Information Sciences 667 (2024) 120458

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Dual auto-weighted multi-view clustering via autoencoder-like nonnegative matrix factorization

Si-Jia Xiang [a], Heng-Chao Li [a,b], Jing-Hua Yang [a,*], Xin-Ru Feng [c,*]

[a] School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China
[b] National Engineering Laboratory of Integrated Transportation Big Data Application Technology, Southwest Jiaotong University, Chengdu 611756, China
[c] College of Computer Science, Sichuan Normal University, Chengdu 610066, China

ARTICLE INFO

Keywords:
Multi-view clustering
Dual auto-weights
Autoencoder-like nonnegative matrix factorization
Adaptive graph learning

ABSTRACT

Multi-view clustering (MVC) can exploit the complementary information among multi-view data to achieve the satisfactory performance, thus having extensive potentials for practical applications. Although Nonnegative Matrix Factorization (NMF) has emerged as an effective technique for MVC, the existing NMF-based methods still have two main limitations: 1) They solely focus on the reconstruction of original data, which can be regarded as the decoder of an autoencoder, while neglecting the low-dimensional representation learning. 2) They lack the ability to effectively capture both linear and nonlinear structures of data. To solve these problems, in this paper, we propose a **D**ual **A**uto-weighted multi-view clustering model based on **A**utoencoder-like **NMF** (DA^{2}NMF), which enables a comprehensive exploration of both linear and nonlinear structures. Specifically, we establish an autoencoder-like NMF model that learns linear low-dimensional representations by integrating data reconstruction and representation learning within a unified framework. Moreover, the adaptive graph learning is introduced to explore the nonlinear structures in data. We further design a dual auto-weighted strategy to adaptively compute weights for different views and low-dimensional representations, thereby obtaining an enhanced consistent graph. An effective algorithm based on Multiplicative Update Rule (MUR) is developed to solve the DA^{2}NMF with the theoretical convergence guarantee. Experimental results show that the proposed DA^{2}NMF can effectively improve the clustering performance compared with the state-of-the-art MVC algorithms.

1. Introduction

In the fields of data mining and analysis [1–4], data often originates from multiple sources and exhibits different characteristics [5], leading to the generation of multi-view data [6]. For example, a single piece of news can be documented across several languages, an image can be characterized by diverse sets of features, and a signal can be displayed by distinct waveforms in the frequency and time domains. These examples all represent multi-view data, where each view holds unique attributes and offers supplementary information compared to that of other views. Therefore, to capture the complementary and consistent information inherent in multi-

* Corresponding authors.
E-mail addresses: yangjinghua110@126.com (J.-H. Yang), xrfeng@sicnu.edu.cn (X.-R. Feng).

https://doi.org/10.1016/j.ins.2024.120458
Received 4 October 2023; Received in revised form 4 March 2024; Accepted 7 March 2024

Available online 13 March 2024

view data, multi-view clustering (MVC) has received widespread attention and research [1,7–9]. However, handling multi-view data poses challenges due to complex features, diverse data sources, large volumes, and high dimensionality within data [10,11].

In recent years, there are many methods to deal with multi-view data, such as spectral clustering [12–15,7], graph-based clustering [16–18,8,19], subspace-based clustering [20,9], and nonnegative matrix factorization (NMF)-based clustering [21,22]. Due to the advantages of NMF in dimensionality reduction and interpretability, we mainly focus on NMF-based clustering methods. Typically, NMF-based clustering involves two steps. Firstly, NMF is used to learn low-dimensional representations of data [23,24]. Then an additional post-processing step like K-means [25] is applied to obtain the final clustering results [26]. For example, Yang et al. [27] introduced sparse NMF and proposed a document clustering algorithm by explicitly constraining the sparsity of the low-dimensional matrices. Wild et al. [28] developed an initialization technique for NMF to improve the speed of clustering tasks. One of the most representative works is Symmetric NMF (SymNMF) [29], which decomposed the data into a symmetric matrix containing pairwise similarity values and captured the clustering structure. Nevertheless, the above methods are only suitable for processing single-view data, which fails to effectively integrate feature information from different views.

