SURER: Structure-Adaptive Unified Graph Neural Network for Multi-View Clustering

Jing Wang[1], **Songhe Feng**[1*], **Gengyu Lyu**[2], **Jiazheng Yuan**[3]
[1]Key Laboratory of Big Data and Artificial Intelligence in Transportation (Ministry of Education), School of Computer and Information Technology, Beijing Jiaotong University
[2]Engineering Research Center of Intelligence Perception and Autonomous Control (Ministry of Education), Beijing University of Technology
[3] College of Science and Technology, Beijing Open University
{jing_w, shfeng}@bjtu.edu.cn, lyugengyu@bjut.edu.cn, jzyuan@139.com

Abstract

Deep Multi-view Graph Clustering (DMGC) aims to partition instances into different groups using the graph information extracted from multi-view data. The mainstream framework of DMGC methods applies graph neural networks to embed structure information into the view-specific representations and fuse them for the consensus representation. However, on one hand, we find that the graph learned in advance is not ideal for clustering as it is constructed by original multi-view data and localized connecting. On the other hand, most existing methods learn the consensus representation in a late fusion manner, which fails to propagate the structure relations across multiple views. Inspired by the observations, we propose a **S**tructure-adaptive **U**nified g**R**aph n**E**ural network for multi-view cluste**R**ing (**SURER**), which can jointly learn a heterogeneous multi-view unified graph and robust graph neural networks for multi-view clustering. Specifically, we first design a graph structure learning module to refine the original view-specific attribute graphs, which removes false edges and discovers the potential connection. According to the view-specific refined attribute graphs, we integrate them into a unified heterogeneous graph by linking the representations of the same sample from different views. Furthermore, we use the unified heterogeneous graph as the input of the graph neural network to learn the consensus representation for each instance, effectively integrating complementary information from various views. Extensive experiments on diverse datasets demonstrate the superior effectiveness of our method compared to other state-of-the-art approaches.

Introduction

In the big data era, the exploration of a comprehensive understanding of heterogeneous features such as images, videos, speech, and text is an important research problem (Wu et al. 2023). Multi-view clustering (MVC) algorithms (Ling et al. 2023; Xu et al. 2023) have found widespread application across many fields. Benefiting from the modeling capacity of deep neural networks, Deep Multi-view Clustering (DMVC) (Li et al. 2019; Yang et al. 2022) methods emerge to provide an effective solution for handling high-dimensional and complex multi-view data. DMVC usually aims at learning the consensus representation by exploring the consistency and complementarity of multi-view data. For instance, Xu et al. (Xu et al. 2022) utilize view-specific deep autoencoders to extract view-specific instance-level and cluster-level representations. After that, the contrastive losses are introduced to learn discriminative features and reliable clustering labels. Zhou et al. (Zhou and Shen 2020) also adopt deep networks to learn view-specific representations and introduce the adversarial learning mechanism to align the latent representations. Although DMVC achieves impressive clustering performance, its effectiveness is constrained by its failure to leverage the topological structure information among instances. To address this issue, many Deep Multi-view Graph Clustering (DMGC) methods (Huang et al. 2023; Xia et al. 2021) are proposed recently. The mainstream framework of DMGC methods, as depicted in Figure 1, initially utilizes multi-view data to construct topological graphs in advance. Subsequently, view-specific Graph Convolutional Networks (GCNs) are employed to encode the graph information into view-specific representations, which are then merged to create the consensus representation. For example, Wang et al. (Wang et al. 2021) adopt GCNs to explore the complementarity of multi-view data and introduce a mutual information maximization module to guide the consensus representation maintaining the structure information of multiple views. However, the graph used in the above methods is constructed by original multi-view data and remains fixed throughout the entire representation learning process. These graphs inherently incorporate

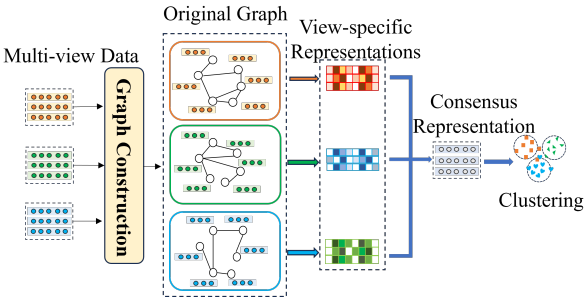


Figure 1: The framework of mainstream deep multi-view graph clustering. It usually constructs view-specific graphs in advance and then adopts graph neural networks to map the representations of various views to the common space.

*Corresponding author.

noise from the raw data, failing to guide the subsequent representation learning. Furthermore, existing DMGC methods usually learn the consensus representation in a late fusion manner, which only considers the intra-view structure information failing to propagate the structure relations across different views.

To alleviate the limitations of existing DMGC methods, we propose a structure-adaptive unified graph neural network for multi-view clustering, which can provide a robust structure graph and comprehensively fuse information from multiple views to learn the consensus representation. Specifically, we first utilize the original attribute graphs to update the instances latent representations with GCNs, which are further employed to refine the view-specific structure graphs via graph decoders. Then, we integrate the above refined view-specific structure graphs into the unified heterogeneous graph by connecting the latent representations, which are from the same instance. Furthermore, we utilize a heterogenous graph neural network to simultaneously propagate both the inter-view and intra-view relations, aiming to learn the consensus representation for each instance. Here, the inter-view relations reflect the instance alignment relationship and the intra-view relations denote the local structure relation within the same view. Finally, the consensus representation is employed to generate the final clustering results by the clustering layer. The contributions are summarized as follows:

- We propose a novel structure-adaptive unified graph neural network for MVC. By introducing a graph structure learning module, SURER can generate a structure-adaptive graph, enhancing the robustness of the structure graph used for subsequent representation learning.
- The proposed SURER learns the consensus representation from a unified heterogeneous attribute graph, which can not only effectively fuse the intra-view structure relations but also propagate the structure relations across multiple views.
- Extensive experimental results across eight datasets demonstrate the superior performance of our proposed method in comparison to other state-of-the-art multi-view clustering algorithms.

