Hypergraph-Enhanced Contrastive Learning for Multi-View Clustering with Hyper-Laplacian Regularization

Zhibin Gu[1,2,3] **Weili Wang**[4*]
[1]College of Computer and Cyber Security, Hebei Normal University, China
[2]Hebei Provincial Key Laboratory of Network & Information Security, Hebei Normal University, China
[3]Hebei Provincial Engineering Research Center for Supply Chain Big Data Analytics & Data Security, Hebei Normal University, China
[4]Independent Researcher, China
guzhibin@hebtu.edu.cn, weiliw295@gmail.com

Abstract

Deep multi-view clustering (DMVC) has emerged as a promising paradigm for integrating information from multiple views by leveraging the representation power of deep neural networks. However, most existing DMVC methods primarily focus on modeling pairwise relationships between samples, while neglecting higher-order structural dependencies among multiple samples, which may hinder further improvements in clustering performance. To address this limitation, we propose a hypergraph neural network (HGNN)-driven multi-view clustering framework, termed **H**ypergraph-enhanced c**O**ntrastive learning with hy**PE**r-Laplacian regula**R**ization (**HOPER**), a novel model that jointly captures high-order correlations and preserves local manifold structures across views. Specifically, we first construct view-specific hypergraph structures and employ the HGNN to learn node representations, thereby capturing high-order relationships among samples. Furthermore, we design a hypergraph-driven dual contrastive learning mechanism that integrates inter-view contrastive learning with intra-hyperedge contrastive learning, promoting cross-view consistency while maintaining discriminability within hyperedges. Finally, a hyper-Laplacian manifold regularization is introduced to preserve the local geometric structure within each view, thereby enhancing the structural fidelity and discriminative power of the learned representations. Extensive experiments on diverse datasets demonstrate the effectiveness of our approach.

1 Introduction

Multi-view data, collected from diverse sources or extracted via multiple feature extractors, contain both consensus and complementary information. As such data become increasingly prevalent in real-world applications, multi-view learning has emerged as a fundamental paradigm in machine learning for enhancing downstream performance by leveraging cross-view correlations [1–6]. Among its various tasks, multi-view clustering (MVC) plays a pivotal role in unsupervised learning by partitioning samples into meaningful groups without label supervision, thereby facilitating effective data analysis and organization [7–10] .

With the advance of deep representation learning, a variety of deep multi-view clustering (DMVC) methods have emerged, which can be broadly categorized into two main paradigms: representation

*Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

 https://doi.org/10.52202/085713-1826

learning-based and graph learning-based approaches [11]. The former typically leverages self-supervised learning frameworks to extract informative and discriminative latent representations directly from raw input features [12–14]. For instance, Wu et al. [15] employed view-specific deep autoencoders to extract embedded features and applied self-weighted contrastive fusion to learn robust global features. Cui et al. [16] enhanced information consistency across views and reduced redundancy by maximizing the lower bound of sufficient representation. Although deep representation learning methods have achieved significant success, they often struggle to explicitly model the complex relationships between samples, particularly those arising from intricate data structures. To address this limitation, deep graph learning methods have gained attention by using a shared graph neural network (GNN) encoder and projection head to represent each view as a graph, enabling unified cross-view representation learning in a common latent space [17–19]. For example, Xia et al. [20] employed graph convolutional networks (GCNs) to learn modality-specific representations, and introduced contrastive losses to encourage discriminative and clustering-friendly alignment across modalities. Similarly, Du et al. [21] utilized multiple GCNs with shared weights to extract view-specific representations, and incorporated a clustering embedding layer to jointly optimize representation learning and clustering performance. In addition, Dong et al. [22] performed contrastive learning at the graph structure level under the guidance of a consensus graph, thereby capturing the underlying structural information of the data.

Despite the performance improvements achieved by deep multi-view clustering (DMVC) methods through modeling pairwise relationships among samples, several critical limitations remain to be addressed. First, most existing DMVC methods primarily focus on modeling pairwise correlations between samples, while overlooking high-order and complex interactions among data points. This simplification significantly limits their ability to capture rich structural priors that are critical for effective clustering. Second, many contrastive learning-based DMVC approaches suffer from the false negative problem, where semantically similar samples from different views are incorrectly treated as negatives. This misidentification undermines the discriminative power of the learned representations. Third, these methods often lack explicit constraints to preserve high-order local geometric structures in the latent space, which are essential for maintaining topological consistency and enhancing the robustness of clustering.

To address the limitations of existing deep multi-view clustering methods, we propose a hypergraph-enhanced contrastive learning approach with hyper-Laplacian regularization. Specifically, for each view, we construct a hypergraph based on the sample features. This hypergraph structure, along with the corresponding features, is then input into a hypergraph neural network with shared weights to learn view-specific node representations, capturing high-order correlations among the data. Subsequently, a hypergraph-induced dual contrastive learning mechanism is employed to regularize the node representations: the inter-view contrastive loss regularization enhances consistency across views, while the intra-view contrastive loss regularization helps mitigate the false negative issue, promoting more discriminative representation learning. Finally, we introduce a hyper-Laplacian manifold regularization term to preserve the view-specific high-order local geometric structure, further enhancing the discriminative power of the node representations. In summary, the contributions of the proposed framework are as follows:

- We introduce a hypergraph neural network (HGNN)-based representation learning framework for multi-view clustering, which captures high-order data correlations by leveraging a hypergraph structure and a shared-weight HGNN.
- We propose a hypergraph-enhanced dual contrastive learning mechanism, which consists of an inter-view contrastive loss to reinforce consistency across different views and a intra-hyperedge contrastive loss to enhance the discriminability of individual samples within each hyperedge.
- We incorporate hyper-Laplacian manifold regularization to preserve view-specific higher-order local geometric structures, thereby further enhancing the robustness and effectiveness of representation learning.
- Experimental results demonstrate the superiority of our approach, highlighting its effectiveness in capturing high-order correlations and improving clustering performance compared to existing methods.

2 Related work

2.1 Hypergraph neural network

A hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a generalized graph structure capable of modeling high-order relationships, where $\mathcal{V}$ denotes the set of vertices and $\mathcal{E}$ denotes the set of hyperedges. The hyperedges can be represented using an incidence matrix $\mathbf{H} \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{E}|}$, where each entry $h(v,e)$ equals 1 if vertex v belongs to hyperedge e, and 0 otherwise. Each hyperedge is associated with a non-negative weight, which can be encoded in a diagonal matrix $\mathbf{W} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$ with diagonal elements $w(e)$. The degree of a vertex and a hyperedge are defined as:

$$d(v) = \sum_{e\in\mathcal{E}} w(e)h(v,e), \quad d(e) = \sum_{v\in\mathcal{V}} h(v,e), \tag{1}$$

which can be organized into diagonal matrices $\mathbf{D}_v$ and $\mathbf{D}_e$, respectively.

