

Towards Learnable Anchor for Deep Multi-View Clustering

Bocheng Wang¹, Chusheng Zeng¹, Mulin Chen^{1*}, Xuelong Li²

¹School of Artificial Intelligence, OPTics and ElectroNics (iOPEN),
Northwestern Polytechnical University, China

²Institute of Artificial Intelligence (TeleAI), China Telecom, China

wangbocheng@mail.nwpu.edu.cn, zcs@mail.nwpu.edu.cn, chenmulin@nwpu.edu.cn, xuelong_li@ieee.org

Abstract

Deep multi-view clustering incorporating graph learning has presented tremendous potential. Most methods encounter costly square time consumption w.r.t. data size. Theoretically, anchor-based graph learning can alleviate this limitation, but related deep models mainly rely on manual discretization approaches to select anchors, which indicates that 1) the anchors are fixed during model training and 2) they may deviate from the true cluster distribution. Consequently, the unreliable anchors may corrupt clustering results. In this paper, we propose the Deep Multi-view Anchor Clustering (DMAC) model that performs clustering in linear time. Concretely, the initial anchors are intervened by the positive-incentive noise sampled from Gaussian distribution, such that they can be optimized with a newly designed anchor learning loss, which promotes a clear relationship between samples and anchors. Afterwards, anchor graph convolution is devised to model the cluster structure formed by the anchors, and the mutual information maximization loss is built to provide cross-view clustering guidance. In this way, the learned anchors can better represent clusters. With the optimal anchors, the full sample graph is calculated to derive a discriminative embedding for clustering. Extensive experiments on several datasets demonstrate the superior performance and efficiency of DMAC compared to state-of-the-art competitors.

Introduction

In real-world data acquisition, a sample is often recorded from different views or sources (Cui et al. 2024), constituting multi-view data. For instance, an image can be distilled by multiple feature descriptors (e.g., color, shape, and spatial relation), and a document can be interpreted from multiple perspectives (e.g., topic distribution, word sequence, and word frequency). To adapt to the actual application environment, many types of unsupervised Multi-View Clustering (MVC) have emerged. Among them, deep MVC incorporates the enormous advantages of neural networks on representation learning, and has presented outstanding performance in real-world scenarios.

In general, deep MVC sets up a separate auto-encoder for each view, and then the learned multiple view-specific embeddings are fused into a consensus to infer clusters. To

mitigate the inter-view conflict and perceive multi-view consistency, many representation alignment strategies are leveraged, such as contrastive learning (Xu et al. 2022b; Hu et al. 2023), and label distribution alignment (Cheng et al. 2021; Xu et al. 2022a; Liu et al. 2024). Overall, most deep MVC models emphasize the discriminability of the output embedding to improve clustering.

Recently, some works integrate the graph learning theory into deep MVC, which considers the structure relationship between samples while learning the discriminative embedding (Yang et al. 2023; Yan et al. 2023; Wang et al. 2024). These novel models capture the topological structure by constructing a data similarity graph, and thus optimizing representation learning based on graph data mining techniques, such as graph convolution network and structure preservation scheme (Wang et al. 2023b; Xiao et al. 2023). These methods usually require computing the edge weights between any two samples to build a full sample graph, which leads to costly time complexity $O(n^2)$ where n is the amount of samples. One theoretical solution to this problem is anchor graph learning that can accelerate the training to linear time w.r.t. n (Dong et al. 2023; Cui et al. 2023). The performance of anchor graph learning and final clustering heavily depends on the anchor quality. However, most deep models employ non-differentiable manual procedures (e.g., random selection and k -means) to determine the final anchors, which means the anchors are not learnable. Ideally, each sample should be represented by a certain anchor, and the anchors should reflect the cluster distribution. If the selected anchors cannot represent the original samples and cluster centroids well, the clustering result may be adversely affected.

To remedy the problems, we establish the Deep Multi-view Anchor Clustering (DMAC) model. As shown in Fig. 1, the pipeline of DMAC can be partitioned into three main items. Firstly, the positive-incentive perturbation is injected into the initial anchors to yield a learnable mechanism for the anchors. Then, anchor graph convolution is used to produce the cluster indicator of anchors for each view, and the cross-view agreement is obtained by the proposed mutual information maximization loss. Finally, the learned anchors are leveraged to reveal the structural relationship between samples to impel a discriminative embedding. The key contributions of this paper are listed as follows.

- Anchor graph learning is incorporated into the deep

*Corresponding author.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

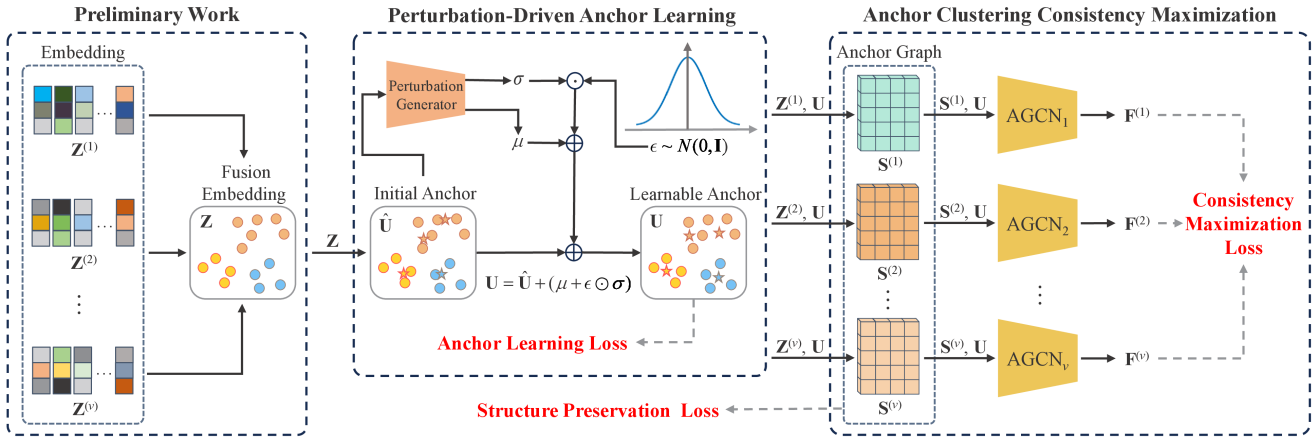


Figure 1: Pipeline of DMAC. Note that the encoders are omitted. For the a -th view, $\mathbf{Z}^{(a)}$ is the data embedding, $\mathbf{A}^{(a)}$ is the anchor graph, AGCN_a is the corresponding anchor graph convolution network, and $\mathbf{F}^{(a)}$ is the anchor clustering distribution that records the probability of an anchor belonging to each cluster. \mathbf{Z} is the shared fusion embedding among views. \mathbf{U} represents the learnable anchors injected with the perturbation. The overall framework is updated by minimizing Eq. (16). The final result is gained by performing k -means on the convergent \mathbf{Z} .