Related Work

In this section, we briefly review multi-view clustering methods, which can be roughly divided into three categories, namely, subspace-based methods (Cao et al. 2015), kernel-based methods (Liu 2021), and graph-based methods (Lin and Kang 2021). The subspace-based methods usually learn a low-dimensional subspace representation from multi-view data, which aims to handle the high-dimensional complex multi-view data. For example, Cuo (Guo 2013) formulates the subspace learning with multiple views as a unified optimization problem, incorporating a common subspace representation matrix and a group sparsity inducing norm. Zhang et al. (Zhang et al. 2019) propose an end-to-end framework, which utilizes the pseudo label to supervise the training of the deep convolutional network, resulting in superior performance. Kernel-based methods aim to provide an optimal combination of a group of pre-specified base kernels to improve the clustering performance. However, kernel-based MVC methods suffer from the problem of high computational complexity. For instance, Zhou et al. (Zhou et al. 2021) propose an efficient sampling strategy in multi-kernel clustering to enhance performance and speed, reducing memory and computational complexity to $\mathcal{O}(n)$ and $\mathcal{O}(n^2)$, respectively. These multi-view methods show satisfactory performance but they rarely consider the structure information between instances.

The graph-based methods for MVC primarily focus on identifying a consensus graph and obtaining the clustering result through spectral clustering. For instance, Tang et al. (Tang, Lu, and Dhillon 2009) propose a linked matrix factorization technique to fuse the information from multiple graphs. In order to reduce the computational complexity of graph-based methods, the anchor graph has been introduced, which selects a small set of representative samples named anchors to capture the manifold structure of each view. Lu et al. (Lu and Feng 2023) adopt a structure fusion strategy to integrate the view-specific anchor graphs for clustering, which greatly improves the representation capability of the target structure optimal graph. With the widespread adoption of deep learning, deep multi-view graph clustering has become increasingly popular to model graph information, which fully explores the structure information by graph neural networks. Xia et al. (Xia et al. 2021) utilize the clustering label to guide the representation learned by the multi-view shared graph attention encoder. Huang et al.

Method

In this paper, we propose a structure-adaptive unified graph neural network to provide robust topological structure and fully explore complementary information from multiple views for improving the performance of clustering. Figure 2 illustrates the framework of SURER.

Graph Structure Learning Module

Graph Neural Networks (GNNs) can fully discover the potential structure hidden in multi-view data and achieve better clustering performance. However, GNNs are highly sensitive to the quality of the given graph structures. In pursuit of an optimal graph for downstream tasks, we propose an adaptive graph structure learning module to refine the original graph. Specifically, given a multi-view dataset $\{\mathbf{X}^v \in \mathbb{R}^{d_v \times n}\}_{v=1}^m$ that includes n samples with m views. Here, the i-th column $\mathbf{x}_i^v \in \mathbb{R}^{d_v}$ in $\mathbf{X}^v$ represents the d_v-dimensional feature vector of sample i under the v-th view.

In most existing DMGC methods, the view-specific graph $\mathbf{A}^v \in \mathbb{R}^{n \times n}$ is constructed using an unsupervised method, such as Euclidean distance, to identify the nearest neighbors of each node. However, they utilize the original feature and local relationship to obtain the structure graph and assume $\mathbf{A}^v$ is fixed, adversely impacting the learning of representations in graph neural networks. Thus, we adopt the graph autoencoder networks to obtain the refined structure graph $\hat{\mathbf{A}}^v \in \mathbb{R}^{n \times n}$.

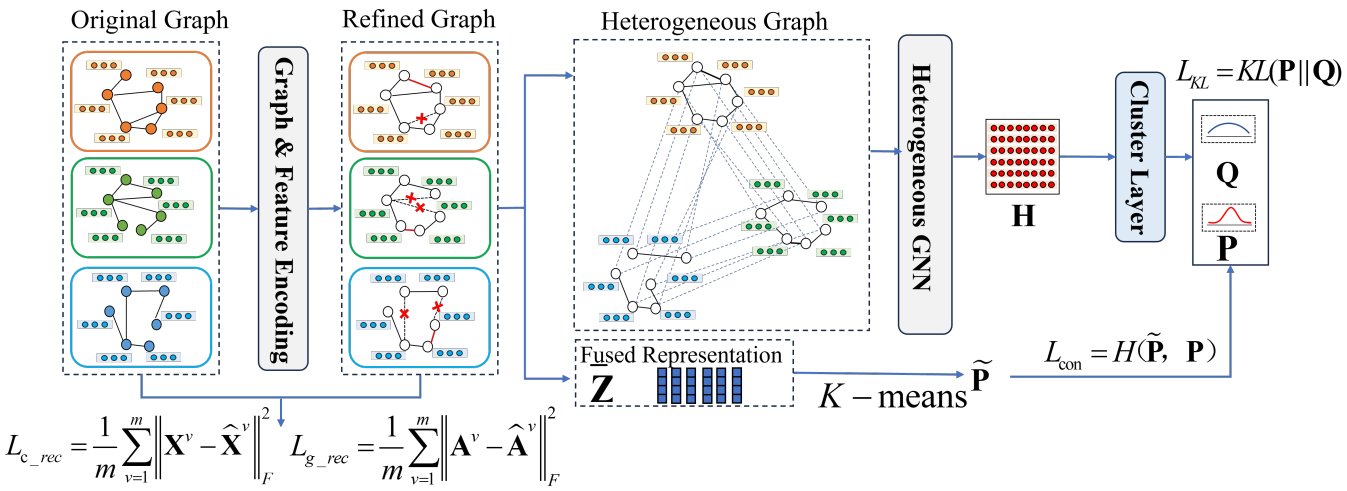


Figure 2: The framework of SURER. To provide more robust structure relations, SURER first adopts view-specific graph encoders to extract latent representations $\{Z^v\}_{v=1}^m$, which are employed to reconstruct robust view-specific graphs $\{\hat{A}^v\}_{v=1}^m$. Then, we build a unified heterogeneous graph $\mathcal{G}$ by linking all view-specific graphs $\{\hat{A}\}_{v=1}^m$ through the instance alignment relationship across multiple views. Furthermore, we adopt the heterogeneous graph neural network to propagate the structure information across different views and obtain an instance consensus representation matrix H. Finally, the consensus representation H is treated as the input of the clustering layer to get clustering results, which are supervised by the clustering information contained in latent representations $\{Z^v\}_{v=1}^m$ and KL divergence loss.