Based on these definitions, the normalized hypergraph Laplacian matrix is formulated as:

$$\mathbf{L}_H = \mathbf{D}_v - \mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top \tag{2}$$

Owing to the expressive power of hypergraph structures in modeling high-order relationships, hypergraph neural networks (HGNNs) have attracted increasing attention in recent years. The general HGNN framework typically consists of two key components: hypergraph construction and hypergraph convolution [23]. According to whether hyperedge construction is performed explicitly or implicitly, hypergraph construction methods can be broadly categorized into four types: distance-based, representation-based, attribute-based, and network-based approaches [24]. Hypergraph convolutions can be further classified into spatial and spectral methods, depending on how the convolutional operators are defined [25]. Recently, hypergraph-based models have demonstrated impressive performance in clustering tasks. For instance, [26] and [27] leverage hypergraphs to effectively capture attribute information and high-order structural dependencies, achieving strong results in single-view node classification and clustering. However, most existing approaches are tailored to single-view settings, and relatively limited attention has been paid to exploring hypergraph learning in the context of multi-view clustering.

2.2 Multi-view contrastive clustering

In the domain of multi-view clustering, contrastive learning demonstrated strong potential in promoting cross-view alignment [28–33]. For instance, Trosten et al. [34] enhanced multi-view clustering by aligning representations at the instance level. Similarly, Xu et al. [35] introduced contrastive learning at multiple feature levels, including high-level semantic consistency and cluster-level consistency across views. Their approach effectively mitigated the negative impact of view-specific inconsistencies in low-level features, resulting in more stable representation alignment. Pan and Kang [36] proposed a method that first obtains a smooth node representation through graph filtering and then learns a robust consensus graph guided by graph contrastive loss for clustering. Xu et al. [37] introduced a self-supervised framework that utilizes global pseudo-labels to help different views collaboratively learn discriminative features, improving both the consistency and robustness of multi-view clustering. In addition, Zhang et al. [38] leveraged contrastive learning to align generated and real views by applying diffusion and reverse denoising processes to intra-view data, enabling the model to capture distributional consistency and improve clustering performance.

3 Method

In this section, we introduce the proposed HOPER model. We first provide an overview of the HOPER framework, then describe the latent representation learning based on hypergraph neural networks, the hypergraph-enhanced dual contrastive learning mechanism, and the hypergraph Laplacian regularization. Lastly, we present the unified loss function that integrates all these components.

3.1 Framework outline

Given a multi-view dataset $\{\mathbf{X}^v \in \mathbb{R}^{N\times D_v}\}_{v=1}^M$ with N samples and M views, where $\mathbf{X}^v$ denotes the raw input of the v-th view, the goal of multi-view clustering is to partition the data into K clusters.

 https://doi.org/10.52202/085713-1826

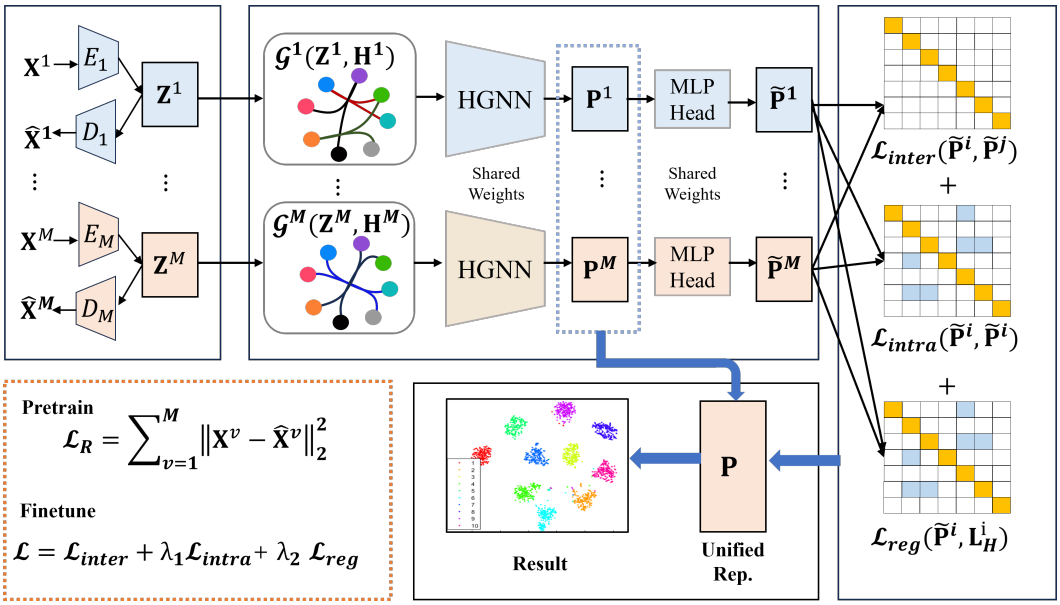


Figure 1: Illustration of the HOPER model. Given a multi-view dataset $\{\mathbf{X}^v\}_{v=1}^M$, we first learn latent representations and build view-specific hypergraphs. These are fed into a shared HGNN to generate embeddings. Then, a dual contrastive learning mechanism enhances consistency and discrimination. Finally, hyper-Laplacian regularization preserves the local geometric structure.

The overall framework of the proposed HOPER model is illustrated in Figure 1. Specifically, for each view, we first employ an autoencoder to project the raw features into a latent space and then construct a corresponding hypergraph structure based on the latent representations. The sample features and the corresponding hypergraph structure are then fed into a hypergraph neural network (HGNN) with shared parameters to learn node representations. Subsequently, a hypergraph-enhanced dual contrastive learning strategy is applied to the embeddings to improve both cross-view consistency and sample-level discriminability. Finally, a hyper-Laplacian regularization term is introduced to automatically preserve the local high-order geometric structure of each view in the embedding space.

