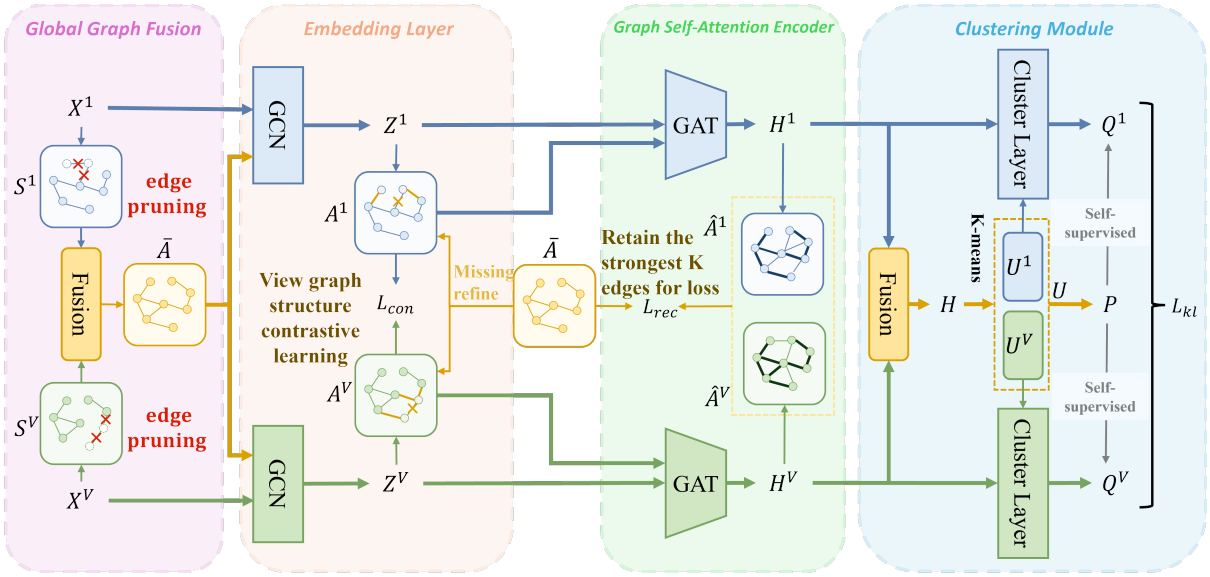

Figure 1: The overview of the DGIMVCM framework. The framework comprises four components: global graph fusion, embedding layer, graph self-attention encoder, and the clustering module. Initially, a global graph $\overline{A}$ is constructed by fusing view-specific similarity matrices with missing sample edge pruning. Subsequently, a GCN-based embedding layer utilizes $\overline{A}$ to extract imputed primary features $\{Z^v\}_{v=1}^V$ and view-specific graphs $\{A^v\}_{v=1}^V$ with missing structure refinement by the global graph, which is optimized via view-specific graph structure consistency contrastive learning. Next, a graph self-attention encoder is employed to extract high-level features $\{H^v\}_{v=1}^V$, optimized by a masked graph reconstruction loss that focuses only on the K strongest edges. Finally, a clustering module performs self-supervised training using pseudo-labels and obtain the final clustering results.

Embedding Layer for Dynamic View-specific Graph Construction

Current GNN-based methods for IMVC are often limited by static graph structure with raw data noise. To overcome these challenges, we introduces a Graph Convolutional Network (GCN)-based embedding layer to dynamically update the refined view-specific graph structure and imputes the feature of missing samples. Taking view v as an example, the embedding layer is designed as follows.

Architecture of the Embedding Layer We implement the embedding layer using GCN with the global graph as input. This approach allows the primary features of missing samples to integrate information from their adjacent complete samples. The computation for the embedding layer of view v is given by:

$$Z^v = \overline{A}' X^v W_e^v + b_e^v, \tag{2}$$

where Z^v represents the primary features for the v-th view, $\overline{A}' = \overline{D}^{-\frac{1}{2}} \overline{A} \overline{D}^{-\frac{1}{2}}$ is the normalized global graph adjacency matrix, and $\overline{D}$ is a diagonal matrix with $\overline{D}_{ii} = \sum_j \overline{A}_{ij}$. Furthermore, W_e^v and b_e^v are the trainable parameters in the embedding layer.