multi-view clustering framework. Compared to most competitors, the proposed model can capture structural information with learnable anchors in a high-efficiency linear time to ameliorate embedding learning.

- A perturbation-driven mechanism is proposed to improve the anchor quality adaptively during model training. By infusing beneficial perturbations, the learned anchors are adjusted to foster a distinct similarity relationship with the samples, thus representing the original data more comprehensively.
- An anchor graph convolution network is devised to infer the anchor clustering distribution of each view. The mutual information among multiple distributions is maximized to explore the consistent anchor clustering distribution, so as to produce clustering-oriented anchors.

Related Work

Deep Multi-View Clustering

Benefiting from the powerful representation learning ability of neural networks, deep MVC has achieved dominant clustering performance in practical applications (Fang et al. 2023b). Generally, deep MVC leverages auto-encoders to learn a series of view-specific embeddings, and then executes representation fusion to infer the consensus cluster indications. Many models are proposed to promote the clustering-friendly deep embedding. DAMC (Li et al. 2019) introduces the adversarial training mechanism to improve the discriminability of the learned embedding. DEMVC (Xu et al. 2021) pursues a consistent cluster structure between views to exploit the cross-view complementary information. In (Trosten et al. 2021), the importance of representation alignment for MVC is analyzed theoretically, and a contrastive learning based deep MVC method is established. MFLVC (Xu et al. 2022b) performs the feature- and cluster-level contrastive learning simultaneously to inhibit the ad-

verse effects of view-private information. CVCL (Chen et al. 2023) advocates unifying the cluster assignment by multi-view contrastive learning.

The abovementioned representatives have presented good clustering capacity in experiments. However, most models focus on the intrinsic features while neglecting the structure relationship among samples that is essential to detect the clustering distribution (Huang et al. 2019; Wang et al. 2019; Chen, Wang, and Li 2024). Recently, some scholars introduce the graph learning theory into deep MVC to mine the structural information explicitly. CMGEC (Wang et al. 2021) employs a graph fusion network to integrate multiple structural graphs into a consensus. DFP-GNN (Xiao et al. 2023) parallels graph neural networks with auto-encoders to learn the feature- and structure-level embedding simultaneously. DealMVC (Yang et al. 2023) constructs the data similarity graph to guide clustering-oriented contrastive learning. GCFAggMVC (Yan et al. 2023) utilizes the similarity relationship among embeddings to ameliorate representation fusion. SURER (Wang et al. 2024) concatenates multiple view-specific graphs into a heterogeneous graph to explore the complementary relationship among views via the heterogeneous graph neural network. Those deep MVC models that incorporate the graph structure information have presented enormous development potential. Nevertheless, existing methods usually require computing the full sample graph, which encounters a squared complexity $O(n^2)$ where n is the number of samples. In this paper, we plan to introduce the anchor theory to achieve deep MVC in linear time.

Anchor-Based Multi-View Clustering

Anchors, also known as landmarks, are widely used in graph-based and sub-space MVC (Yang et al. 2024). An anchor is the representative of local data (Chen and Cai 2011). By learning an anchor graph that records the adjacency relation between samples and anchors, the similarity among

samples can be estimated approximately to derive the clusters. SFMC (Li et al. 2020) fuses multiple anchor graphs into a consensus bipartite graph with the rank constraint. MSGL (Kang et al. 2021) uses the data self-expression property to realize the adaptive anchor selection and graph optimization. FDAGF (Zhang et al. 2023) allows multiple anchor combinations as inputs to improve the flexibility and generalization ability. E²OMVC (Wang et al. 2023a) calculates the spectral embeddings of anchor graphs and fuses them into the final cluster representation. CAMVC (Zhang et al. 2024) utilizes the estimated labels to optimize cluster-wise anchors.

Since the number of anchors is much smaller than the sample size, anchor-based MVC has outstanding efficiency. Nevertheless, most relevant algorithms are limited by shallow graph learning that the anchor graph is calculated via the original features directly. There is little exploration (Dong et al. 2023; Cui et al. 2023) of anchor-based deep MVC, which neglects the optimization of anchor quality with model training. Inspired by the positive-incentive noise (Li 2022), we plan to generate positive noise perturbation to guide high-quality anchor learning, and design the anchor graph convolution module to capture cross-view anchor clustering consistency.

Methodology

In this section, the proposed DMAC is elaborated. DMAC generates positive perturbation to ameliorate the anchors shared among views, and utilizes anchor graph convolution to extract the cluster distribution of anchors.

Notations: matrices and vectors are expressed as uppercase and lowercase letters, respectively. For a matrix \mathbf{X} , both \mathbf{X}_i and \mathbf{x}_i mean the i -th row. $\|\mathbf{x}_i\|_1$, $\|\mathbf{x}_i\|_2$, and $\|\mathbf{X}\|_F$ denote ℓ_1 , ℓ_2 , and Frobenius norm, respectively.

Preliminary Work

Denote $\{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(v)}\}$ as the multi-view data with n samples, v views, and c clusters. DMAC follows the mainstream deep MVC framework consisting of view-specific embedding learning and embedding fusion. In this part, the framework is introduced briefly to pave the subsequent innovative modules.