3.2 Hypergraph construction and HGNN-based representation learning

To effectively model high-order correlations among data instances, we first construct view-specific hypergraph structures. Considering that raw multi-view data may contain noise and redundant information, directly constructing hyperedges based on the original features can degrade the quality of the hypergraphs. To address this issue, instead of relying on raw input features, we adopt an autoencoder to project the original data $\mathbf{X}^v$ into a compact latent space. Specifically, the autoencoder comprises an encoder $\mathbf{Z}^v = f(\mathbf{X}^v; \theta^v)$ and a decoder $\hat{\mathbf{X}}^v = g(\mathbf{Z}^v; \phi^v)$, where θ^v and ϕ^v are trainable parameters. The model is trained to minimize the reconstruction loss, as defined in Eq. (3).

$$\mathcal{L}_R = \sum_{v=1}^{M} \|\mathbf{X}^v - \hat{\mathbf{X}}^v\|_2^2 \tag{3}$$

After training, compact latent embeddings $\mathbf{Z}^v$ are obtained from the encoder module. Subsequently, multi-view hypergraph structures are constructed by applying a k-nearest neighbors strategy to these latent embeddings, where k is a predefined parameter specifying the number of neighbors to select. Specifically, for each view, we compute the pairwise Euclidean distances between all samples to capture local geometric relationships. For each sample, we identify its top-k nearest neighbors and construct a hyperedge connecting the sample with its neighbors. This process generates a collection of hyperedges that encode high-order relationships beyond simple pairwise connections, represented by the incidence matrix $\mathbf{H}^v$. Finally, the hypergraph for the v-th view is obtained, denoted as $\mathcal{G}^v = (\mathcal{V}^v, \mathcal{E}^v)$, with feature matrix $\mathbf{Z}^v$, and incidence matrix $\mathbf{H}^v$.

https://doi.org/10.52202/085713-1826

To further enhance the discriminative capability of latent representations by leveraging high-order relationships among samples, we incorporate a hypergraph neural network (HGNN) into our framework to learn more expressive node embeddings. Specifically, we adopt a two-layer HGNN module composed of stacked hypergraph convolution layers for node representation learning. Following [23], a hyperedge convolutional layer is formulated as:

$$\mathbf{P}_l^v = \sigma(\mathbf{D}_v^{-1}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{P}_{l-1}^v\Theta_l), \tag{4}$$

where σ denotes the nonlinear activation function, Θ_l denotes the learnable parameter, $\mathbf{P}_l^v$ is the node representation at l layer, $\mathbf{P}_0^v = \mathbf{Z}^v$. Notably, the HGNN network of different views share parameters in order to better align the learned representations across views.

3.3 Hypergraph-enhanced dual contrastive learning

Multi-view data provide complementary perspectives of the same underlying object, and typically exhibit semantic consistency across views. This cross-view consistency can be effectively exploited through contrastive learning. However, existing multi-view contrastive learning methods focus on aligning paired samples across views, treating them as positives while largely ignoring intra-view structural characteristics. Specifically, in each view, samples connected by the same hyperedge in the underlying hypergraph often interact through a message passing scheme, which can induce an over-smoothing effect—making originally distinguishable samples overly similar in representation space. While such samples may share local semantic information, they should still remain discriminative to preserve meaningful structural distinctions. The failure to account for this nuance limits the expressiveness of the learned representations, ultimately hindering clustering performance.

To address this limitation, we propose a hypergraph-enhanced dual contrastive learning mechanism, which jointly performs inter-view and intra-view contrastive learning. Specifically, inter-view contrastive learning aligns representations across views by encouraging cross-view consistency, while the intra-view contrastive learning leverages hyperedge structural information to enhance the discriminability among samples within each view.

Inter-view contrastive loss: This loss function is typically designed to compare node representations learned from different views, aiming to maintain consistency across views. Let $\tilde{\mathbf{P}}^v$ denote the node features, i.e., the output of the projection head for node representations. Specifically, representations of the same instance across different views are treated as positive pairs, while all other instances are regarded as negative pairs. The inter-view contrastive loss $\mathcal{L}_{inter}$ is defined as:

$$\mathcal{L}_{inter}(\tilde{\mathbf{p}}_i^v) = -\log \frac{\exp\left(s(\tilde{\mathbf{p}}_i^v, \tilde{\mathbf{p}}_i^m)/\tau\right)}{\sum_{j=1}^{N}\sum_{m\neq v}\exp\left(s(\tilde{\mathbf{p}}_i^v, \tilde{\mathbf{p}}_j^m)/\tau\right)}, \tag{5}$$

where τ denotes the temperature parameter, $s(\cdot)$ denotes the similarity function which is implemented as cosine similarity.

Intra-view contrastive loss: Due to the strong connectivity introduced by hyperedges, the message passing mechanism in HGNN may lead to an over-smoothing issue, where node representations tend to become indistinguishable—especially for nodes connected by the same hyperedge—thus undermining their discriminative power [39]. To mitigate this, we introduce an intra-view contrastive learning strategy that enhances the distinctiveness of individual samples. Specifically, each node is treated as a positive pair with itself, while other nodes within the same hyperedge are considered negative samples. The intra-view contrastive loss $\mathcal{L}_{intra}$ is formulated as

$$\mathcal{L}_{\text{intra}}(\tilde{\mathbf{p}}_i^v) = -\log \frac{\exp\left(s(\tilde{\mathbf{p}}_i^v, \tilde{\mathbf{p}}_i^v)/\tau\right)}{\sum_{j\in\mathcal{N}_i^v}\exp\left(s(\tilde{\mathbf{p}}_i^v, \tilde{\mathbf{p}}_j^v)/\tau\right)}, \tag{6}$$

where $\mathcal{N}_i^v$ denotes the set of neighbors which are in the same hyperedges as node i in v-th view. In our experiments, the temperature parameters in Equation (5) and Equation (6) are shared optimized.

By integrating the aforementioned dual contrastive learning strategy, the model not only enforces global alignment across views but also enhances local discriminability within each view, thereby yielding more robust and semantically meaningful representations.

3.4 Hyper-Laplacian regularization

To better preserve the intrinsic local structure of the data, we incorporate a hypergraph Laplacian regularization term into our model. By leveraging the hypergraph Laplacian, this regularizer embeds

 https://doi.org/10.52202/085713-1826

Algorithm 1 Hypergraph-enhanced Multi-view Representation Learning

Input: Multi-view raw features $\{\mathbf{X}^v\}_{v=1}^M$, number of clusters K
Output: Cluster assignments via k-means on unified representations

Pretraining:
for each view $v = 1$ to M **do**
    Pretrain the view-specific autoencoder by optimizing Eq.(3)
end for
Feature Encoding:
Obtain node features $\{\mathbf{Z}^v\}_{v=1}^M$ from encoder networks
Hypergraph Construction:
Construct view-specific hypergraphs via k-NN strategy on $\mathbf{Z}^v$
Joint Optimization:
for $t = 1$ to $T_{\max}$ **do**
    Update the shared hypergraph encoder and projection head by optimizing Eq.(8)
end for
Fusion:
Compute unified representations using Eq.(9)
Clustering:
Apply k-means to obtain final clustering results

the manifold assumption into the learning process—that is, if multiple data points are close in the intrinsic geometry of the data space, their representations in the latent space should also be similar. This mechanism promotes smoothness of the learned representations over the hypergraph, thereby enhancing the model's ability to capture higher-order structural information and improving its generalization performance on downstream tasks. The mathematical expression is as follows:

$$\mathcal{L}_{reg} = \sum_{v=1}^{M} tr(\tilde{\mathbf{P}}^v \mathbf{L}_h^v (\tilde{\mathbf{P}}^v)^\top), \tag{7}$$

where $\mathbf{L}_h^v$ is hyper-graph Laplacian matrix of the v-th view. By introducing this hyper-Laplacian regularization, our model gains the ability to exploit richer structural information beyond simple pairwise constraints, leading to improved representation quality and better generalization performance across downstream tasks.