(a)View-specific graph encoders. Based on the original structure graph $\mathbf{A}^v$, we utilize the GCN to embed the neighbor structure relationship into instance latent representations $\mathbf{Z}^{v,l} \in \mathbb{R}^{d_{(v,l)} \times n}$, which can be obtained by following graph convolutional operation:

$$\mathbf{Z}^{(v,l)} = f^{(v,l)}(\mathbf{Z}^{(v,l-1)}, \hat{\mathbf{A}}^v; \theta^v) \tag{1}$$

where, the computation of the l-th graph convolutional layer $f^{(v,l)}$ is formulated as follows:

$$\mathbf{Z}^{(v,l)} = \sigma(\widetilde{\mathbf{D}}_v^{\frac{1}{2}} \widetilde{\mathbf{A}}^v \widetilde{\mathbf{D}}_v^{\frac{1}{2}} \mathbf{Z}^{(v,l-1)} \mathbf{W}^{(v,l)} + \mathbf{b}^{(v,l)}) \tag{2}$$

where, $\widetilde{\mathbf{A}}^v = \mathbf{A}^v + \mathbf{I}_n \in \mathbb{R}^{n \times n}$ is the self-connection variant of $\mathbf{A}^v$, and $\widetilde{\mathbf{D}}_{ii}^v = \sum_j \mathbf{A}_{ij}^v \in \mathbb{R}^{n \times n}$ is the degree matrix of $\mathbf{A}^v$. Furthermore, σ denotes the activation function of the l-th layer. Besides, the input of the first layer is the multi-view data $\{\mathbf{X}^v \in \mathbb{R}^{d_v \times n}\}_{v=1}^m$, and the corresponding latent representations $\mathbf{Z}^{v,1}$ is obtained by:

$$\mathbf{Z}^{(v,1)} = \sigma(\widetilde{\mathbf{D}}_v^{\frac{1}{2}} \widetilde{\mathbf{A}}^v \widetilde{\mathbf{D}}_v^{\frac{1}{2}} \mathbf{X}^v \mathbf{W}^{(v,1)} + \mathbf{b}^{(v,1)}) \tag{3}$$

(b)View-specific feature decoders. After obtaining m view-specific latent representations $\{\mathbf{Z}^v\}_{v=1}^m$ by graph encoder networks, the view-specific feature decoders are introduced to extract the content information of instances underlying different views. Specifically, we use the view-specific decoder network $g^v(\cdot)$ with parameters μ^v to reconstruct multi-view data $\hat{\mathbf{X}}^v = g^v(\mathbf{Z}^v; \mu^v)$, where the decoder network is composed of fully-connected layers. In order to learn the content information of instances, the reconstruction loss between the original data and their corresponding reconstructed features is defined by:

$$\mathcal{L}_{c\_rec} = \frac{1}{m}\sum_{v=1}^{m}\left\|\mathbf{X}^v - \hat{\mathbf{X}}^v\right\|_F^2 \tag{4}$$

(c)View-specific graph decoders. For mitigating the negative impact of noise present in the original data, we utilize the features extracted by the graph encoders to reconstruct the clean graph $\{\hat{\mathbf{A}}^v\}_{v=1}^m$. Since the representations $\{\mathbf{Z}^v\}_{v=1}^m$ extracted by view-specific graph encoders contain comprehensive structure and view-specific content information, we directly employ an inner product decoder to predict the affinity relationships between instances, resulting in a more robust graph. The graph decoder network of v-th view can be written as:

$$\hat{\mathbf{A}}^v = sigmoid(\mathbf{Z}^v \mathbf{W}^v \mathbf{Z}^{v^T}) \tag{5}$$

where $\mathbf{W}^v$ is the learned parameter of the v-th decoder.

To preserve the structure of the refined graph $\hat{\mathbf{A}}^v$, we minimize the graph reconstruction error for each individual view:

$$\mathcal{L}_{g\_rec} = \frac{1}{m}\sum_{v=1}^{m}\left\|\mathbf{A}^v - \hat{\mathbf{A}}^v\right\|_F^2 \tag{6}$$

Unified Heterogeneous Graph Construction

The view-specific graph $\{\hat{\mathbf{A}}^v\}_{v=1}^m$ and latent representations $\{\mathbf{Z}^v\}_{v=1}^m$ retain a lot of view-private information. By fully analyzing complementary information from multiple views, we can achieve more accurate clustering results compared to relying on a single view. Existing deep multi-view graph clustering methods usually use different methods, such as

contrastive learning to learn the consensus representation, or perform convolutions on a fusion graph to obtain the consensus representation. However, they focus on exploring the structure relationships within a single view and fail to propagate the structure relationships across different views.

In our work, we construct a novel unified attribute heterogeneous graph, which simultaneously contains the intra-view edges and inter-view edges. Specifically, we establish connections between distinct representations $\mathbf{Z}_i^m$ and $\mathbf{Z}_i^n$, which correspond to the same instance i observed from different views. The operation contributes to uncovering the structure relation across different views. For intra-view structure relation exploring, if instance j is the neighbor of i then there exists an edge between $\mathbf{Z}_j^v$ and $\mathbf{Z}_i^v$.