Dynamic View-specific Graph Construction Similar to the global graph construction, we first compute a similarity matrix $\hat{S}^v \in (0,1]^{N \times N}$ based on Z^v by the radial basis function as previously described. At this stage, the similarity matrix is computed using the imputed features and integrates information from the global graph, thereby possessing enhanced representational power. Subsequently, we construct the view-specific adjacency matrix $A^v \in \{0,1\}^{N \times N}$ using the TopK$(\cdot)$ function:

$$A^v = \text{TopK}(\hat{S}^v), \tag{3}$$

where the function TopK$(\cdot)$ is defined consistently with its usage in Eq. (1). To mitigate the impact of missing data, following related works (Chao et al. 2025b), $\mathbf{A}^v$ undergoes a refinement process: inaccurate edges corresponding to missing samples are replaced with their counterparts from $\overline{\mathbf{A}}$.

Contrastive Learning for View-specific Graphs Direct application of contrastive learning to features can force multi-view features into a single representation space, thereby reducing the diversity of feature representation. To circumvent this issue, we instead employ view-graph structure contrastive learning to simultaneously uncover consistency in graph structures across different views and enhance the distinctiveness of graph structures among samples. Specifically, graph structures of the same sample across different views are considered positive pairs, whereas graph structures of different samples are considered as negative pairs. The contrastive loss for graph structures between view v and view w is then calculated as:

$$L_{con}^{(v,w)} = \frac{1}{N} \sum_{i=1}^{N} -\log \frac{\exp(d(\hat{s}_i^v, \hat{s}_i^w)/\tau)}{\sum_{u=v,w} \sum_{j \neq i} \exp(d(\hat{s}_i^v, \hat{s}_j^u)/\tau)}, \tag{4}$$

where $d(\hat{s}_i^v, \hat{s}_i^w)$ denotes the cosine similarity between $\hat{s}_i^v$ and $\hat{s}_i^w$, $\hat{s}_i^v$ represents the i-th row of the similarity matrix $\hat{S}^v$, and τ is the temperature parameter. The contrastive loss for graph structures across different views is given by:

$$L_{con} = \sum_{v=1}^{V} \sum_{w \neq v} L_{con}^{(v,w)}. \tag{5}$$

Graph Self-Attention Encoder for Representation Learning

Architecture of the Encoder To overcome the limitations of fixed edge weights inherent in GCN, we employ Graph Attention Networks (GATs) to construct the encoder. GATs can automatically learn self-attention weights, which serve as dynamic edge weights. Specifically, for view v, the encoder takes Z^v and A^v as input, mapping them to refined high-level feature representations $H^v \in \mathbb{R}^{N \times \hat{d}^v}$, where $\hat{d}^v$ is the dimension of high-level feature for view v.

Considering the l-th layer, the core of the encoder involves computing the self-attention weight $e_{ij,(l)}^v$ between sample i and sample j. This is achieved using a Dot-Product Attention mechanism (Vaswani et al. 2017):

$$e_{ij,(l)}^v = \left(H_{i,(l-1)}^v W_{Q,(l)}^v\right) \left(H_{j,(l-1)}^v W_{K,(l)}^v\right)^{\mathrm{T}}, \tag{6}$$

where $H_{i,(l-1)}^v$ denotes the feature representation of sample i in view v at the $(l-1)$-th layer of the encoder, $W_{Q,(l)}^v$ and $W_{K,(l)}^v$ are the learnable projection matrices for computing the Query and Key respectively at the l-th layer. This enables the model to adaptively learn attention weights. Notably, the features for the 0-th layer are initialized as $H_{(0)}^v = Z^v$.

Next, to integrate the original graph structure information, we constrain the computed attention weight matrix using the view-specific graph and normalize it row-wise using softmax function, yielding the edge weight matrix $\tilde{A}_{(l)}^v$:

$$\tilde{A}_{ij,(l)}^v = \begin{cases} \frac{\exp(e_{ij,(l)}^v)}{\sum_{k \in \{k | A_{ik}^v = 1\}} \exp(e_{ik,(l)}^v)}, & \text{if } A_{ij}^v = 1 \\ 0, & \text{if } A_{ij}^v = 0, \end{cases} \tag{7}$$

where $A_{ij}^v = 1$ indicates the existence of an edge between samples i and j in the original view-specific graph A^v. This operation ensures that self-attention edge weights exist only between originally connected nodes.

Utilizing this learnable edge weight matrix $\tilde{A}_{(l)}^v$, the features $H_{(l)}^v$ at the l-th layer are computed as follows:

$$H_{(l)}^v = \sigma((\tilde{A}_{(l)}^v H_{(l-1)}^v W_{V,(l)}^v) W_{(l)}^v + b_{(l)}^v) + H_{(l-1)}^v, \tag{8}$$

where $W_{V,(l)}^v$ is a projection matrix for computing the Value, $b_{(l)}^v$ is the bias vector, $\sigma(\cdot)$ denotes the activation function, and $W_{(l)}^v$ is a projection matrix used for further feature transformation. It is noteworthy that residual connections are employed to mitigate degradation issues in networks.