3.5 The overall loss function

By integrating the intra-view and inter-view contrastive loss, and the hypergraph Laplacian regularization, the overall loss function of the proposed HOPER model is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{inter} + \lambda_1 \mathcal{L}_{intra} + \lambda_2 \mathcal{L}_{reg} \tag{8}$$

Overall, the optimization of our method consists of two stages: initialization and fine-tuning. During the initialization stage, we first pretrain a view-specific autoencoder for each view by minimizing the reconstruction loss, as defined in Eq.(3). Once pretraining is completed, we construct a hypergraph structure for each view based on the learned latent representations. In the fine-tuning stage, the entire network is trained by optimizing the objective function given in Eq.(8). After optimization, the learned node embeddings from all views are concatenated to form the unified representation $\mathbf{P}$:

$$\mathbf{P} = [\mathbf{P}^1, \mathbf{P}^2, ..., \mathbf{P}^M] \in \mathbb{R}^{N \times \sum_{v=1}^{M} d_v}, \tag{9}$$

where d_v denotes the dimension of $\mathbf{P}^v$ Finally, the unified representation $\mathbf{P}$ is used as input to the k-means algorithm to produce the clustering results. The whole learning process is summarized in Algorithm 1.

3.6 Comparison with Previous Studies

Although recent studies have explored Hypergraph Contrastive Learning (HCL) [40–44] and Hyper-Laplacian Regularization [45–51], our approach differs fundamentally from these existing methods.

First, regarding application scenarios, existing HCL methods [40–42] generally treat multi-view data as augmented variants of a single view, whereas our framework defines multi-view data as heterogeneous feature sets extracted from the same instance, capturing genuinely distinct and complementary perspectives. Second, the learning objectives differ significantly: [40–42] primarily target node classification, while [43] focuses on recommendation tasks. In contrast, our model is specifically designed for unsupervised multi-view clustering, aiming to discover shared semantics and cross-view consistency without label supervision. Although [44] incorporates HCL into multi-view clustering, the two methods diverge fundamentally in terms of model design motivation, the perspective for exploiting multi-view data, and the contrastive learning mechanism. Furthermore, while some studies have incorporated Hyper-Laplacian Regularization into multi-view clustering [45–52], most adopt shallow learning paradigms based on matrix/tensor factorization or graph self-representation, lacking the representational capacity of deep neural models. In contrast, our approach leverages a deep learning framework that integrates Hyper-Laplacian Regularization with hypergraph neural networks, enabling richer feature representations and improved clustering quality.

4 Experiments

This section presents a comprehensive empirical evaluation of the proposed HOPER model, encompassing experimental settings, performance comparisons, parameter sensitivity analysis, feature visualizations, and ablation studies.

4.1 Experimental settings

Datasets: To comprehensively evaluate the effectiveness of the proposed HOPER framework, we conduct experiments on six publicly available multi-view datasets. Their statistical details are shown in Table 1. BBCsport contains 544 samples with 2 views, corresponding to five categories. Synthetic3d is a 3-D dataset containing 600 samples with 3 views. WebKB contains 203 web pages of 4 categories. Each web page is described from 3 views. COIL-20 consists of 1440 samples with 3 views which belongs to 20 categories. Handwritten contains 2000 samples of handwritten digits from 0-9, where each sample is described from 6 views. Hdigit is a digit dataset from MNIST Handwritten Digits and USPS Handwritten Digits which consists of 10000 samples described by 2 views.

Table 1: Statistics of six benchmark datasets.

Dataset	#Samples	#Views	#Clusters
BBCSport	544	2	5
Synthetic3d	600	3	3
WebKB	1051	2	2
COIL-20	1440	3	20
Handwritten	2000	6	10
Hdigit	10000	2	10

Evaluation metrics: For evaluation, three widely-used metrics, including the clustering Accuracy (ACC), Normalized Mutual Information (NMI), Adjusted Rand Index (ARI) are calculated to comprehensively compare the performance of various methods.

Comparison methods: We compare our framework with the following state-of-the-art DMVC algorithms to investigate the effectiveness of our framework, i.e., MFLVC (2022) [35], CVCL (2023) [53], DealMVC (2023) [54], SEM (2023) [55], DIVIDE (2024) [56], MAGA (2024) [33].

Implementation details: The proposed framework consists of two main modules: initialization and fine-tuning. In the initialization module, we utilize a four-layer autoencoder to obtain the latent embeddings. The number of nearest neighbors of hyperedge construction is tuned over $\{5, 10, 15, 20, 25, 30\}$ on different datasets. In the fine-tuning module, we optimize the hyperparameters λ_1 and λ_2. Based on empirical observations, we perform a grid search over the values $\{0.0001, 0.001, 0.01, 0.1, 1, 5, 10\}$ to select the optimal values for both hyperparameters on multiple datasets. The best performance is obtained under the combination of $\lambda_1 = 5$ and $\lambda_2 = 0.0001$. These specific values were then used for all experiments reported in this paper. The training process consists of 2000 epochs for the autoencoder initialization phase and 200 epochs for the fine-tuning phase. We employ a cosine learning rate decay to adjust the learning rate dynamically. All experiments are conducted using the PyTorch framework on an NVIDIA GeForce RTX 3090 GPU.

 https://doi.org/10.52202/085713-1826

Table 2: Clustering performance across benchmark datasets.