Heterogeneous Graph Relations Propagation

To encode both intra-view and inter-view structure relations, we employ heterogeneous graph neural networks to learn the consensus representation $\mathbf{H} \in \mathbb{R}^{h \times n}$. The initial representations $\{\mathbf{Z}^v\}_{v=1}^m$ are obtained by view-specific graph encoders, which are fed into the heterogeneous graph neural network. Specifically, the heterogeneous graph neural network iteratively refines the instance representation with the message propagation as follows:

$$\mathbf{H}_i^v = \sigma(\mathbf{W}_1\mathbf{Z}_i^v + \sum_{j=1}^{k}\frac{1}{k}\mathbf{W}_2\mathbf{Z}_j^v + \sum_{n=1}^{m-1}\frac{1}{m-1}\mathbf{W}_3\mathbf{Z}_i^n) \quad (7)$$

where $\mathbf{W}_1$, $\mathbf{W}_2$, and $\mathbf{W}_3$ are learned weight matrices, k denotes the number of neighbors. Furthermore, to consider the neighboring information of instances on other views and fuse more inter-view complementary information into each instance representation, in experiments, we also exploit the outputs of Eq (7) as its inputs and repeat such propagation operation. According to the above operation, we can obtain the consensus representation $\mathbf{H}$:

$$\mathbf{H} = \frac{1}{m}\sum_{v=1}^{m}\mathbf{H}_i^v \quad (8)$$

Self-supervised Multi-view Clustering

Most existing multi-view clustering methods employ either K-means or spectral clustering algorithm on the consensus matrix to obtain the clustering results. In our work, we introduce a clustering layer constrained by KL divergence loss to achieve final results. Specifically, we first leverage Student t-distribution to generate the soft assignments $\mathbf{P}$ for all instances based on the consensus representation $\mathbf{H}$, and the p_{ij} can be calculated as:

$$p_{ij} = \frac{(1 + \|\mathbf{H}_i - \mu_j\|^2)^{-1}}{\sum_j^C (1 + \|\mathbf{H}_i - \mu_j\|^2)^{-1}} \quad (9)$$

where C present the number of clusters, μ_j is the centroid of j-th cluster, learning during the training state. p_{ij} presents the probability about assigning instance i into cluster j. Moreover, the target distribution $\mathbf{Q} \in \mathbb{R}^{n \times C}$ is introduced to supervise the clustering process, which is defined as:

$$q_{ij} = \frac{(p_{ij})^2 / \sum_i p_{ij}}{\sum_j^C ((p_{ij})^2 / \sum_i p_{ij})} \quad (10)$$

Algorithm 1: The Algorithm of **SURER**

Input: Multi-view data $\{\mathbf{X}^{(v)}\}_{v=1}^m$;Training iterations T.
Process: 1. Construct the graphs for each view and obtain the view-specific affinity matrices $\{\mathbf{A}^{(v)}\}_{v=1}^m$.
Pretrain: 2. Pertrain the graph structure learning module by optimizing $\mathcal{L}_{c\_rec}$,$\mathcal{L}_{g\_rec}$ in Eq (4) and (6).
Finetuning: 3. for epoch = 1 to T
4. Obtain the instance latent representations $\{\mathbf{Z}_j^v\}_{v=1}^m$ and refined affinity matrices $\{\hat{\mathbf{A}}^v\}_{v=1}^m$ by Eq (1) and (5).
5. Integrate the m individual attribute graphs into a unified heterogeneous graph by connecting different feature representation nodes within an instance.
6. Update $\mathbf{H}_i^v$ by Eq (7).
7. Repeat Step 6 and obtain $\mathbf{H}$ by Eq (8).
8. Update the model parameters by minimizing Eq (14).
9. end for
Output: the clustering result y by Eq (13).

Thus, the KL divergence loss can be formulated as:

$$\mathcal{L}_{KL} = KL(\mathbf{Q}\|\mathbf{P}) = \sum_{i=1}^{n}\sum_{j=1}^{C} q_{ij}\mathbf{log}\frac{q_{ij}}{p_{ij}} \quad (11)$$

As the number of convolutional layers increases, instance representation tends to become increasingly similar, leading to the reduction of diversity and information. Thus, we leverage the clustering information contained in the latent representations $\{\mathbf{Z}^v\}_{v=1}^m$ to improve the clustering effectiveness of the soft assignments $\mathbf{P}$. Specifically, we first weighted fuse the latent representations, i.e., $\overline{\mathbf{Z}} = \frac{1}{m}\sum_{v=1}^m \mathbf{Z}_i^v$, which is treated as the input of K-means to achieve the clustering result $\widetilde{\mathbf{P}} \in \mathbb{R}^{n \times C}$. Then, we regard $\widetilde{\mathbf{P}}$ as anchors to modify $\mathbf{P}$ by minimizing the following cross-entry loss:

$$\mathcal{L}_{con} = H(\widetilde{\mathbf{P}}, \mathbf{P}) = -\sum_{i=1}^{N}\widetilde{\mathbf{P}}_i\mathbf{log}\mathbf{P}_i \quad (12)$$

Concretely, the clustering result of the i-th instance is obtained by:

$$y_i = \arg\max_j p_{ij} \quad (13)$$

The Overall Loss Function of SURER

In summary, we have introduced a structure-adaptive unified graph neural network for multi-view clustering. In the training stage, the graph structure learning module, heterogeneous graph neural network, and the clustering layer are jointly optimized according to the following objective function:

$$\mathcal{L} = \mathcal{L}_{c\_rec} + \mathcal{L}_{g\_rec} + \lambda_1\mathcal{L}_{con} + \lambda_2\mathcal{L}_{KL} \quad (14)$$

where λ_1, and λ_2 are hyper-parameters to balance the weight of different loss functions. The whole learning process of SURER is summarized in Algorithm 1.

Experiments

In this section, we evaluate the proposed SURER model on eight widely used multi-view datasets to verify the effec-

Data	Samples	Clusters	Views	View dimensions
MSRCv1	210	7	5	24/576/512/256/254
BBCSports	544	5	2	3183/3203
100leaves	1600	100	3	60/60/60
Mfeat	2000	10	6	216/76/64/6/240/47
Scene15	4568	15	3	20/59/40
VOC	5649	20	2	512/399
Hdigit	10000	10	2	784/256
Noisyminist	13000	10	2	784/784

Table 1: Statistical characteristics of eight datasets.

tiveness and superiority of SURER. Furthermore, an ablation study and parameter sensitivity analysis are conducted to investigate the properties of the SURER.