Specifically, the unshared encoder is used to learn view-specific deep embedding

$$\mathbf{Z}^{(a)} = \text{Encoder}_a(\mathbf{X}^{(a)}). \quad (1)$$

Based on the resultant embeddings $\{\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \dots, \mathbf{Z}^{(v)}\}$, the simple but effective average weighting (Wang et al. 2024) is employed to calculate the fusion embedding

$$\mathbf{Z} = \frac{1}{v} \sum_i \mathbf{Z}^{(i)}, \quad (2)$$

which is fed into a clustering algorithm to gain the labels.

Very lately, graph learning theory is introduced into the above framework, aiming to extract the topological structure of sample space to enhance embedding learning. Most related models encounter an expensive time complexity $O(n^2)$. Differently, we integrate anchor graph learning into deep MVC to reduce the complexity to linear time.

Perturbation-Driven Anchor Learning

Anchor graph learning requires estimating multiple anchors in advance. Existing models usually use the manual setting strategy to select anchors, which inhibits the learnability of anchors. Therefore, we construct a generator to learn the perturbation to adjust anchors, and design the anchor learning loss to obtain the positive perturbation that improves anchor quality.

Perturbation Generation Network. Denoting $\hat{\mathbf{U}}$ as the m initial shared anchors obtained by performing k -means on \mathbf{Z} , the generation network produces learnable perturbation ε to inject $\hat{\mathbf{U}}$. In this way, the anchors can be optimized by updating ε through backpropagation.

To begin with, the initial perturbation ε is sampled from the standard multivariate Gaussian distribution

$$\varepsilon \sim N(0, \mathbf{I}), \quad (3)$$

where \mathbf{I} is an identity matrix, and ε has the same dimensionality as $\hat{\mathbf{U}}$.

Then, we use a pseudo-siamese perceptron to simulate the mean μ and deviation σ of the perturbation, which are formulated as

$$\mu = \text{MLP}_\mu(\hat{\mathbf{U}}), \quad \sigma = \text{MLP}_\sigma(\hat{\mathbf{U}}). \quad (4)$$

Consequently, the perturbation ε is updated as

$$\varepsilon = \mu + \sigma \odot \varepsilon, \quad (5)$$

where \odot refers to the Hadamard product. Eq. (5) meets the reparameterization trick (Kingma and Welling 2014) that optimizes ε by backpropagation. Finally, the anchor matrix is updated as

$$\mathbf{U} = \hat{\mathbf{U}} + \varepsilon. \quad (6)$$

Since ε is optimized gradually during training, the learnability of the anchor matrix \mathbf{U} is achieved.

Anchor Learning Loss. Given the learnable anchors, the relationship between samples and anchors is explored to improve the quality of anchors.

Intuitively, the ideal anchors can be considered as sub-centroids, which are appropriately dispersed, and each sample is strongly associated with a corresponding anchor. Therefore, the similarity of the sample and anchors is crucial to evaluate the anchor quality. Denoting $q_i^{(a)} \in \mathbb{R}^{1 \times m}$ as the similarity vector between the sample $\mathbf{Z}_i^{(a)}$ and m anchors \mathbf{U} , $q_{ij}^{(a)}$ is computed as

$$q_{ij}^{(a)} = \frac{(1 + \|\mathbf{Z}_i^{(a)} - \mathbf{U}_j\|_2^2)^{-1}}{\sum_k (1 + \|\mathbf{Z}_i^{(a)} - \mathbf{U}_k\|_2^2)^{-1}}. \quad (7)$$

Then, we introduce the positive-incentive noise theory to pave the anchor learning loss. In other words, the perturbation ε is regarded as the potential positive noise, which satisfies the definition in (Li 2022).

Definition 1. Mathematically, the positive noise ε_π satisfies

$$E(\mathcal{T}|\varepsilon_\pi) < E(\mathcal{T}), \quad (8)$$

where \mathcal{T} represents a specific downstream task, and $E(\cdot)$ computes the information entropy.

According to Definition 1, the positive-incentive noise aims to reduce the uncertainty of a specific downstream task. Considering that the anchors are representatives of the original data, the uncertainty of anchor selection task is mainly reflected by the distribution of $q_i^{(a)}$. To be specific, for a sample $\mathbf{Z}_i^{(a)}$, if all values in $q_i^{(a)}$ are very close, the relationship between the sample and anchors is ambiguous, which indicates a high uncertainty. That is to say, we need to learn a extremely unbalanced $q_i^{(a)}$, wherein one element is much larger than the others.

Therefore, for the a -th view, the task entropy $E(\mathcal{T}|\epsilon_\pi)$ in Eq. (8) can be quantified as

$$\mathcal{L}_{AL}^{(a)} = \frac{1}{n} \sum_i^n E\left(q_i^{(a)}\right) = -\frac{1}{n} \sum_i^n \sum_j^m q_{ij}^{(a)} \log\left(q_{ij}^{(a)}\right). \quad (9)$$

By minimizing Eq. (9), each sample tends to hold a highly correlated anchor, such that the anchors can represent the data distribution well. In the following part, we aim to make the anchors aligned with the cluster distribution.

Anchor Clustering Consistency Maximization

To learn clustering-oriented anchors, we devise anchor graph convolution to infer the anchor clustering distribution of each view, and introduce mutual information to capture cross-view anchor clustering consistency, so as to provide training guidance for anchor learning.

Anchor Graph Learning. Since the proposed anchor graph convolution network requires graph data as input, we first present anchor graph learning.

Anchor graph records the structural dependence between samples and anchors. For the a -th view, an ideal anchor graph $\mathbf{S}^{(a)}$ respects the following assumption.

Assumption 1. *The edge $s_{ij}^{(a)}$ is negatively correlated with the distance between the corresponding nodes $\mathbf{Z}_i^{(a)}$ and \mathbf{U}_j .*

Assumption 1 also reflects the fundamental clustering scenario, that points with small distances are more likely to be within the same cluster. Hence, for the a -th view, the anchor graph learning problem can be expressed as

$$\min_{\mathbf{S}^{(a)}} \sum_i^n \sum_j^m \|\mathbf{Z}_i^{(a)} - \mathbf{U}_j\|_2^2 s_{ij}^{(a)} + \gamma \|\mathbf{S}^{(a)}\|_F^2, \quad (10)$$

$$s.t. \forall i \|\mathbf{s}_i^{(a)}\|_1 = 1, 0 \leq s_i^{(a)} \leq 1,$$

where $\mathbf{s}_i^{(a)}$ is the i -th row of the anchor graph $\mathbf{S}^{(a)} \in \mathbb{R}^{n \times m}$, and the second term evades that each sample only connects with the nearest anchor. The constraint that the sum of each row in $\mathbf{S}^{(a)}$ is 1 aims to approach Assumption 1.