Dataset	Metric	MFLVC	CVCL	DealMVC	SEM	DIVIDE	MAGA	HOPER
BBCSport	ACC	0.7224	0.6211	0.8070	0.6085	0.4467	0.5533	**0.9504**
	NMI	0.5344	0.3645	0.6559	0.3666	0.1507	0.2808	**0.8509**
	ARI	0.5874	0.3137	0.6005	0.2918	0.1091	0.2286	**0.8677**
Synthetic3d	ACC	0.9500	0.9546	0.8033	0.9467	0.9497	0.9600	**0.9717**
	NMI	0.8218	0.8158	0.5797	0.8095	0.8083	0.8388	**0.8775**
	ARI	0.8582	0.8689	0.5667	0.8494	0.8553	0.8846	**0.9165**
WebKB	ACC	0.7174	0.7181	0.6974	0.9486	0.8325	0.9010	**0.9829**
	NMI	0.2986	0.2832	0.2474	0.6809	0.1791	0.4201	**0.8416**
	ARI	0.1885	0.7810	0.1552	0.7897	0.2985	0.5986	**0.9249**
COIL-20	ACC	0.3875	0.6882	0.2299	0.7403	0.6486	0.4569	**0.8285**
	NMI	0.5450	0.7851	0.4783	0.8296	0.7608	0.6317	**0.8880**
	ARI	0.3093	0.7007	0.1735	0.6588	0.5466	0.4151	**0.7869**
Handwritten	ACC	0.8990	0.9194	0.8220	0.7645	0.8708	0.9415	**0.9685**
	NMI	0.8259	0.8878	0.8163	0.7285	0.8277	0.9083	**0.9317**
	ARI	0.7939	0.8473	0.7367	0.6317	0.8086	0.8854	**0.9308**
Hdigit	ACC	0.9882	0.9505	**0.9980**	0.9864	0.9646	0.9954	0.9961
	NMI	0.9841	0.8900	**0.9934**	0.9606	0.9103	0.9856	0.9876
	ARI	0.9882	0.8930	**0.9980**	0.9701	0.9229	0.9898	0.9961

4.2 Performance comparison

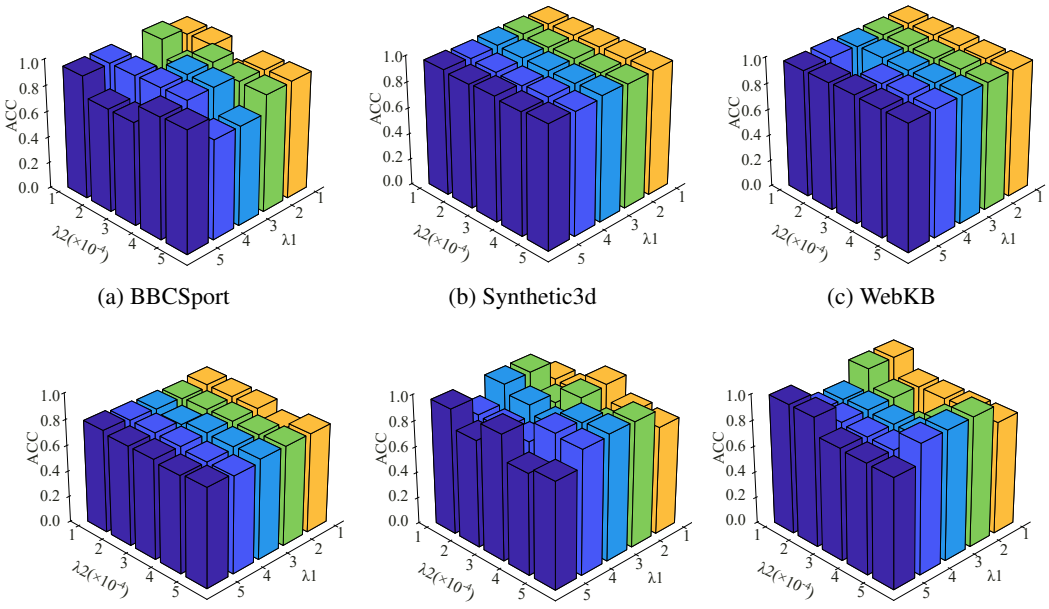
Table 2 summarizes the performance of all methods, with the best and second-best results highlighted in **bold** and underlined, respectively. The experimental comparison demonstrates that the HOPER model achieves highly competitive clustering performance compared to existing baseline methods across all evaluation metrics. For instance, on the COIL-20 dataset, HOPER improves upon the second-best SEM model by approximately 8.8%, 5.4%, and 12.8% in terms of ACC, NMI, and ARI, respectively. These results validate that our approach, leveraging hypergraph-enhanced contrastive learning and hypergraph Laplacian manifold regularization, enhances the discriminability of latent representations, thereby improving clustering performance. Moreover, although the HOPER model does not achieve the best result on the Hdigit dataset, its accuracy is nearly 100% and only 0.19% lower than that of DealMVC, further validating the effectiveness of our approach.

4.3 Parameter sensitivity analysis

In the HOPER model, two hyperparameters, λ_1 and λ_2, are introduced to balance the contrastive learning objective and the hyper-Laplacian regularization term. To evaluate the model's sensitivity, we conduct a grid search over $\lambda_1 \in \{1, 2, 3, 4, 5\}$ and $\lambda_2 \in \{0.0001, 0.0002, 0.0003, 0.0004, 0.0005\}$. The clustering results (ACC) on various datasets are reported in Figure 2. On Synthetic3D, WebKB, and COIL-20 datasets, HOPER demonstrates consistent and stable clustering performance across the range of parameter values. Some fluctuations are observed on BBCSport, Handwritten, and Hdigit datasets, which may be attributed to the trade-off between the regularization terms affecting the discriminative quality of the learned representations. Overall, despite minor variations, HOPER maintains robust and competitive clustering results within the evaluated hyperparameter ranges.

4.4 Visualization of representation evolution

In this section, we present a qualitative analysis of the proposed HOPER model on the BBCSport dataset by visualizing the learned representations at different stages of the framework. As shown in Figure 3, subfigures (a) and (b) illustrate the raw data from two views, $\mathbf{X}^1$ and $\mathbf{X}^2$, while (c) and (d) show the corresponding latent embeddings $\mathbf{Z}^1$ and $\mathbf{Z}^2$ obtained via the autoencoder module. Compared with the raw inputs, the latent embeddings reveal a clearer clustering structure, suggesting that the autoencoder effectively denoises the data and captures more compact representations. Subfigures (e) and (f) visualize the node representations $\mathbf{P}^1$ and $\mathbf{P}^2$ refined by the hypergraph contrastive learning module and the hypergraph Laplacian regularization. These representations exhibit more discriminative and well-separated clusters than their latent counterparts, highlighting the effectiveness of the proposed hypergraph-based enhancements in capturing high-order and local relational structures