Masked Graph Reconstruction Loss For simplicity, we compute a similarity matrix serving as the reconstructed graph $\hat{A}^v$ using the radial basis function:

$$\hat{A}_{ij}^v = e^{-\frac{||h_i^v - h_j^v||_2^2}{t}}, \tag{9}$$

where h_i^v represents the high-level feature of the i-th sample in view v. To guide the features from different views towards learning a unified graph structure, the global graph $\overline{A}$ serves as the reconstruction target. Unlike conventional graph reconstruction losses, the masked graph reconstruction loss focuses on the strongest k edges for each sample within the reconstructed graph, and is computed as:

$$L_{rec} = \frac{1}{V} \sum_{v=1}^{V} \frac{1}{N} ||M^v \odot \hat{A}^v - \overline{A}||_2^2, \tag{10}$$

where $M^v \in \{0,1\}^{N \times N}$ is the graph mask matrix. Let $\hat{A}_{i\cdot}^v$ denote the i-th row of $\hat{A}^v$, then $M_{ij}^v = 1$ if $\hat{A}_{ij}^v$ is among the top-k largest elements in $\hat{A}_{i\cdot}^v$, otherwise $M_{ij}^v = 0$.

Gradient Analysis of Masked Graph Reconstruction Loss This subsection highlights the advantages of our masked graph reconstruction loss over traditional approaches by analyzing their gradients. Detailed derivations are provided in Appendix A. Here, we present only the results.

Without loss of generality, consider the loss associated with i-th sample feature h_i^v from view v. We first analyze the gradient of the traditional graph reconstruction loss $L_{rec\_t,i}^v = \sum_{j=1}^{N} (\hat{A}_{ij}^v - \overline{A}_{ij})^2$, with respect to h_i^v:

$$\frac{\partial L_{rec\_t,i}^v}{\partial h_i^v} = \frac{4}{t}\Big(\sum_{j \in E_i} (\hat{A}_{ij}^v - 1)\hat{A}_{ij}^v (h_j^v - h_i^v) + \sum_{j \notin E_i} (\hat{A}_{ij}^v)^2 (h_j^v - h_i^v)\Big), \tag{11}$$

where $E_i = \{j \mid \overline{A}_{ij} = 1\}$ denotes the set of indices for samples adjacent to sample i in the global graph. Two types of sample loss gradient are considered. For $j \in E_i$, the term $(\hat{A}_{ij}^v - 1)\hat{A}_{ij}^v$ is negative. Consequently, the gradient direction is $h_i^v - h_j^v$. As the model updates in the direction opposite to the loss gradient, this implies that h_i^v is updated towards h_j^v, which can be interpreted as an attractive force. Conversely, if $j \notin E_i$, h_i^v is updated away from h_j^v, acting as a repulsive force. Attraction forces are considered more critical than repulsive forces, as they facilitate the formation of clusters. However, because k is generally small, leading to $|E_i| \ll N$, the second term in Eq. (11) encompasses a large number of components. Moreover, in the early stages of training, $\hat{A}_{ij}^v$ values are not negligible, and the repulsive forces exhibit diverse directions. This introduces significant gradient noise during optimization. Additionally, as the global graph is sparse, non-adjacent samples may still belong to the same cluster. The repulsive forces between such samples can excessively reduce their similarity.

For clarity, we refer to edges present in the global graph as "informative edges". Edges that are absent in the global

graph but present in the reconstructed graph are termed "hard edges", as they indicate samples that are difficult for the encoder to distinguish. The gradient of our masked graph reconstruction loss $L^v_{rec\_m,i} = \sum_{j=1}^{N}(M^v_{ij}\hat{A}^v_{ij} - \overline{A}_{ij})^2$, with respect to h^v_i is given by:

$$\frac{\partial L^v_{rec\_m,i}}{\partial h^v_i} = \frac{4}{t}\Big(\sum_{j:j\in E_i, M^v_{ij}=1}(\hat{A^v_{ij}} - 1)\hat{A}^v_{ij}(h^v_j - h^v_i) + \sum_{j:j\notin E_i, M^v_{ij}=1}\hat{A^v_{ij}}^2(h^v_j - h^v_i)\Big). \tag{12}$$

Here, the first term represents informative edges, while the second term addresses hard edges. During optimization, the masking mechanism selects and retains the k strongest edges in the reconstructed graph. The small value of k significantly reduces gradient noise, predominantly manifesting as attractive forces, thereby enabling h^v_i to be optimized towards well-defined directions. For the selected edges, if they are also present in the global graph, the loss reinforces these connections. If they are absent, the loss weakens these connections, allowing other more relevant edges to be selected while preventing excessive weakening.