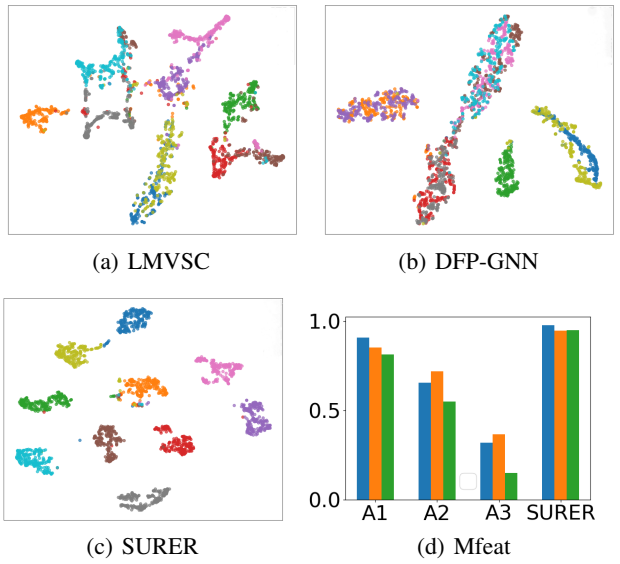
Experimental Settings

Datasets. In our experiments, we employ eight widely-used multi-view datasets to evaluate the performance of SURER, whose detailed characteristics are illustrated in Table 1. MSRCv1 (Winn and Jojic 2005) contains 210 objects from 7 clusters, each of which is represented by five visual features. BBCSports (Greene and Cunningham 2006) is a document dataset containing 544 instances with 5 distinct classes, each image is characterized by two features. 100leaves (Wang, Yang, and Liu 2019) contains 1600 leaves from 100 clusters with three kinds of features. Mfeat (Wang, Yang, and Liu 2019) is a hand-written dataset, which includes 2000 numbers from 10 clusters and is represented by six features. Scene15 (Li and Perona 2005) consists of 4568 natural scenes categorized into 15 groups, where each scene extracts three types of features: GIST, SIFT, and LBP. VOC (Hwang and Grauman 2010) is composed of 5649 image-text pairs covering 20 clusters, where each image is presented by the Gist feature, and each text is composed of word frequency counts. Hdigit (Chen et al. 2022) is a digit dataset from MNIST Handwritten Digits and USPS Handwritten Digits, which consists of 10000 instances described by 2 views. Noisyminist (Wang et al. 2015) is also a handwritten dataset characterized by two heterogeneous features, that are composed of 70k images belonging to 10 different categories. As most baseline models struggle with such a vast dataset, we randomly chose 13k instances from Noisyminist.

Baseline models. All compared methods are implemented according to the source codes released by the authors, and the optimal parameters are set according to the suggestion in the corresponding literature.

- **MCGC** (Zhan et al. 2019) proposes a novel approach by using a disagreement cost function to align view-specific graphs towards the consensus graph.
- **LMSC** (Zhang et al. 2017) extracts latent representations from multi-view data, facilitating the generation of an accurate instance structure graph to enhance downstream clustering tasks.
- **LMVSC** (Kang et al. 2020) introduces an anchor graph construction method and a graph fusion mechanism simultaneously to effectively tackle the challenge of large-scale multi-view subspace clustering.
- **CDIMC** (Wen et al. 2020) uses a cognitive-inspired self-paced K-means clustering layer that identifies high-confidence samples, effectively mitigating the influence of outliers.
- **SiMVC** (Trosten et al. 2021) is a deep multi-view clustering approach that leverages autoencoder networks to acquire view-specific representations and subsequently merge them into a unified final representation.
- **CoMVC** (Trosten et al. 2021) extends the SiMVC by introducing a selective contrastive learning mechanism to formulate the consensus representation, where the positive pairs consist of samples sharing the same pseudo-label assignments.
- **MFLVC** (Xu et al. 2022) simultaneously applies instance-level and cluster-level contrastive objects, enhancing feature extracting and clustering results in an end-to-end manner.
- **DFP-GNN** (Xiao et al. 2023) is composed of three submodules, namely view-specific, cross-view propagation, and fusion module, which aims at capturing consistency and complementarity information from multi-view data.

Metrics. Seven commonly employed metrics, specifically ACC, NMI, PUR, ARI, PRE, REC, and F-score, are utilized to evaluate the clustering performance of all algorithms. The detailed definitions of these metrics are elaborated in the (Cao et al. 2015).



(a) LMVSC (b) DFP-GNN (c) SURER (d) Mfeat

Figure 3: (a),(b), and (c) The visualizations of the consensus representation of LMVSC, DFP-GNN, and SURER on the Mfeat dataset. (d) Ablation study on the Mfeat dataset.

Implementation Details. For all datasets, the view-specific graph encoder module consists of a three-layer graph convolutional encoder with a ReLU activation function, and the dimension is set as $\{d_v, 512, 2048, 256\}$. The view-specific feature decoders are formed by four fully-connected layers and the dimensions are respectively set as