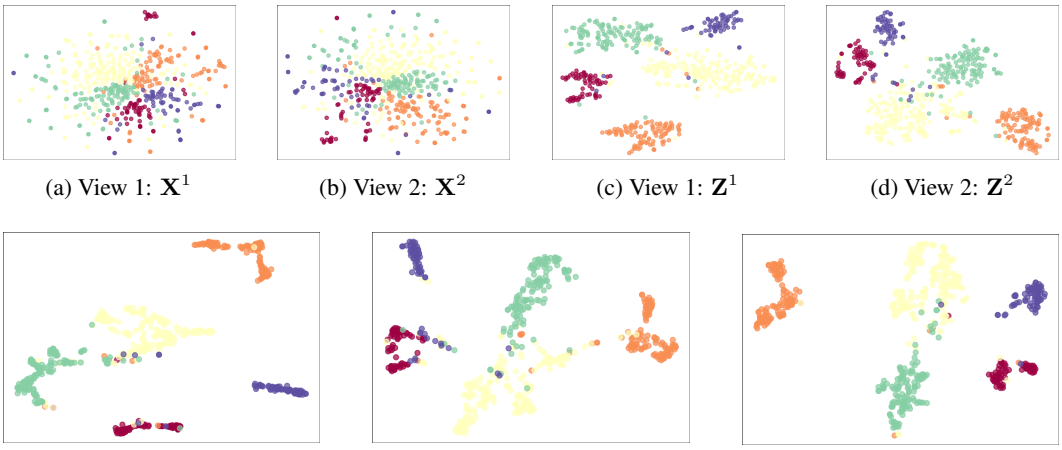


(a) BBCSport (b) Synthetic3d (c) WebKB

(d) COIL-20 (e) Handwritten (f) Hdigit

Figure 2: Hyperparameter sensitivity analysis of the HOPER model on multiple datasets.

within each view. Finally, subfigure (g) depicts the unified representation aggregated from multiple views. It exhibits the most compact and separable cluster structures among all stages, demonstrating the ability of HOPER to effectively exploit cross-view complementary information and enhance the overall representation quality, thereby leading to improved clustering performance.



(a) View 1: $\mathbf{X}^1$ (b) View 2: $\mathbf{X}^2$ (c) View 1: $\mathbf{Z}^1$ (d) View 2: $\mathbf{Z}^2$

(e) View 1: $\mathbf{P}^1$. (f) View 2: $\mathbf{P}^2$. (g) Unified $\mathbf{P}$.

Figure 3: Visualization of the learned representations at different stages of the HOPER.

4.5 Convergence analysis

In this subsection, we demonstrate the convergence of HOPER across six datasets by reporting the loss values. As shown in Figure 4, the horizontal axis represents the training epochs and the vertical axis denotes the loss value. It can be observed that the loss drops rapidly during the first 100 epochs and then gradually decreases until convergence.

https://doi.org/10.52202/085713-1826

4.6 Ablation studies

As defined in Eq. 8, the overall loss function of HOPER consists of three components. To investigate the individual contribution of each term, we conduct ablation studies by systematically removing each component and retraining the model under the same experimental settings. Table 3 reports the clustering performance on multiple datasets under different loss configurations, where a checkmark indicates that the corresponding loss term is included. The experimental results demonstrate that removing any single component from the complete HOPER model consistently leads to performance degradation across all datasets, with some cases exhibiting significant drops. This highlights that HOPER effectively integrates inter-view contrastive learning, intra-view contrastive learning, and hypergraph Laplacian manifold regularization, which together promote cross-view consistency in sample representations while preserving inter-sample discriminability, ultimately enhancing clustering performance.

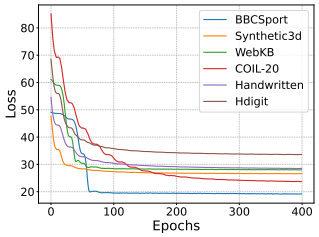


Figure 4: Convergence analysis of the HOPER model on multiple datasets.

Table 3: Ablation study on the effects of individual loss components in HOPER.

Components			BBCSport			Synthetic3d			WebKB		
$\mathcal{L}_{intra}$	$\mathcal{L}_{inter}$	$\mathcal{L}_{reg}$	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
✓	✓		0.8897	0.7785	0.8076	0.9650	0.8624	0.8990	0.9686	0.7520	0.8659
✓		✓	0.9449	0.8371	0.8529	0.9500	0.8117	0.9504	0.9724	0.7744	0.8815
	✓	✓	0.5147	0.4724	0.2553	0.9667	0.8607	0.9027	0.9629	0.7151	0.8387
✓	✓	✓	**0.9504**	**0.8509**	**0.8677**	**0.9717**	**0.8775**	**0.9165**	**0.9829**	**0.8416**	**0.9249**
Components			COIL-20			Handwritten			Hdigit		
$\mathcal{L}_{intra}$	$\mathcal{L}_{inter}$	$\mathcal{L}_{reg}$	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
✓	✓		0.7910	0.8740	0.7597	0.8285	0.8573	0.7857	0.7022	0.8661	0.7173
✓		✓	0.7549	0.8781	0.7358	0.8695	0.8529	0.7915	0.8471	0.9117	0.8406
	✓	✓	0.7847	0.8687	0.7496	0.8240	0.8814	0.8410	0.9268	0.8439	0.6537
✓	✓	✓	**0.8285**	**0.8880**	**0.7869**	**0.9685**	**0.9317**	**0.9308**	**0.9961**	**0.9876**	**0.9961**

5 Conclusion

This paper proposes a novel multi-view clustering framework, HOPER, which integrates hypergraph-enhanced contrastive learning with hypergraph Laplacian regularization to learn discriminative feature representations. Specifically, HOPER captures high-order relationships among samples through hypergraph construction and hypergraph neural networks. To further improve representation quality, a hypergraph-driven dual contrastive learning mechanism is introduced, comprising inter-view contrastive learning and intra-hyperedge contrastive learning, which promotes cross-view consistency while preserving discriminability within hyperedge. In addition, hypergraph Laplacian regularization is employed to preserve high-order local structural information. Extensive experiments on six benchmark datasets demonstrate that HOPER achieves highly competitive performance, validating its effectiveness for discriminative representation learning.

6 Limitations

A potential limitation of our method is its relatively higher computational complexity compared to traditional graph-based approaches. This is because hypergraphs introduce hyperedges that connect multiple nodes to capture high-order relationships, leading to more complex operations involving the incidence and Laplacian matrices than in conventional graphs where edges link only two nodes.

Acknowledgements

This work was supported by the Natural Science Foundation of Hebei Province (No. F2025205006), the Science Foundation of Hebei Normal University (No. L2025B38) and the Backbone Talent Program (Program for Returned Overseas Scholars) (No. A2025016).

References

[1] Changqing Zhang, Huazhu Fu, Jing Wang, Wen Li, Xiaochun Cao, and Qinghua Hu. Tensorized multi-view subspace representation learning. *International Journal of Computer Vision*, 128:2344–2361, 2020.

[2] Jiaqi Jin, Siwei Wang, Zhibin Dong, Xinwang Liu, and En Zhu. Deep incomplete multi-view clustering with cross-view partial sample and prototype alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11600–11609, 2023.

[3] Zhibin Gu, Zhendong Li, and Songhe Feng. Topology-driven multi-view clustering via tensorial refined sigmoid rank minimization. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, page 920–931, 2024.