According to (Nie, Zhu, and Li 2017), problem (10) can be solved with an efficient closed-form solution, which also evades the selection of parameter γ .

Anchor Graph Convolution. Based on the view-specific anchor graph $\mathbf{S}^{(a)}$ and shared anchors \mathbf{U} , we introduce graph convolution (Kipf and Welling 2016) to calculate the anchor clustering distribution, leading to Anchor Graph Convolution Network (AGCN).

For the a -th view, the row in anchor clustering distribution $\mathbf{F}^{(a)} \in \mathbb{R}^{m \times c}$ records the probability of an anchor belonging to each cluster. Specifically, the forward propagation of l -th hidden layer in AGCN _{a} is

$$\mathbf{F}^{(a)(l+1)} = \varphi\left(\mathbf{D}_{\mathbf{S}^{(a)}}^{-1} (\mathbf{S}^{(a)})^T \mathbf{S}^{(a)} \mathbf{F}^{(a)(l)} \mathbf{W}^{(l)}\right), \quad (11)$$

where $\mathbf{D}_{\mathbf{S}^{(a)}} \in \mathbb{R}^{m \times m}$ is the diagonal degree matrix of $\mathbf{S}^{(a)}$ that the j -th item is $\sum_i^n s_{ij}^{(a)}$, $(\mathbf{D}_{\mathbf{S}^{(a)}}^{-1} (\mathbf{S}^{(a)})^T \mathbf{S}^{(a)}) \in \mathbb{R}^{m \times m}$ is the symmetric and doubly stochastic anchor similarity graph (Zhang et al. 2022) that conforms to the criterion of GCN, $\mathbf{W}^{(l)}$ is the parameter matrix, and $\varphi(\cdot)$ is a certain activation function. Note that $\mathbf{F}^{(a)(0)} = \mathbf{U}$.

The neuron number of the last layer in AGCN is the cluster number c , and the softmax function is used for activation. Multiple AGCNs do not share parameters to learn the anchor cluster structure for each view.

Consistency Maximization Loss. We adopt Mutual Information (MI) to measure the difference among $\{\mathbf{F}^{(1)}, \mathbf{F}^{(2)}, \dots, \mathbf{F}^{(v)}\}$. Compared to the widespread KL divergence, MI satisfies symmetry to increase computational efficiency. The large MI, the more similar the two distributions is. Based on MI, the consistency maximization loss for the a -th view is

$$\mathcal{L}_{CM}^{(a)} = -\frac{1}{m} \sum_{b=a+1}^v \sum_i^m MI\left(\mathbf{F}_i^{(a)}, \mathbf{F}_i^{(b)}\right), \quad (12)$$

where

$$MI\left(\mathbf{F}_i^{(a)}, \mathbf{F}_i^{(b)}\right) = \sum_{x \sim \mathbf{F}_i^{(a)}} \sum_{y \sim \mathbf{F}_i^{(b)}} p(x, y) \log\left(\frac{p(x, y)}{p(x)p(y)}\right). \quad (13)$$

By minimizing Eq. (12), multiple anchor clustering distributions are aligned to relieve the adverse effects of view conflict and view-private information. Unlike existing deep MVC, the module achieves cross-view consensus from the perspective of anchors rather than samples, which is conducive to propelling an explicit cluster structure of anchors.

Structure Preservation via Anchor Graph

In this part, the structural graph of samples is calculated via the learned anchor graph. On this basis, the structure preservation loss is developed to promote a discriminative multi-view embedding.

Based on the anchor graph $\mathbf{S}^{(a)}$, the full sample graph of the a -view can be measured with

$$\mathbf{G}^{(a)} = \mathbf{S}^{(a)} \mathbf{D}_{\mathbf{S}^{(a)}}^{-1} (\mathbf{S}^{(a)})^T. \quad (14)$$

Obviously, the resultant $\mathbf{G}^{(a)} \in \mathbb{R}^{n \times n}$ is a symmetric and doubly stochastic graph (Zhang et al. 2023).

With the a -th full sample graph $\mathbf{G}^{(a)}$, the structure preservation loss is

$$\mathcal{L}_{SP}^{(a)} = \sum_{i, j}^n \|\mathbf{Z}_i - \mathbf{Z}_j\|_2^2 g_{ij}^{(a)}, \quad (15)$$

which can be replaced with Eq. (19) to accelerate the matrix multiplication. There is a concern that Eq. (15) may trivially cause all samples to be mapped to the same embedding, which is called representation collapse. In Theorem 1,

we deduce that the anchor learning loss can be seen as a regularization term to penalize the trivial solution.

By minimizing Eq. (15), the learned fusion embedding \mathbf{Z} is actuated to reserve the internal data structure of each view, that is, the samples in the same class remain compact. The complementary structural information across views is mined to ameliorate a discriminative fusion embedding for clustering performance improvement.

Joint Loss and Optimizer

Combining Eqs. (9), (12), and (15), the joint loss is

$$\mathcal{L} = \sum_a^v \left(\mathcal{L}_{AL}^{(a)} + \alpha \mathcal{L}_{CM}^{(a)} + \beta \mathcal{L}_{SP}^{(a)} \right), \quad (16)$$

where both α and β are the trade-off parameters.

The classical RMSprop optimizer (Zhang and Sennrich 2019) is adopted to train DMAC. The final result is obtained by performing k -means (Hartigan and Wong 1979) on the fusion representation \mathbf{Z} .

Discussion and Analysis

Theoretical Advantage of Anchor Learning Loss

In this part, we discuss that the proposed anchor learning loss shown in Eq. (9) is beneficial for relieving representation collapse. The anchor learning loss can be regarded as a regularization term of the structure preservation loss shown in Eq. (15) to boost a discriminative fusion embedding \mathbf{Z} .