Algorithm 1: Optimization algorithm for DGIMVCM

Input: Incompleted multi-view data with N samples and V views $\{X^v\}_{v=1}^V$, number of clusters K, maximum iterations $epochs$.

Output: Clustering results Y.

```
Compute global graph Ā by Eq. (1)
for epoch=1 to epochs do
    Compute primary features {Z^v}_{v=1}^V by Eq. (2)
    Compute view-specific graphs {A^v}_{v=1}^V by Eq. (3)
    Compute the loss L_con by Eq. (5)
    Compute the high-level features {H^v}_{v=1}^V by Eq. (8)
    Compute the {Â^v}_{v=1}^V by Eq. (9)
    Compute the loss L_rec by Eq. (10)
    Compute the fused features H by Eq. (13)
    Perform K-means on H to obtain U
    Compute P by Eq. (15)
    Compute {Q^v}_{v=1}^V by Eq. (16)
    Compute the loss L_kl by Eq. (17)
    Compute the overall loss L by Eq. (19)
    Update through gradient descent to minimize L
end for
Obtain the final clustering result Y by Eq. (18)
```

Self-supervised Clustering Module

Pseudo-label Acquisition To provide global clustering supervision, we first compute global pseudo-labels P. Specifically, High-level features $\{H^v\}_{v=1}^V$ from all views are fused to construct a global feature representation $H \in \mathbb{R}^{N\times\sum_{v=1}^V \hat{d}_v}$:

$$H = [H^1, \ldots, H^v, \ldots, H^V], \tag{13}$$

where $[\cdot, \ldots, \cdot]$ denotes the horizontal concatenation of matrices. Subsequently, during training, the K-means algorithm is applied to H to iteratively update the global cluster centers $U = [U^1, \ldots, U^V] \in \mathbb{R}^{K\times\sum_{v=1}^V \hat{d}_v}$. Here, $U^v \in \mathbb{R}^{K\times\hat{d}^v}$ represents the cluster centers for view v. Consequently, the soft labels Q are computed as follows:

$$q_{ij} = \frac{(1 + ||h_i - U_j||_2^2)^{-1}}{\sum_{k=1}^K (1 + ||h_i - U_k||_2^2)^{-1}}, \tag{14}$$

where q_{ij} denotes the probability of the i-th sample being assigned to the j-th cluster, h_i represents the global feature of the i-th sample, and U_j is the j-th global cluster center. Finally, by sharpening the soft labels, the pseudo-labels P are obtained via:

$$p_{ij} = \frac{(\frac{q_{ij}}{\sum_j q_{ij}})^2}{\sum_j (\frac{q_{ij}}{\sum_j q_{ij}})^2}. \tag{15}$$

Clustering Layer The clustering layer computes soft labels based on the cluster centers $\{U^v\}_{v=1}^V$ using Student's t-distribution, as presented below:

$$q^v_{ij} = \frac{(1 + ||h_i - U^v_j||_2^2)^{-1}}{\sum_{k=1}^K (1 + ||h_i - U^v_k||_2^2)^{-1}}, \tag{16}$$

where q^v_{ij} denotes the probability that the i-th sample in view v is assigned to the j-th cluster, h^v_i denotes the high level feature of the i-th sample in view v, and U^v_j denotes the j-th cluster center for view v, which implicitly updates with the global cluster centers U. The global pseudo-labels P serve as the supervisory signal to optimize the soft distributions of each view. This is achieved by introducing kl divergence loss:

$$L_{kl} = \sum_{v=1}^V D_{KL}(P||Q^v) = \sum_{v=1}^V \sum_{i=1}^N \sum_{j=1}^K p_{ij} \log \frac{p_{ij}}{q^v_{ij}}. \tag{17}$$

Finally, by combining the results from all view-specific soft labels, the ultimate clustering assignment $Y = \{y_i\}_{i=1}^N$ is obtained:

$$y_i = \arg\max_k \sum_{v=1}^V q^v_{ik}, \tag{18}$$

where y_i represents the assigned cluster label for the i-th sample.