[4] Cai Xu, Jiajun Si, Ziyu Guan, Wei Zhao, Yue Wu, and Xiyue Gao. Reliable conflictive multi-view learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16129–16137, 2024.

[5] Suyuan Liu, Ke Liang, Zhibin Dong, Siwei Wang, Xihong Yang, Sihang Zhou, En Zhu, and Xinwang Liu. Learn from view correlation: An anchor enhancement strategy for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26151–26161, 2024.

[6] Weixuan Liang, Chang Tang, Xinwang Liu, Yong Liu, Jiyuan Liu, En Zhu, and Kunlun He. On the consistency and large-scale extension of multiple kernel clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(10):6935–6947, 2024.

[7] Xiaoqiang Yan, Zhixiang Jin, Fengshou Han, and Yangdong Ye. Differentiable Information Bottleneck for Deterministic Multi-View Clustering . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27425–27434, 2024.

[8] Emanuele Palumbo, Laura Manduchi, Sonia Laguna, Daphné Chopard, and Julia E Vogt. Deep generative clustering with multimodal diffusion variational autoencoders. In *The Twelfth International Conference on Learning Representations*, pages 1–9, 2024.

[9] Haonan Huang, Guoxu Zhou, Yanghang Zheng, Yuning Qiu, Andong Wang, and Qibin Zhao. Adversarially robust deep multi-view clustering: A novel attack and defense framework. In *Forty-first International Conference on Machine Learning*, pages 1–9, 2024.

[10] Weixuan Liang, Xinwang Liu, Ke Liang, Jiyuan Liu, and En Zhu. COKE: Core kernel for more efficient approximation of kernel weights in multiple kernel clustering. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pages 37257–37280, 2025.

[11] Uno Fang, Man Li, Jianxin Li, Longxiang Gao, Tao Jia, and Yanchun Zhang. A comprehensive survey on multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12350–12368, 2023.

[12] Fangfei Lin, Bing Bai, Yiwen Guo, Hao Chen, Yazhou Ren, and Zenglin Xu. Mhcn: A hyperbolic neural network model for multi-view hierarchical clustering. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16479–16489, 2023.

[13] Weitian Huang, Sirui Yang, and Hongmin Cai. Generalized information-theoretic multi-view clustering. In *Thirty-seventh Conference on Neural Information Processing Systems*, pages 1–13, 2023.

[14] Guanzhou Ke, Bo Wang, Xiaoli Wang, and Shengfeng He. Rethinking multi-view representation learning via distilled disentangling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26774–26783, June 2024.

[15] Song Wu, Yan Zheng, Yazhou Ren, Jing He, Xiaorong Pu, Shudong Huang, Zhifeng Hao, and Lifang He. Self-weighted contrastive fusion for deep multi-view clustering. *IEEE Transactions on Multimedia*, 2024.

 https://doi.org/10.52202/085713-1826

[16] Chenhang Cui, Yazhou Ren, Jingyu Pu, Jiawei Li, Xiaorong Pu, Tianyi Wu, Yutao Shi, and Lifang He. A novel approach for effective multi-view clustering with information-theoretic perspective. *Advances in neural information processing systems*, 36:44847–44859, 2023.

[17] Zhe Xue, Junping Du, Hai Zhou, Zhongchao Guan, Yunfei Long, Yu Zang, and Meiyu Liang. Robust diversified graph contrastive network for incomplete multi-view clustering. In *Proceedings of the 30th ACM international conference on multimedia*, pages 3936–3944, 2022.

[18] Xuejiao Yu, Yi Jiang, Guoqing Chao, and Dianhui Chu. Deep contrastive multi-view subspace clustering with representation and cluster interactive learning. *IEEE Transactions on Knowledge and Data Engineering*, 37(1):188–199, 2025.

[19] Yazhou Ren, Jingyu Pu, Chenhang Cui, Yan Zheng, Xinyue Chen, Xiaorong Pu, and Lifang He. Dynamic weighted graph fusion for deep multi-view clustering. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 4842–4850, 2024.

[20] Wei Xia, Tianxiu Wang, Quanxue Gao, Ming Yang, and Xinbo Gao. Graph embedding contrastive multi-modal representation learning for clustering. *IEEE Transactions on Image Processing*, 32:1170–1183, 2023.

[21] Guowang Du, Lihua Zhou, Zhongxue Li, Lizhen Wang, and Kevin Lü. Neighbor-aware deep multi-view clustering via graph convolutional network. *Information Fusion*, 93:330–343, 2023.

[22] Zhibin Dong, Jiaqi Jin, Yuyang Xiao, Siwei Wang, Xinzhong Zhu, Xinwang Liu, and En Zhu. Iterative deep structural graph contrast clustering for multiview raw data. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[23] Yue Gao, Yifan Feng, Shuyi Ji, and Rongrong Ji. Hgnn+: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3181–3199, 2022.

[24] Yue Gao, Zizhao Zhang, Haojie Lin, Xibin Zhao, Shaoyi Du, and Changqing Zou. Hypergraph learning: Methods and practices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2548–2566, 2020.

[25] Alessia Antelmi, Gennaro Cordasco, Mirko Polato, Vittorio Scarano, Carmine Spagnuolo, and Dingqi Yang. A survey on hypergraph representation learning. *ACM Computing Surveys*, 56(1):1–38, 2023.

[26] Yumeng Song, Yu Gu, Tianyi Li, Jianzhong Qi, Zhenghao Liu, Christian S Jensen, and Ge Yu. Chgnn: a semi-supervised contrastive hypergraph learning network. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[27] Dongjin Lee and Kijung Shin. I'm me, we're us, and i'm us: Tri-directional contrastive learning on hypergraphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 8456–8464, 2023.

[28] Zhizhong Huang, Jie Chen, Junping Zhang, and Hongming Shan. Learning representation for clustering via prototype scattering and positive sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6):7509–7524, 2023.

[29] Chenhang Cui, Yazhou Ren, Jingyu Pu, Jiawei Li, Xiaorong Pu, Tianyi Wu, Yutao Shi, and Lifang He. A novel approach for effective multi-view clustering with information-theoretic perspective. In *Advances in Neural Information Processing Systems*, volume 36, pages 44847–44859, 2023.

[30] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):857–876, 2023.

[31] Mouxing Yang, Yunfan Li, Peng Hu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. Robust multi-view clustering with incomplete information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):1055–1069, 2023.

[32] Yijie Lin, Yuanbiao Gou, Xiaotian Liu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. Dual contrastive prediction for incomplete multi-view representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4447–4461, 2023.

[33] Jintang Bian, Xiaohua Xie, Jian-Huang Lai, and Feiping Nie. Multi-view contrastive clustering via integrating graph aggregation and confidence enhancement. *Information Fusion*, 108:102393, 2024.