Theorem 1. *Minimizing Eq. (9) is equivalent to penalizing the trivial solution (i.e., representation collapse) to Eq. (15).*

Proof. Without loss of generality, we develop the proof from the perspective of the a -th view. The converse-negative proposition corresponding to the theorem is

$$(\mathbf{Z} \rightarrow \mathbf{Z}^*) \Rightarrow \left(\mathcal{L}_{AL}^{(a)} \not\rightarrow 0 \right), \quad (17)$$

where \mathbf{Z}^* is the trivial solution to Eq. (15), that is, all rows in \mathbf{Z} tend to be the same.

Because $\mathbf{U} \sim \mathbf{Z}$ (i.e., \mathbf{U} is sampled from \mathbf{Z}), all anchors also tend to be the same when $\mathbf{Z} = \mathbf{Z}^*$. Hence, the probability distribution $q_i^{(a)}$ between sample $\mathbf{Z}_i^{(a)}$ and anchor matrix \mathbf{U} is very smooth, and then the anchor learning loss reaches the upper bound. The above deduction can be formulized as

$$\begin{aligned} (\mathbf{Z} \rightarrow \mathbf{Z}^*) &\Rightarrow (\mathbf{U} \rightarrow \mathbf{U}^*) \Rightarrow \left(\forall i \forall j q_{ij}^{(a)} \rightarrow \frac{1}{m} \right) \\ &\Rightarrow \left(\mathcal{L}_{AL}^{(a)} \rightarrow \log(m) \not\rightarrow 0 \right). \end{aligned} \quad (18)$$

The original proposition and converse-negative proposition possess the same truth and falsehood property. Proposition (17) is true, so the theorem is proven. The proof can be generalized to any view easily. \square

Linear Computation Complexity

DMAC is able to accomplish MVC in the linear time complexity $O(n)$. To avoid excessive symbol definition and improve readability, we only analyze the influence of sample size n and anchor number m .

Dataset	Samples	Views	Classes	Dimensions
Yale	165	3	15	4096, 3304, 6750
PIE	680	3	68	484, 256, 279
BBC	685	4	5	4659, 4633, 4665, 4684
NUS	2400	6	10	64, 144, 73, 128, 225, 500
CCV	6773	3	20	4000, 5000, 5000
ALOI	10800	4	100	77, 13, 64, 125

Table 1: Descriptions of real-world datasets.

In each forward propagation, the computation complexity of embedding learning and fusion is $O(n)$. Then, the initial anchor selection needs $O(nm)$ via k -means, and the generator requires $O(m)$ to output the perturbation matrix. Finally, considering that $\mathbf{D}_{\mathbf{S}^{(a)}}^{-1}$ is a diagonal matrix, the consumption of anchor graph convolution is $O(nm)$.

In each back propagation, the anchor learning loss needs $O(nm)$ to calculate the entropies of all rows in \mathbf{Q} . The structure preservation loss can be written as the trace form

$$\text{Tr} \left(\mathbf{Z}^T \mathbf{D}_{\mathbf{G}}^{(a)} \mathbf{Z} - \mathbf{Z}^T \mathbf{S}^{(a)} \mathbf{D}_{\mathbf{S}^{(a)}}^{-1} \left(\mathbf{S}^{(a)} \right)^T \mathbf{Z} \right), \quad (19)$$

where $\mathbf{D}_{\mathbf{G}}^{(a)} \in \mathbb{R}^{n \times n}$ is the degree matrix of $\mathbf{G}^{(a)}$ shown in Eq. (14). Since $\mathbf{G}^{(a)}$ is a doubly stochastic matrix, $\mathbf{D}_{\mathbf{G}}^{(a)} \in \mathbb{R}^{n \times n}$ is an identity matrix. The calculation consumption of structure preservation loss is also $O(nm)$ with Eq. (19). Finally, the consistency maximization loss needs $O(m)$.

In conclusion, the time complexity of DMAC is $O(nm)$. Normally, the quantity of anchors is much smaller than the sample size (i.e., $m \ll n$), so the average complexity of each iteration can be seen as $O(n)$.

Experiments

In this section, the proposed DMAC is compared with advanced competitors. The ablation analysis is also conducted.

Real-World Datasets

Six public real-world datasets that are widely used in clustering study are collected as benchmarks, including image-type Yale (Belhumeur, Hespanha, and Kriegman 1997), PIE (Gross et al. 2010) and ALOI (Houle et al. 2010), text-type BBC (Greene and Cunningham 2006) and NUS (Bryant and Ng 2015), and video-type CCV (Jiang et al. 2011). Each sample is preprocessed with the ℓ_2 norm normalization. Table 1 displays the basic information of each dataset.

Evaluation Metrics

Two widespread metrics are adopted to quantify the clustering result, including Accuracy (ACC) and Normalized Mutual Information (NMI). Both ACC and NMI are positively correlated with the clustering performance. The mathematical expression can be found in (Wang et al. 2020).

Comparison with Competitors

Competitors. Nine state-of-the-art methods are selected as competitors, including four shallow algorithms GMC

Method	Yale		PIE		BBC		NUS		CCV		ALOI		Avg	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
GMC	69.70	70.06	21.18	44.24	69.05	47.87	18.24	9.96	10.66	0.43	57.05	73.50	40.98	41.01
MSGL	40.61	47.32	15.74	46.08	46.28	23.15	15.25	5.27	12.42	7.11	15.81	39.66	24.35	28.10
LMVSC	57.58	58.10	<u>36.32</u>	63.33	66.42	53.92	20.67	8.58	18.29	14.09	<u>58.51</u>	76.37	42.97	45.73
UDBGL	53.33	58.76	24.26	52.72	72.85	50.94	24.08	13.11	25.57	20.83	52.44	61.02	42.09	42.90
CMGEC	36.36	42.60	14.77	45.33	<u>87.37</u>	<u>71.44</u>	24.87	10.83	22.21	23.67	56.42	72.89	40.33	44.46
DealMVC	<u>75.18</u>	<u>76.81</u>	23.82	52.14	64.75	41.20	20.04	9.49	13.95	6.87	17.50	44.50	35.87	38.50
GCFAggMVC	66.06	66.51	27.94	59.15	63.65	48.87	23.42	10.69	<u>35.43</u>	<u>32.92</u>	54.52	72.21	<u>45.17</u>	48.39
DFP-GNN	56.36	63.39	24.26	56.88	75.09	58.73	29.42	16.12	21.33	19.36	49.15	66.12	42.60	46.77
SURER	61.82	67.68	30.29	<u>64.16</u>	79.85	64.24	27.33	<u>16.17</u>	24.91	26.86	43.94	64.03	44.69	<u>50.52</u>
DMAC	78.18	78.06	43.24	68.16	88.61	74.49	<u>29.29</u>	16.20	36.18	33.17	60.35	<u>74.29</u>	55.98	57.40