The Overall Loss Function

In summary, the overall loss function is defined as:

$$L = L_{rec} + \alpha L_{con} + \beta L_{kl}, \tag{19}$$

where, α and β represent hyperparameters that balance the contributions of the three loss functions. The complete algorithn procedure is detailed in Algorithm 1.

Experiments

Experimental Settings

To construct incomplete datasets, the missing rate is defined as $\delta = (N - n)/N$, which represents the proportion of samples with missing views in a multi-view dataset. Here, N denotes the total number of samples in the dataset, and n represents the number of samples with complete views. For incomplete samples, a random number of views are removed.

δ	Methods	HW			Scene-15			Landuse-21			100Leaves		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
0	DIMVC	0.446	0.533	0.381	0.350	0.309	0.179	0.243	0.301	0.109	0.825	0.922	0.753
	DSIMVC	0.759	0.756	0.661	0.281	0.299	0.146	0.177	0.173	0.049	0.401	0.725	0.294
	GIGA	0.807	0.853	0.756	0.221	0.263	0.041	0.131	0.257	0.017	0.742	0.877	0.483
	GIMVC	0.935	0.886	0.874	0.426	0.465	0.279	0.258	0.337	0.114	0.857	0.952	0.819
	CDIMC-net	0.861	0.890	0.827	0.347	0.421	0.198	0.184	0.236	0.054	0.799	0.938	0.751
	MRL_CAL	0.478	0.526	0.337	0.194	0.168	0.069	0.163	0.169	0.045	0.224	0.587	0.126
	DCP	0.828	0.852	0.771	0.401	0.436	0.240	0.260	0.311	0.121	0.606	0.843	0.496
	GHICMC	0.854	0.860	0.794	0.434	0.438	0.273	0.266	0.313	0.128	0.940	0.969	0.914
	Ours	**0.979**	**0.953**	**0.953**	**0.503**	**0.508**	**0.330**	**0.319**	**0.393**	**0.171**	**0.956**	**0.984**	**0.940**
0.5	DIMVC	0.322	0.255	0.151	0.310	0.261	0.143	0.226	0.278	0.099	0.579	0.731	0.380
	DSIMVC	0.729	0.687	0.586	0.260	0.267	0.125	0.172	0.169	0.048	0.295	0.616	0.171
	GIGA	0.764	0.730	0.594	0.146	0.127	0.008	0.182	0.279	0.025	0.418	0.649	0.055
	GIMVC	0.911	0.838	0.825	0.385	0.373	0.218	0.228	0.273	0.085	0.688	0.842	0.555
	CDIMC-net	0.858	0.861	0.792	0.217	0.268	0.067	0.122	0.161	0.020	0.330	0.643	0.207
	MRL_CAL	0.358	0.370	0.191	0.189	0.150	0.065	0.162	0.168	0.044	0.145	0.434	0.052
	DCP	0.628	0.671	0.463	0.397	0.408	0.232	0.256	0.292	0.119	0.449	0.689	0.209
	GHICMC	0.844	0.844	0.784	0.413	0.403	0.247	0.251	0.285	0.112	0.707	0.826	0.565
	Ours	**0.939**	**0.879**	**0.870**	**0.452**	**0.430**	**0.273**	**0.287**	**0.328**	**0.140**	**0.812**	**0.891**	**0.703**

Table 1. Experimental results on the four datasets. The best results in each column are shown in bold and the second best results are underlined. $\delta = 0$ indicates complete, while $\delta = 0.5$ indicates incomplete.

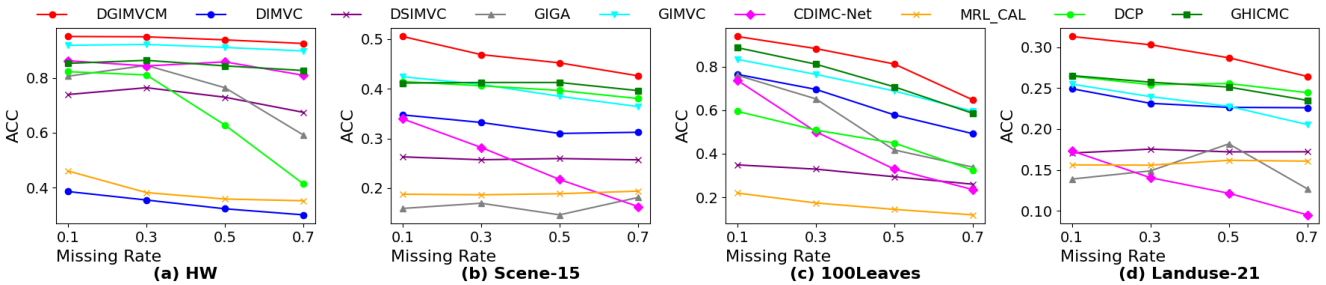

Figure 2: Accuracy on four datasets with different missing rates.