Data	Metric	MCGC	LMSC	LMVSC	CDIMIC	SiMVC	CoMVC	MFLVC	DFP-GNN	**SURER**
100leaves	ACC	0.5769	0.4394	0.5306	0.8569	0.7341	0.7024	0.7344	0.5856	**0.9100**
	NMI	0.7009	0.6845	0.7419	**0.9545**	0.9029	0.8906	0.8725	0.8382	0.9538
	PUR	0.6056	0.4688	0.6044	0.8994	0.7762	0.7450	0.7594	0.6150	**0.9125**
	ARI	0.1307	0.3106	0.3792	0.8250	0.6776	0.6421	0.6341	0.4876	**0.8540**
	PRE	0.0811	0.2988	0.4497	0.8505	0.6938	0.6612	0.6670	0.5043	**0.8691**
	REC	0.6970	0.3387	0.3379	0.8995	0.7890	0.7659	0.7320	0.7348	**0.9032**
	F-score	0.1453	0.3175	0.3859	0.8743	0.7382	0.7096	0.6980	0.5981	**0.8858**
BBCSports	ACC	0.3842	0.4228	0.3346	0.4301	0.2451	0.2643	0.6544	0.9430	**0.9632**
	NMI	0.0373	0.2468	0.1428	0.1946	0.0158	0.0220	0.4040	0.8440	**0.8911**
	PUR	0.3842	0.4908	0.5919	0.4504	0.3682	0.3842	0.6544	0.9581	**0.9780**
	ARI	0.0180	0.1565	0.0247	0.0618	0.0027	0.0007	0.3607	0.8525	**0.9013**
	PRE	0.2455	0.3423	0.4210	0.3306	0.2480	0.2522	0.5326	0.8925	**0.9295**
	REC	0.9593	0.4265	0.2524	0.8885	0.2113	0.2206	0.5155	0.8937	**0.9296**
	F-score	0.3910	0.3798	0.3156	0.4819	0.2282	0.2328	0.5239	0.8931	**0.9295**
Hdigit	ACC	0.5814	0.6681	0.5482	0.5071	0.7435	0.8027	0.9478	0.8847	**0.9834**
	NMI	0.6339	0.6207	0.5050	0.5412	0.7638	0.8244	0.8834	0.8810	**0.9544**
	PUR	0.5816	0.7151	0.6045	0.5199	0.7564	0.8175	0.9478	0.8919	**0.9841**
	ARI	0.5386	0.5269	0.3544	0.3584	0.6893	0.7680	0.8885	0.8312	**0.9636**
	PRE	0.4488	0.5625	0.4937	0.4361	0.6975	0.7759	0.9002	0.8414	**0.9673**
	REC	0.9060	0.5886	0.3770	0.5386	0.6933	0.7722	0.9004	0.8663	**0.9673**
	F-score	0.6002	0.5753	0.4275	0.4820	0.6541	0.7349	0.9003	0.8537	**0.9673**
VOC	ACC	0.2935	0.1837	0.1637	0.1855	0.5376	0.5151	0.5249	0.6113	**0.9786**
	NMI	0.1387	0.1246	0.1357	0.1346	0.5511	0.5307	0.4570	0.5350	**0.9423**
	PUR	0.2953	0.2822	0.1772	0.2857	0.6640	0.6435	0.5304	0.6375	**0.9791**
	ARI	0.1116	0.0657	0.0523	0.0520	0.4788	0.4173	0.3152	0.4731	**0.9532**
	PRE	0.1395	0.1693	0.0939	0.1519	0.5533	0.5358	0.3566	0.4998	**0.9580**
	REC	0.8007	0.1013	0.1456	0.1393	0.4254	0.4000	0.5858	0.4854	**0.9581**
	F-score	0.2337	0.1252	0.1142	0.1451	0.4806	0.4579	0.4433	0.4925	**0.9581**

Table 2: The clustering performance comparisons on 100leaves, BBCSports, Hdigit, and VOC datasets.

$\{256, 2048, 512, d_v\}$, where RELU is selected as the activation function. For the hyperparameters (i.e., λ_1 and λ_2) configuration, a grid searching method is adopted to select the optimal values from $\{0.01, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0\}$. In order to enhance the robustness of refined attribute graphs, we first conduct an initial pre-training phase for the GSL module spanning some epochs. Subsequently, we optimize GSL and the HGNN jointly.

Comparison Results

In Table 2 and 3, we present a comprehensive comparison of our proposed method with the other eight compared methods on all evaluation metrics. Notably, the best and sub-optimal performances are indicated by bold and underlined formatting, respectively. In Figure 3, we adopt t-SNE to visualize the consensus representation learned by LMVSC, DFP-GNN, and SURER on the Mfeat. By analyzing the data in Table 2-3, and Figure 3 we can observe that:

(1) The proposed SURER significantly and consistently outperforms all compared multi-view clustering methods in most cases. The encouraging performance demonstrates that our graph structure learning module and unified heterogeneous graph construction method can provide more robust structure relations and fully consider the multi-view complementary information. (2) Compared with other multi-view graph clustering methods (i.e., DFP-GNN, LMVSC, and MCGC), our SURER obtains the best clustering results. Notably, DFP-GNN also adopts deep graph neural networks to explore the structure relations between instances. The phenomenon indicates that our graph structure learning module can mitigate the influence of noise from the original attribute graphs. Furthermore, the unified heterogeneous graph constructed in our method can propagate the structure information across different views. (3) As illustrated in Figure 3(a),(b), and (c), SURER is capable of acquiring more distinct cluster structures. This observation demonstrates the capacity of SURER to effectively explore structure relationships among samples, thereby grouping similar instances into the same cluster.

Ablation Study

To analyze the effectiveness of the graph structure learning module and the construction of the unified heterogeneous graph, we perform an ablation study and compare our proposed SURER with its degenerated methods. In Figure 3(d), we analyze the following cases:

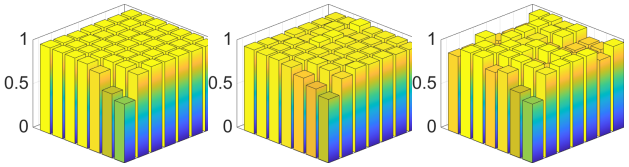
- **A1**: SRUER w.o. the graph structure learning module. In this case, we use original structure graph $\{\mathbf{A}^v\}_{v=1}^m$ and latent representation $\{\mathbf{Z}^v\}_{v=1}^m$ to construct the unified heterogeneous graph.
- **A2**: SRUER w.o. the construction of the unified heterogeneous graph. The heterogeneous GNN is replaced by the view-specific graph neural networks to obtain multiple representations, which are fused into a consensus one for subsequent clustering.
- **A3**: SRUER w.o. the graph structure learning module and unified heterogeneous graph. We adopt view-specific GNNs to learn the latent representations $\{\mathbf{Z}^v\}_{v=1}^m$, subsequently merging them into the consensus representation