[34] Daniel J Trosten, Sigurd Lokse, Robert Jenssen, and Michael Kampffmeyer. Reconsidering representation alignment for multi-view clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1255–1265, 2021.

[35] Jie Xu, Huayi Tang, Yazhou Ren, Liang Peng, Xiaofeng Zhu, and Lifang He. Multi-level feature learning for contrastive multi-view clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16051–16060, 2022.

[36] Erlin Pan and Zhao Kang. Multi-view contrastive graph clustering. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, pages 2148 – 2159, 2021.

[37] Jie Xu, Yazhou Ren, Huayi Tang, Zhimeng Yang, Lili Pan, Yang Yang, Xiaorong Pu, Philip S. Yu, and Lifang He. Self-supervised discriminative feature learning for deep multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):7470–7482, 2023.

[38] Yuanyang Zhang, Yijie Lin, Weiqing Yan, Li Yao, Xinhang Wan, Guangyuan Li, Chao Zhang, Guanzhou Ke, and Jie Xu. Incomplete multi-view clustering via diffusion contrastive generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 22650–22658, 2025.

[39] Derun Cai, Moxian Song, Chenxi Sun, Baofeng Zhang, Shenda Hong, and Hongyan Li. Hypergraph structure learning for hypergraph neural networks. In *IJCAI*, pages 1923–1929, 2022.

[40] Jongsoo Lee and Dong-Kyu Chae. Multi-view mixed attention for contrastive learning on hypergraphs. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 2543–2547, 2024.

[41] Rong Qian, ZongFang Lv, and YuChen Zhou. Multi-view hypergraph adaptive contrastive learning. In *Advances in Knowledge Discovery and Data Mining*, pages 293–305, 2025.

[42] Jianian Zhu, Weixin Zeng, Junfeng Zhang, Jiuyang Tang, and Xiang Zhao. Cross-view graph contrastive learning with hypergraph. *Information Fusion*, 99:101867, 2023.

[43] Sen Zhao, Wei Wei, Xian-Ling Mao, Shuai Zhu, Minghui Yang, Zujie Wen, Dangyang Chen, and Feida Zhu. Multi-view hypergraph contrastive policy learning for conversational recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 654–664, 2023.

[44] Liang Chen, Zhe Xue, Yawen Li, Meiyu Liang, Yan Wang, Anton Van Den Hengel, and Yuankai Qi. Medusa: A multi-scale high-order contrastive dual-diffusion approach for multi-view clustering. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10295–10304, 2025.

[45] Qilun Luo, Ming Yang, Wen Li, and Mingqing Xiao. Hyper-laplacian regularized multi-view clustering with exclusive l21 regularization and tensor log-determinant minimization approach. *ACM Transactions on Intelligent Systems and Technology*, 14:1–29, 03 2023.

[46] Zixiao Yu, Lele Fu, Yongyong Chen, Zhiling Cai, and Guoqing Chao. Hyper-laplacian regularized concept factorization in low-rank tensor space for multi-view clustering. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 9(2):1728–1742, 2025.

[47] Gui-Fu Lu, Qin-Ru Yu, Yong Wang, and Ganyi Tang. Hyper-laplacian regularized multi-view subspace clustering with low-rank tensor constraint. *Neural Networks*, 125:214–223, 2020.

 https://doi.org/10.52202/085713-1826

[48] Shuqin Wang, Yongyong Chen, Linna Zhang, Yigang Cen, and Viacheslav Voronin. Hyper-laplacian regularized nonconvex low-rank representation for multi-view subspace clustering. *IEEE Transactions on Signal and Information Processing over Networks*, 8:376–388, 2022.

[49] Yuan Xie, Wensheng Zhang, Yanyun Qu, Longquan Dai, and Dacheng Tao. Hyper-laplacian regularized multilinear multiview self-representations for clustering and semisupervised learning. *IEEE Transactions on Cybernetics*, 50(2):572–586, 2020.

[50] Peng Song, Shixuan Zhou, Jinshuai Mu, Meng Duan, Yanwei Yu, and Wenming Zheng. Clean affinity matrix induced hyper-laplacian regularization for unsupervised multi-view feature selection. *Information Sciences*, 682:121276, 2024.

[51] Xiao Yu, Hui Liu, Yan Zhang, Yuan Gao, and Caiming Zhang. Robust multi-view clustering with hyper-laplacian regularization. *Information Sciences*, 694:121718, 2025.

[52] Zhibin Gu and Songhe Feng. From dictionary to tensor: A scalable multi-view subspace clustering framework with triple information enhancement. In *Advances in Neural Information Processing Systems*, volume 37, pages 103545–103573, 2024.

[53] Jie Chen, Hua Mao, Wai Lok Woo, and Xi Peng. Deep multiview clustering by contrasting cluster assignments. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16752–16761, 2023.

[54] Xihong Yang, Jin Jiaqi, Siwei Wang, Ke Liang, Yue Liu, Yi Wen, Suyuan Liu, Sihang Zhou, Xinwang Liu, and En Zhu. Dealmvc: Dual contrastive calibration for multi-view clustering. In *Proceedings of the 31st ACM international conference on multimedia*, pages 337–346, 2023.

[55] Jie Xu, Shuo Chen, Yazhou Ren, Xiaoshuang Shi, Hengtao Shen, Gang Niu, and Xiaofeng Zhu. Self-weighted contrastive learning among multiple views for mitigating representation degeneration. *Advances in neural information processing systems*, 36:1119–1131, 2023.

[56] Yiding Lu, Yijie Lin, Mouxing Yang, Dezhong Peng, Peng Hu, and Xi Peng. Decoupled contrastive multi-view clustering with high-order random walks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 14193–14201, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly summarize the key contributions and align well with the paper's overall scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper acknowledges a limitation of the proposed approach, namely the higher complexity of hypergraph modeling compared to traditional graphs.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

https://doi.org/10.52202/085713-1826

Answer: [NA]

Justification: The paper does not involve any theoretical results or formal proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides sufficient details on the experimental setup, datasets, evaluation metrics, and implementation settings to support reproducibility of the main results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is provided in the appendix, and all datasets used are publicly available, ensuring reproducibility of the results.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The paper details data splits, hyperparameters, and optimizer settings to ensure clarity and reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We reported the experimental results but did not include error bars or other information regarding statistical significance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

https://doi.org/10.52202/085713-1826

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provides information on the type of computing devices used for the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research fully complies with all aspects of the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This research is foundational and is not related to specific applications or practical deployments.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The authors have cited the original papers that produced the code packages or datasets used.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

https://doi.org/10.52202/085713-1826

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The code is provided in the supplementary material.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This study does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this study does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

https://doi.org/10.52202/085713-1826