Table 2: Clustering performance of ten methods on six datasets. Bold and underlined values mean the optimal and sub-optimal results respectively. The column termed avg displays the average ACC and NMI of each method.

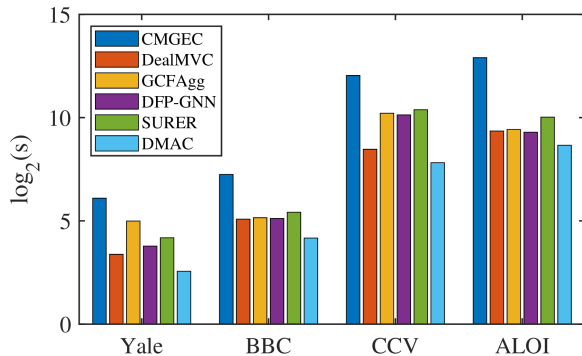


Figure 2: Runtime (s) of deep models on four datasets. Note that all records are converted by logarithmic base 2.

(Wang, Yang, and Liu 2019), MSGL (Kang et al. 2021), LMVSC (Kang et al. 2020) and UDBGL (Fang et al. 2023a), and five deep models CMGEC (Wang et al. 2021), DealMVC (Yang et al. 2023), GCFAggMVC (Yan et al. 2023), DFP-GNN (Xiao et al. 2023) and SURER (Wang et al. 2024). Among them, MSGL, LMVSC and UDBGL are anchor-based MVC, and all deep models incorporate the graph structure information.

Setups. The grid search is used to explore the optimal parameter setup for each algorithm. The parameter grid of competitors are set as the recommendations in the original article. For example, the parameter α of MSGL is selected from $\{0.001, 0.01, 0.1, 1, 10, 50\}$. For the proposed DMAC, the number of anchors m is set automatically according to (Nie, Wang, and Li 2019), i.e., the lower bound of $\sqrt{n \times c}$. The grid for both α and β is $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3\}$. The maximal iterations are 100.

The traditional methods are executed on Matlab 2019a with an Intel i9-12900HX CPU. All deep models are implemented via PyTorch, and trained with a NVIDIA RTX-3090 GPU. Each algorithm is repeated 10 times for objectivity.

Performance Comparison. Table 2 records the clustering performance of all algorithms. For ease of comparison, we

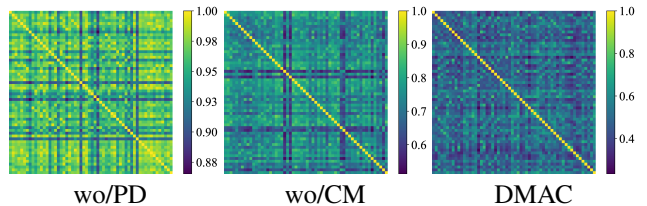


Figure 3: Anchor similarity matrix UU^T on BBC.

also calculate the average ACC and NMI of each algorithm on all datasets. In general, DMAC presents the best clustering ability. The success of DMAC proves the feasibility of applying anchor graph learning to deep MVC. DMAC learns high-quality anchors with the proposed perturbation-driven anchor learning scheme, and then mines the multi-view anchor clustering consistency via anchor graph convolution and mutual information maximization to further accelerate clustering-oriented anchors, so as to accurately reveal the structural graph for clustering improvement. According to the experimental results, we also summarize the following viewpoints. Firstly, compared with the traditional shallow methods, the deep models achieve better clustering scores, which indicates the enormous potential of neural networks on improving MVC. Secondly, the performance of GCFAggMVC and SURER are more prominent than other deep methods. SURER and GCFAggMVC leverage data similarity graphs to guide heterogeneous graph embedding learning and feature aggregation respectively, which reflects the positive effects of structural information on the two key steps of deep MVC, namely, view-specific representation learning and multi-view representation fusion.

Efficiency Comparison. Fig. 2 displays the clustering efficiency of deep models. It is observed that DMAC has the shortest runtime. Compared with advanced deep MVC models that incorporate graph structure learning, DMAC avoids inefficient full sample graph learning and graph convolution. The new anchor learning mechanism and anchor graph convolution network have linear time complexity theoretically,

Dataset	Metric	wo/PD	wo/CM	DMAC
Yale	ACC	66.02	72.73	78.18
	NMI	66.90	75.15	78.06
PIE	ACC	34.12	33.82	43.24
	NMI	62.94	62.97	68.16
BBC	ACC	87.15	88.47	88.61
	NMI	70.68	73.99	74.49
NUS	ACC	24.53	27.67	29.29
	NMI	14.29	15.35	16.20
CCV	ACC	32.28	34.26	36.18
	NMI	29.96	31.72	33.17
ALOI	ACC	44.45	54.79	60.35
	NMI	65.86	70.10	74.29

Table 3: Ablation results of main modules in DMAC. Bold values emphasize the optimal results.

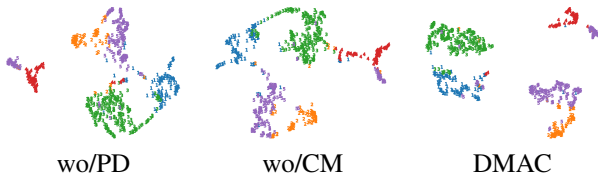


Figure 4: Visualization of fusion embedding \mathbf{Z} on BBC. Each point is drawn as its actual label value.

so as to speed up the training process.