Datasets and Metrics

Experiments are conducted on four widely used multi-view datasets. Specifically, **Scene-15** (Fei-Fei and Perona 2005; Lazebnik, Schmid, and Ponce 2006) comprises 4485 scene images categorized into 15 classes, with each sample represented by three distinct views. **HandWritten** (Li et al. 2015) contains 2000 samples across ten numeric categories, each characterized by six views. **Landuse-21** (Yang and Newsam 2010) consists of 2100 satellite images distributed among 21 categories, with each category containing 100 images, and each image represented by three views. Finally, the **100leaves** dataset (Mallah et al. 2013) includes 1600 image samples derived from 100 plant species, with each sample possessing three different views.

To evaluate the effectiveness of the clustering approach, three widely recognized metrics are employed: clustering accuracy (ACC), normalized mutual information (NMI), and adjusted random index (ARI). For each of these metrics, a higher value signifies superior clustering performance.

Compared Methods

To demonstrate the superiority of our proposed method, we conduct a comparative analysis against eight state-of-the-art incomplete multi-view clustering methods. These methods are enumerated as follows:

- **DIMVC** (Xu et al. 2022): This method proposes an imputation-free and fusion-free deep framework specifically designed for incomplete multi-view clustering.
- **DSIMVC** (Tang and Liu 2022): This method introduces a bi-level optimization framework that addresses missing views by leveraging learned neighbor semantics.
- **GIGA** (Yang et al. 2024): This method adaptively estimates the factual weight of each available view to mitigate the adverse effects of missing data.
- **GIMVC** (Bai et al. 2024): This is an imputation-free incomplete multi-view clustering method that incorporates a graph-guided mechanism.
- **CDIMC-net** (Wen et al. 2021b): This method integrates

encoders with a graph embedding strategy to capture both high-level features and local structural information.

- **MRL_CAL** (Wang, Zhang, and Ma 2024): This method facilitates data recovery, consistent representation learning, and clustering through the joint learning of features across distinct subspaces.
- **DCP** (Lin et al. 2022): This method performs multi-view clustering by jointly maximizing inter-view mutual information and minimizing conditional entropy.
- **GHICMC** (Chao et al. 2025a): This method integrates representation learning, global graph propagation and contrastive clustering.

Experimental Results and Analysis

Table 1 presents the average clustering results obtained over five runs for each method. It can be observed that our method consistently outperforms baseline methods on all datasets, under both complete and incomplete data conditions.

To further validate the effectiveness of our method, experiments are conducted with missing rates ranging from 0.1 to 0.7, with an interval of 0.2. Figure 2 illustrates the performance of all methods across four datasets. It is evident that our proposed model achieves superior performance consistently across all datasets and missing rate settings.

Ablation Study

To validate the effectiveness of each component in our proposed method, we conducted ablation studies by progressively removing: (A) the masked graph reconstruction loss L_{rec}, (B) the embedding layer with the contrastive loss L_{con}, and (C) the self-supervised clustering module with L_{kl}. Table 2 presents the experimental results on the 100leaves dataset with a missing rate of 0.5. It is noteworthy that when the embedding layer is removed, view-specific graphs are constructed from the raw data X. The performance of the model consistently degrades to varying degrees upon the removal of each component. Notably, the absence of the embedding layer leads to a substantial decrease in performance. In contrast, the self-supervised clustering module demonstrates the least impact on the overall performance.

Components			100Leaves		
A	B	C	ACC	NMI	ARI
✓	✓	✓	**0.812**	**0.891**	**0.703**
	✓	✓	0.704	0.824	0.561
✓		✓	0.529	0.714	0.290
✓	✓		0.806	0.890	0.698

Table 2. Ablation studies on 100Leaves when $\delta = 0.5$.

Loss Function Comparison

To validate the effectiveness of the masked graph reconstruction loss, we conduct experiments comparing its performance against the traditional graph reconstruction loss. Table 3 summarizes the model performance on the Scene-15 dataset, evaluated under missing data rates ranging from 0.1 to 0.7, with an interval of 0.2. The results clearly demonstrate that the masked graph reconstruction loss consistently outperforms the traditional graph reconstruction loss across most experimental conditions.