Data	Metric	MCGC	LMSC	LMVSC	CDIMIC	SiMVC	CoMVC	MFLVC	DFP-GNN	**SURER**
MRSCv1	ACC	0.7476	0.3286	0.3429	0.8476	0.6019	0.5914	0.6190	0.9238	**1.0000**
	NMI	0.6491	0.2291	0.2460	0.7814	0.54008	0.5363	0.5743	0.8429	**1.0000**
	PUR	0.7667	0.3381	0.3810	0.8524	0.6609	0.6391	0.4244	**0.9286**	0.8571
	ARI	0.5838	0.1026	0.1250	0.7053	0.5037	0.5073	0.6333	0.8291	**1.0000**
	PRE	0.6112	0.2263	0.2496	0.7582	0.5318	0.5359	0.5252	0.8602	**1.0000**
	REC	0.6821	0.2299	0.2501	0.7819	0.5168	0.5211	0.5254	0.8599	**1.0000**
	F-score	0.6447	0.2281	0.2474	0.7699	0.4177	0.4128	0.5253	0.8595	**1.0000**
Scene15	ACC	0.2085	0.3610	0.3222	0.4109	0.4383	0.4347	0.3173	0.2732	**0.4696**
	NMI	0.1762	0.3439	0.3396	0.4116	**0.4657**	0.4627	0.3392	0.2741	0.4531
	PUR	0.2107	0.4042	0.3922	0.4769	0.5083	0.5001	0.3456	0.3050	**0.5284**
	ARI	0.0835	0.2013	0.1714	0.2333	0.2787	0.2710	0.1784	0.1589	**0.2938**
	PRE	0.1119	0.2548	0.2521	0.3143	0.3433	0.3353	0.2304	0.1788	**0.3506**
	REC	0.6887	0.2593	0.2165	0.3232	0.3507	0.3497	0.2641	0.2998	**0.3533**
	F-score	0.1925	0.2570	0.2330	0.3187	0.3470	0.3423	0.2461	0.2238	**0.3519**
Mfeat	ACC	0.9540	0.1907	0.6550	0.8360	0.8001	0.7750	0.8300	0.9490	**0.9765**
	NMI	0.9070	0.1285	0.6386	0.8882	0.8407	0.8243	0.8093	0.9070	**0.9470**
	PUR	0.9540	0.2779	0.7461	0.8705	0.8413	0.8147	0.8300	0.8520	**0.9775**
	ARI	0.9005	0.0696	0.5283	0.8145	0.7563	0.7207	0.7358	0.8900	**0.9480**
	PRE	0.9083	0.1684	0.6218	0.8423	0.7853	0.7610	0.7641	0.9088	**0.9552**
	REC	0.9126	0.1044	0.5413	0.8962	0.8257	0.7967	0.7707	0.9498	**0.9545**
	F-score	0.9105	0.1289	0.5788	0.8588	0.8047	0.7757	0.7674	0.9082	**0.9548**
Noisymnisit	ACC	0.4916	0.3887	0.2878	0.2150	0.3948	0.3918	0.4565	0.4135	**0.5282**
	NMI	**0.5303**	0.3471	0.3109	0.1124	0.3275	0.3980	0.4617	0.4185	0.5178
	PUR	0.4922	0.4484	0.4708	0.2581	0.4275	0.4432	0.4708	0.4244	**0.5357**
	ARI	**0.3983**	0.2303	0.1625	0.0498	0.2238	0.2580	0.3396	0.2743	0.2338
	PRE	0.3319	0.3008	0.3503	0.1465	0.3050	0.3498	0.3729	0.3290	**0.4722**
	REC	**0.9133**	0.3198	0.2153	0.2951	0.2985	0.3404	0.5205	0.4571	**0.6110**
	F-score	0.4859	0.3100	0.2667	0.1958	0.3018	0.3450	0.4345	0.3817	**0.5327**

Table 3: The clustering performance comparisons on MSRCv1, Scene15, Mfeat, and Noisymnisit datasets.

$\mathbf{H}$ for clustering.

According to Figure 3(d), A1 and A2 show better performance than A3, which clearly demonstrates that each kind of strategy of SURER can improve the clustering performances effectively, especially in constructing the unified graph by linking view-specific structure graphs from multiple views. Moreover, SURER outperforms A1, and A2 demonstrating that our method seamlessly integrates the graph structure learning module and a unified heterogeneous graph into deep multi-view graph clustering, which ensures the comprehensive exploration of the multi-view complementarity information and structure information.



(a) ACC($\lambda_1 vs \cdot \lambda_2$) (b) NMI($\lambda_1 vs \cdot \lambda_2$) (c) PUR($\lambda_1 vs \cdot \lambda_2$)

Figure 4: Parameters sensitivity analysis: the clustering performances (ACC, NMI, and PUR) with different parameters λ_1 and λ_2 on the Mfeat dataset.

Parameter Sensitivity Analysis

In this subsection, we analyze the parameter sensitivity of SURER with respect to its two employed parameters (i.e., λ_1 and λ_2). Specifically, we adopt the cross-validation strategy to fine-tune λ_1, and λ_2 within the range of $\{0.01, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0\}$. Figure 4 illustrates the experiment results of ACC, NMI, and PUR on the Mfeat dataset. These experiment results indicate that our approach achieves good performance across an extensive parameter range, further emphasizing the insensitivity of our SURER framework to variations in λ_1 and λ_2.