Ablation Study and Visualization

In the ablation experiment, we design two variants based on the complete DMAC. Concretely, wo/PD removes the perturbation generation network and anchor learning loss, and wo/CM suspends the consistency maximization loss. Table 3 shows the ablation comparison. DMAC still maintains the best clustering performance, which proves the positive role of the proposed modules.

To intuitively display the effects of new modules, we visualize the similarity matrix of anchors (i.e., $\mathbf{U}\mathbf{U}^T$) in Fig. 3. It is exhibited that the anchor similarity matrix learned by DMAC has the most distinct diagonal, which means the anchors are relatively dispersive to adequately represent the sample clusters. The significant degradation of wo/PD compared to DMAC further proves the advantages of learnable anchors. The disparity between wo/CM and DMAC indicates that mining cross-view anchor clustering information is beneficial to guide clustering-oriented anchor learning for performance improvement.

In addition, we utilize UMAP (McInnes, Healy, and Melville 2018) to visualize the learned fusion embedding \mathbf{Z} on BBC. As shown in Fig. 4, the result derived by DMAC has a more obvious inter-class partition, which means a small inter-class similarity. The ablation comparison again indicates that, the new modules are conducive to learn high-quality and clustering-friendly anchors for accurate structure learning, so as to improve the discriminative ability of

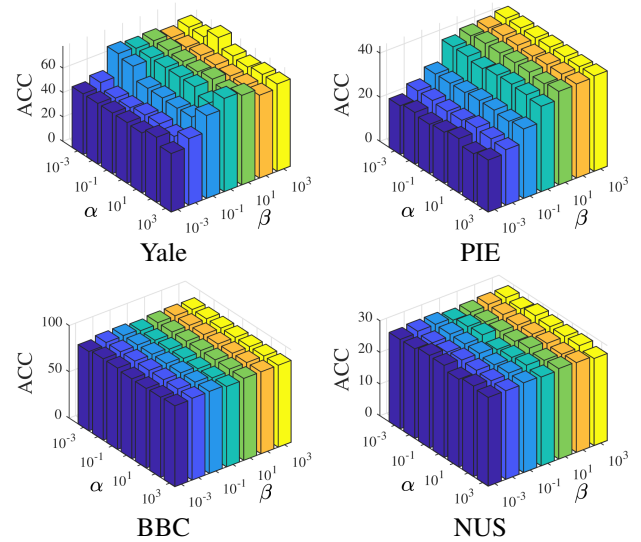


Figure 5: ACC of DMAC with different parameters α and β .

fusion embedding via the structure preservation loss. The advantage of DMAC over wo/PD reflects the practicability of Theorem 1, that is, the proposed anchor learning loss can suppress representation collapse.

Parameter Sensitivity

Finally, the sensitivity of DMAC to the trade-off parameters α and β is explored. Fig. 5 exhibits ACC of DMAC under the predefined parameter grid. It can be seen that the influence of β is more significant than that of α , because β is directly related to the fusion representation \mathbf{Z} that derives the final result. The preliminary observation suggests that the combination of large α and β is more likely to facilitate a good clustering result. Overall, the performance fluctuation is relatively smooth within an appropriate range.

Conclusion

In this paper, we propose an anchor-based deep multi-view clustering model termed DMAC. Different from traditional manual anchor selection ways, DMAC introduces a perturbation-driven anchor learning mechanism to make the anchors learnable. Specifically, inspired by the positive-incentive noise theory, a noise generation network is established to produce the perturbation adaptively, which is injected into anchors under the guidance of anchor learning loss. Besides, the anchor graph convolution module is designed to extract the cluster structure of anchors within each view, and then the multi-view anchor clustering consistency can be perceived with mutual information maximization. In this manner, DMAC is able to optimize the anchors during the training procedure, and pursue a desired anchor distribution for clustering. Theoretical analysis shows that DMAC has a linear time complexity $O(n)$. Experiments report the superior performance and efficiency of DMAC.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (Grant No: 2022ZD0160803), and the National Natural Science Foundation of China (Grant No: 61871470).