Parameter Sensitivity Analysis

In this subsection, we analyze the model's sensitivity to hyperparameters α and β. Both hyperparameters are varied within the range $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$. Figure 3 presents the experimental results obtained on the Landuse-21 dataset with a missing rate of 0.5. Our model demonstrates robustness to these hyperparameters, consistently achieving competitive performance across their specified reasonable ranges.

δ	Type of the loss	ACC	NMI	ARI
0.1	masked	**0.505**	**0.497**	**0.326**
	traditional	0.479	0.486	0.305
0.3	masked	0.469	**0.464**	**0.298**
	traditional	**0.470**	0.458	0.286
0.5	masked	**0.452**	**0.430**	**0.273**
	traditional	0.447	0.425	0.262
0.7	masked	**0.426**	**0.397**	**0.244**
	traditional	0.403	0.390	0.227

Table 3. Comparison of graph reconstruction loss functions on Scene-15 for varying δ.

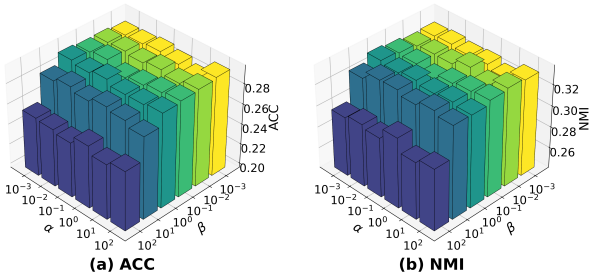

Figure 3: Parameter sensitivity analysis for α and β on Landuse-21 with missing rate of 0.5.

Conclusion

This paper proposes DGIMVCM, a novel incomplete multi-view clustering framework. The framework incorporates a GCN-based embedding layer, designed to dynamically extract view-specific graphs and effectively address the missing data problem. Furthermore, it employs a GAT-based encoder, which adaptively learns edge weights. To optimize this encoder and mitigate gradient noise during optimization, a masked graph reconstruction loss is utilized. Extensive experiments consistently demonstrate the superiority of DGIMVCM compared to state-of-the-art methods.

References

Bai, S.; Zheng, Q.; Ren, X.; and Zhu, J. 2024. Graph-guided imputation-free incomplete multi-view clustering. *Expert Systems with Applications*, 258: 125165.

Chao, G.; Jiang, Y.; and Chu, D. 2024. Incomplete contrastive multi-view clustering with high-confidence guiding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 11221–11229.

Chao, G.; Xu, K.; Xie, X.; and Chen, Y. 2025a. Global Graph Propagation with Hierarchical Information Transfer for Incomplete Contrastive Multi-view Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 15713–15721.

Chao, G.; Zhang, Z.; Meng, L.; Wen, J.; and Chu, D. 2025b. Federated Incomplete Multi-view Clustering with Globally Fused Graph Guidance. In *Forty-second International Conference on Machine Learning*.

Cheng, J.; Wang, Q.; Tao, Z.; Xie, D.; and Gao, Q. 2021. Multi-view attribute graph convolution networks for clustering. In *Proceedings of the Twenty-ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2973–2979.

Fang, U.; Li, M.; Li, J.; Gao, L.; Jia, T.; and Zhang, Y. 2023. A comprehensive survey on multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(12): 12350–12368.

Fei-Fei, L.; and Perona, P. 2005. A bayesian hierarchical model for learning natural scene categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, 524–531. IEEE.

Fu, L.; Lin, P.; Vasilakos, A. V.; and Wang, S. 2020. An overview of recent multi-view clustering. *Neurocomputing*, 402: 148–161.

Huang, Z.; Ren, Y.; Pu, X.; Huang, S.; Xu, Z.; and He, L. 2023. Self-supervised graph attention networks for deep weighted multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 7936–7943.

Lazebnik, S.; Schmid, C.; and Ponce, J. 2006. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, 2169–2178. IEEE.

Li, Y.; Nie, F.; Huang, H.; and Huang, J. 2015. Large-scale multi-view spectral clustering via bipartite graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Lin, Y.; Gou, Y.; Liu, X.; Bai, J.; Lv, J.; and Peng, X. 2022. Dual contrastive prediction for incomplete multi-view representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 4447–4461.

Mallah, C.; Cope, J.; Orwell, J.; et al. 2013. Plant leaf classification using probabilistic integration of shape, texture and margin features. *Signal Processing, Pattern Recognition and Applications*, 5(1): 45–54.

Pu, J.; Cui, C.; Chen, X.; Ren, Y.; Pu, X.; Hao, Z.; Yu, P. S.; and He, L. 2024. Adaptive feature imputation with latent graph for deep incomplete multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 14633–14641.