Conclusions

In this work, we propose a structure-adaptive unified graph neural network to perform multi-view clustering. To provide a robust structure graph for each view, the graph structure learning module is employed, which explores both structure and content information to reconstruct the corresponding refined graph. The heterogeneous unified graph is further constructed to propagate the structure relations within and across different views, serving to obtain the final consensus representation. By fully capturing the topological structure from multiple views and comprehensively fusing them for clustering, our framework enjoys more powerful clustering performance, which has been verified on various multi-view datasets as compared with state-of-art methods.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (No.2022JBZY019), the Beijing Natural Science Foundation (No. 4242046), and

the National Natural Science Foundation of China (No. 62306020).

References

Cao, X.; Zhang, C.; Fu, H.; Liu, S.; and Zhang, H. 2015. Diversity-induced multi-view subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 586–594.

Chen, M.-S.; Lin, J.-Q.; Li, X.-L.; Liu, B.-Y.; Wang, C.-D.; Huang, D.; and Lai, J.-H. 2022. Representation learning in multi-view clustering: A literature review. *Data Science and Engineering*, 225–241.

Greene, D.; and Cunningham, P. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *In International Conference on Machine Learning*, 377–384.

Guo, Y. 2013. Convex subspace representation learning from multi-view data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 387–393.

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*, 7936–7943.

Hwang, S. J.; and Grauman, K. 2010. Accounting for the relative importance of objects in image retrieval. In *BMVC*, 5.

Kang, Z.; Zhou, W.; Zhao, Z.; Shao, J.; Han, M.; and Xu, Z. 2020. Large-scale multi-view subspace clustering in linear time. In *Proceedings of the AAAI conference on artificial intelligence*, 4412–4419.

Li, F.-F.; and Perona, P. 2005. A bayesian hierarchical model for learning natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition*, 524–531.

Li, Z.; Wang, Q.; Tao, Z.; Gao, Q.; Yang, Z.; et al. 2019. Deep adversarial multi-view clustering network. In *International Joint Conference on Artificial Intelligence*, 4.

Lin, Z.; and Kang, Z. 2021. Graph filter-based multi-view attributed graph clustering. In *International Joint Conference on Artificial Intelligence*, 2723–2729.

Ling, Y.; Chen, J.; Ren, Y.; Pu, X.; Xu, J.; Zhu, X.; and He, L. 2023. Dual label-guided graph refinement for multi-view graph clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 8791–8798.

Liu, X. 2021. Incomplete multiple kernel alignment maximization for clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1.

Lu, X.; and Feng, S. 2023. Structure diversity-induced anchor graph fusion for multi-view clustering. *ACM Transactions on Knowledge Discovery from Data*, 1–18.

Tang, W.; Lu, Z.; and Dhillon, I. S. 2009. Clustering with multiple graphs. In *IEEE International Conference on Data Mining*, 1016–1021.

Trosten, D. J.; Lokse, S.; Jenssen, R.; and Kampffmeyer, M. 2021. Reconsidering representation alignment for multi-view clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1255–1265.

Wang, H.; Yang, Y.; and Liu, B. 2019. GMC: Graph-based multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 1116–1129.

Wang, W.; Arora, R.; Livescu, K.; and Bilmes, J. 2015. On deep multi-view representation learning. In *International Conference on Machine Learning*, 1083–1092.

Wang, Y.; Chang, D.; Fu, Z.; and Zhao, Y. 2021. Consistent multiple graph embedding for multi-view clustering. *IEEE Transactions on Multimedia*, 1–10.

Wen, J.; Zhang, Z.; Xu, Y.; Zhang, B.; Fei, L.; and Xie, G. 2020. CDIMC-net: cognitive deep incomplete multi-view clustering network. In *International Joint Conference on Artificial Intelligence*, 3538–3542.

Winn, J.; and Jojic, N. 2005. Locus: Learning object classes with unsupervised segmentation. In *Tenth IEEE International Conference on Computer Vision*, 756–763.

Wu, Y.; Chi, Z.; Wang, Y.; and Feng, S. 2023. Metagcd: Learning to continually learn in generalized category discovery. In *Tenth IEEE International Conference on Computer Vision*, 1655–1665.

Xia, W.; Wang, Q.; Gao, Q.; Zhang, X.; and Gao, X. 2021. Self-supervised graph convolutional network for multi-view clustering. *IEEE Transactions on Multimedia*, 3182–3192.

Xiao, S.; Du, S.; Chen, Z.; Zhang, Y.; and Wang, S. 2023. Dual fusion-propagation graph neural network for multi-view clustering. *IEEE Transactions on Multimedia*, 1–13.

Xu, C.; Zhao, W.; Zhao, J.; Guan, Z.; Yang, Y.; Chen, L.; and Song, X. 2023. Progressive deep multi-view comprehensive representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 10557–10565.

Xu, J.; Tang, H.; Ren, Y.; Peng, L.; Zhu, X.; and He, L. 2022. Multi-level feature learning for contrastive multi-view clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 16051–16060.

Yang, M.; Li, Y.; Hu, P.; Bai, J.; Lv, J.; and Peng, X. 2022. Robust multi-view clustering with incomplete information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1055–1069.

Zhan, K.; Nie, F.; Wang, J.; and Yang, Y. 2019. Multiview consensus graph clustering. *IEEE Transactions on Image Processing*, 1261–1270.

Zhang, C.; Hu, Q.; Fu, H.; Zhu, P.; and Cao, X. 2017. Latent multi-view subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 4279–4287.

Zhang, J.; Li, C.-G.; You, C.; Qi, X.; Zhang, H.; Guo, J.; and Lin, Z. 2019. Self-supervised convolutional subspace clustering network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 5473–5482.

Zhou, R.; and Shen, Y.-D. 2020. End-to-end adversarial-attention network for multi-modal clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 14619–14628.

Zhou, S.; Ou, Q.; Liu, X.; Wang, S.; Liu, L.; Wang, S.; Zhu, E.; Yin, J.; and Xu, X. 2021. Multiple kernel clustering with compressed subspace alignment. *IEEE Transactions on Neural Networks and Learning Systems*, 252–263.