References

- Belhumeur, P. N.; Hespanha, J. P.; and Kriegman, D. J. 1997. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7): 711–720.
- Bryant, C.; and Ng, H. T. 2015. How far are we from fully automatic high quality grammatical error correction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 697–707.
- Chen, J.; Mao, H.; Woo, W. L.; and Peng, X. 2023. Deep multiview clustering by contrasting cluster assignments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16752–16761.
- Chen, M.; Wang, B.; and Li, X. 2024. Deep Contrastive Graph Learning with Clustering-Oriented Guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 11364–11372.
- Chen, X.; and Cai, D. 2011. Large scale spectral clustering with landmark-based representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 313–318.
- Cheng, J.; Wang, Q.; Tao, Z.; Xie, D.; and Gao, Q. 2021. Multi-view attribute graph convolution networks for clustering. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, 2973–2979.
- Cui, C.; Ren, Y.; Pu, J.; Li, J.; Pu, X.; Wu, T.; Shi, Y.; and He, L. 2024. A novel approach for effective multi-view clustering with information-theoretic perspective. *Advances in Neural Information Processing Systems*, 36.
- Cui, C.; Ren, Y.; Pu, J.; Pu, X.; and He, L. 2023. Deep Multi-view Subspace Clustering with Anchor Graph. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 3577–3585.
- Dong, Z.; Wang, S.; Jin, J.; Liu, X.; and Zhu, E. 2023. Cross-view topology based consistent and complementary information for deep multi-view clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19440–19451.
- Fang, S.-G.; Huang, D.; Cai, X.-S.; Wang, C.-D.; He, C.; and Tang, Y. 2023a. Efficient multi-view clustering via unified and discrete bipartite graph learning. *IEEE Transactions on Neural Networks and Learning Systems*.
- Fang, U.; Li, M.; Li, J.; Gao, L.; Jia, T.; and Zhang, Y. 2023b. A comprehensive survey on multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(12): 12350–12368.
- Greene, D.; and Cunningham, P. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the International Conference on Machine Learning*, 377–384.
- Gross, R.; Matthews, I.; Cohn, J.; Kanade, T.; and Baker, S. 2010. Multi-pie. *Image and Vision Computing*, 28(5): 807–813.
- Hartigan, J. A.; and Wong, M. A. 1979. Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society: Series C (applied statistics)*, 28(1): 100–108.
- Houle, M. E.; Kriegel, H.-P.; Kröger, P.; Schubert, E.; and Zimek, A. 2010. Can shared-neighbor distances defeat the curse of dimensionality. In *Proceedings of the Scientific and Statistical Database Management*, 482–500.
- Hu, S.; Zou, G.; Zhang, C.; Lou, Z.; Geng, R.; and Ye, Y. 2023. Joint contrastive triple-learning for deep multi-view clustering. *Information Processing and Management*, 60(3): 103284.
- Huang, Z.; Zhou, J. T.; Peng, X.; Zhang, C.; Zhu, H.; and Lv, J. 2019. Multi-view Spectral Clustering Network. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2563–2569.
- Jiang, Y.-G.; Ye, G.; Chang, S.-F.; Ellis, D.; and Loui, A. C. 2011. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *Proceedings of the ACM International Conference on Multimedia Retrieval*, 1–8.
- Kang, Z.; Lin, Z.; Zhu, X.; and Xu, W. 2021. Structured graph learning for scalable subspace clustering: From single view to multiview. *IEEE Transactions on Cybernetics*, 52(9): 8976–8986.
- 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.
- Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*.
- Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Li, X. 2022. Positive-incentive noise. *IEEE Transactions on Neural Networks and Learning Systems*.
- Li, X.; Zhang, H.; Wang, R.; and Nie, F. 2020. Multiview clustering: A scalable and parameter-free bipartite graph fusion method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1): 330–344.
- Li, Z.; Wang, Q.; Tao, Z.; Gao, Q.; Yang, Z.; et al. 2019. Deep adversarial multi-view clustering network. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2952–2958.
- Liu, J.; Cao, F.; Jing, X.; and Liang, J. 2024. Deep multi-view graph clustering network with weighting mechanism and collaborative training. *Expert Systems with Applications*, 236: 121298.
- McInnes, L.; Healy, J.; and Melville, J. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.
- Nie, F.; Wang, C.-L.; and Li, X. 2019. K-multiple-means: A multiple-means clustering method with specified k clusters.

- In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 959–967.
- Nie, F.; Zhu, W.; and Li, X. 2017. Unsupervised large graph embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2422–2428.
- Trosten, D. J.; Lokse, S.; Jenssen, R.; and Kampffmeyer, M. 2021. Reconsidering representation alignment for multi-view clustering. In *Proceedings of the IEEE/CVF 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*, 32(6): 1116–1129.
- Wang, H.; Yang, Y.; Liu, B.; and Fujita, H. 2019. A study of graph-based system for multi-view clustering. *Knowledge-Based Systems*, 163: 1009–1019.
- Wang, J.; Feng, S.; Lyu, G.; and Yuan, J. 2024. SURER: Structure-Adaptive Unified Graph Neural Network for Multi-View Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15520–15527.
- Wang, J.; Tang, C.; Wan, Z.; Zhang, W.; Sun, K.; and Zomaya, A. Y. 2023a. Efficient and effective one-step multiview clustering. *IEEE Transactions on Neural Networks and Learning Systems*.
- Wang, J.; Wu, B.; Ren, Z.; Zhang, H.; and Zhou, Y. 2023b. Multi-scale deep multi-view subspace clustering with self-weighting fusion and structure preserving. *Expert Systems with Applications*, 213: 119031.
- Wang, Q.; He, X.; Jiang, X.; and Li, X. 2020. Robust bi-stochastic graph regularized matrix factorization for data clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1): 390–403.
- Wang, Y.; Chang, D.; Fu, Z.; and Zhao, Y. 2021. Consistent multiple graph embedding for multi-view clustering. *IEEE Transactions on Multimedia*, 25: 1008–1018.
- 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*, 25: 9203–9215.
- Xu, J.; Ren, Y.; Li, G.; Pan, L.; Zhu, C.; and Xu, Z. 2021. Deep embedded multi-view clustering with collaborative training. *Information Sciences*, 573: 279–290.
- Xu, J.; Ren, Y.; Tang, H.; Yang, Z.; Pan, L.; Yang, Y.; Pu, X.; Philip, S. Y.; and He, L. 2022a. Self-supervised discriminative feature learning for deep multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(7): 7470–7482.
- Xu, J.; Tang, H.; Ren, Y.; Peng, L.; Zhu, X.; and He, L. 2022b. Multi-level feature learning for contrastive multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16051–16060.
- Yan, W.; Zhang, Y.; Lv, C.; Tang, C.; Yue, G.; Liao, L.; and Lin, W. 2023. Gcfagg: Global and cross-view feature aggregation for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19863–19872.
- Yang, B.; Wu, J.; Zhang, X.; Zheng, X.; Nie, F.; and Chen, B. 2024. Discrete correntropy-based multi-view anchor-graph clustering. *Information Fusion*, 103: 102097.
- Yang, X.; Jiaqi, J.; Wang, S.; Liang, K.; Liu, Y.; Wen, Y.; Liu, S.; Zhou, S.; Liu, X.; and Zhu, E. 2023. Dealmvc: Dual contrastive calibration for multi-view clustering. In *Proceedings of the ACM International Conference on Multimedia*, 337–346.
- Zhang, B.; and Sennrich, R. 2019. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32.
- Zhang, C.; Jia, X.; Li, Z.; Chen, C.; and Li, H. 2024. Learning Cluster-Wise Anchors for Multi-View Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 16696–16704.
- Zhang, H.; Shi, J.; Zhang, R.; and Li, X. 2022. Non-Graph Data Clustering via O(n) Bipartite Graph Convolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7): 8729–8742.
- Zhang, P.; Wang, S.; Li, L.; Zhang, C.; Liu, X.; Zhu, E.; Liu, Z.; Zhou, L.; and Luo, L. 2023. Let the data choose: Flexible and diverse anchor graph fusion for scalable multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 11262–11269.