Ren, Y.; Ke, J.; Wen, Z.; Wu, T.; Yang, Y.; Pu, X.; and He, L. 2025. Multi-View Graph Clustering via Node-Guided Contrastive Encoding. In *Forty-second International Conference on Machine Learning*.

Ren, Y.; Pu, J.; Cui, C.; Zheng, Y.; Chen, X.; Pu, X.; and He, L. 2024. Dynamic weighted graph fusion for deep multi-view clustering. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 4842–4850.

Shao, Z.; Xu, Y.; Wei, W.; Wang, F.; Zhang, Z.; and Zhu, F. 2022. Heterogeneous graph neural network with multi-view representation learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(11): 11476–11488.

Tang, H.; and Liu, Y. 2022. Deep safe incomplete multi-view clustering: Theorem and algorithm. In *International Conference on Machine Learning*, 21090–21110. PMLR.

Tang, J.; Yi, Q.; Fu, S.; and Tian, Y. 2024. Incomplete multi-view learning: Review, analysis, and prospects. *Applied Soft Computing*, 153: 111278.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Wang, H.; Zhang, W.; and Ma, X. 2024. Contrastive and adversarial regularized multi-level representation learning for incomplete multi-view clustering. *Neural Networks*, 172: 106102.

Wang, J.; and Feng, S. 2024. Contrastive and view-interaction structure learning for multi-view clustering. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 5055–5063.

Wang, J.; Feng, S.; Lyu, G.; and Yuan, J. 2024. Surer: Structure-adaptive unified graph neural network for multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15520–15527.

Wen, J.; Wu, Z.; Zhang, Z.; Fei, L.; Zhang, B.; and Xu, Y. 2021a. Structural deep incomplete multi-view clustering network. In *Proceedings of the 30th ACM international conference on information & knowledge management*, 3538–3542.

Wen, J.; Zhang, Z.; Xu, Y.; Zhang, B.; Fei, L.; and Xie, G.-S. 2021b. CDIMC-net: cognitive deep incomplete multi-view clustering network. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 3230–3236.

Wen, Z.; Wu, T.; Ren, Y.; Ling, Y.; Cui, C.; Pu, X.; and He, L. 2024. Dual-Optimized Adaptive Graph Reconstruction for Multi-View Graph Clustering. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 1819–1828.

Wu, S.; Zheng, Y.; Ren, Y.; He, J.; Pu, X.; Huang, S.; Hao, Z.; and He, L. 2024. Self-weighted contrastive fusion for deep multi-view clustering. *IEEE Transactions on Multimedia*, 26: 9150–9162.

Xu, C.; Tao, D.; and Xu, C. 2013. A survey on multi-view learning. *arXiv preprint arXiv:1304.5634*.

Xu, G.; Wen, J.; Liu, C.; Hu, B.; Liu, Y.; Fei, L.; and Wang, W. 2024. Deep variational incomplete multi-view clustering: Exploring shared clustering structures. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16147–16155.

Xu, J.; Li, C.; Ren, Y.; Peng, L.; Mo, Y.; Shi, X.; and Zhu, X. 2022. Deep incomplete multi-view clustering via mining cluster complementarity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8761–8769.

Xu, J.; Ren, Y.; Li, G.; Pan, L.; Zhu, C.; and Xu, Z. 2021. Deep embedded multi-view clustering with collaborative training. *Information Sciences*, 573: 279–290.

Yang, Y.; and Newsam, S. 2010. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 270–279.

Yang, Y.; and Wang, H. 2018. Multi-view clustering: A survey. *Big Data Mining and Analytics*, 1(2): 83–107.

Yang, Z.; Zhang, H.; Wei, Y.; Wang, Z.; Nie, F.; and Hu, D. 2024. Geometric-inspired graph-based incomplete multi-view clustering. *Pattern Recognition*, 147: 110082.

Yu, X.; Chao, G.; Jiang, Y.; Ke, G.; and Chu, D. 2025. Incomplete Multi-View Clustering via Mutual Information. *IEEE Transactions on Multimedia*.

Yu, Z.; Dong, Z.; Yu, C.; Yang, K.; Fan, Z.; and Chen, C. L. P. 2024. A review on multi-view learning. *Frontiers of Computer Science*, 19(7): 197334.

Zhou, L.; Du, G.; Lue, K.; Wang, L.; and Du, J. 2024. A survey and an empirical evaluation of multi-view clustering approaches. *ACM Computing Surveys*, 56(7): 1–38.