To handle multi-view data, a substantial number of MVC methods based on NMF and its variants [30,23] have emerged. Gao et al. [21] proposed a multi-view NMF (MultiNMF) model, where the clustering structures of different views are preserved during the joint factorization process, and the consensus representation is learnt by coefficient matrices of different views. Unlike most MVC methods that only consider the consistency among multiple views, Liu et al. [30] introduced a semi-supervised multi-view learning method to jointly explore both the consistent and complementary information across views. Cai et al. [31] proposed a semi-supervised MVC method based on orthonormality-constrained NMF, which utilized constrained NMF to learn low-dimensional representations of data and used co-regularization to integrate complementary information from different views. Though these NMF-based methods allow for data dimensionality reduction and achieve latent linear representation of data, they are hardly to preserve geometric structures, limiting their ability to capture complex nonlinear relationships in multi-view data. To preserve the geometric structure of data, graph regularization-based NMF has been proposed for MVC task. Zhang et al. [32] applied graph regularization on the coefficient matrices of each view to preserve the intrinsic structure of data. Xu et al. proposed a MVC via consistent and specific NMF with graph regularization (MCCS) method and designed a disagreement regularization term to learn a common representation, thus a same underlying cluster structure from multiple views can be ensured. Liu et al. [33] proposed deep manifold regularized semi-NMF for MVC and employed graph regularization at each layer of deep matrix factorization to extract complex structural information within data. The studies mentioned above use a fixed graph construction method that may produce clustering results sensitive to the predefined similarity matrix and potentially disrupt the local connectivity of the data [34]. Despite significant progress in NMF-based MVC, these methods still exhibit three limitations: 1) NMF-based methods primarily emphasize the decoder, which reconstructs the original data from its low-dimensional representation, neglecting the encoding process of directly projecting input data into a low-dimensional representation. 2) The linear and nonlinear structures inherent in multi-view data have not been thoroughly explored. 3) These methods do not adequately consider the contributions of different views, resulting in suboptimal clustering results.

To solve the above limitations, we propose a Dual Auto-weighted multi-view clustering model based on Autoencoder-like NMF (DA^2NMF), which not only retains the linear information from specific views but also incorporates adaptive graph learning to explore the nonlinear structures of multi-view data, as illustrated in Fig. 1. Unlike traditional NMF-based algorithms that focus solely on reconstructing the original data through the decoder, our model introduces both an encoder and a decoder to learn low-dimensional representations. Moreover, we design a dual auto-weighted strategy to adaptively assign weights for each view and view-specific low-dimensional representation, enabling DA^2NMF to obtain a more discriminative fused graph. Throughout the optimization iterations, the three modules, i.e., low-dimensional representation learning, consistent graph learning, and dual auto-weighted strategy are simultaneously optimized and they enhance each other in our proposed method. The contributions of this work are as follows:

1) We propose a Dual Auto-weighted multi-view clustering model based on Autoencoder-like NMF (DA^2NMF), which integrates the autoencoder-like NMF and adaptive graph learning into a unified framework. Moreover, by considering the dual auto-weighted strategy, the DA^2NMF effectively accounts for the importances of different views and the weights of view-specific low-dimensional representations to learn a more discriminative consistent graph.

2) By considering both the reconstruction of multi-view data and representation learning, an autoencoder-like NMF is constituted to obtain the better latent representations in a low-dimensional space. Moreover, based on the view-specific low-dimensional representations, we introduce the adaptive graph learning to adaptively explore the nonlinear structure of multi-view data. Our framework effectively integrates both the latent linear and nonlinear structures of multi-view data to obtain an enhanced consistent graph.

3) We develop an iterative algorithm based on Multiplicative Update Rules (MUR) to effectively solve the optimization problem of the DA^2NMF model. We also theoretically prove the convergence of the algorithm. In the experiments, the proposed DA^2NMF outperforms the comparative methods, showing the effectiveness of our method.

The paper is organized as follows: Section 2 provides a brief review of related works. In Section 3, the DA^2NMF model and optimization algorithm are introduced. In Section 4, extensive experiments are tested to validate the effectiveness of DA^2NMF. Section 5 involves a discussion on the experimental results. Section 6 concludes the paper.

2. Related work

MVC has been a popular topic which aims at exploring the inherent correlation and consistency among different views for clustering [35]. In this section, we review some related works about MVC, which consists of four parts: spectral MVC, graph-based MVC, subspace-based MVC, and NMF-based MVC.

2.1. Spectral MVC

Spectral clustering transforms the clustering problem into a graph partitioning problem [36]. The input to spectral clustering is a similarity graph, whose optimization objective is to minimize the similarity relations between different components of the graph [37]. Kumar et al. [12] constructed a spectral clustering framework, which introduces pairwise co-regularization as well as centroid-based co-regularization to balance view-specific eigenvectors and consensus eigenvectors. To achieve a robust spectral clustering result, Huang et al. [13] proposed affinity aggregation spectral clustering (AASC) method for alleviating the impact of irrelevant features. Motivated by the same purpose, Xia et al. [14] used Markov chain method and proposed robust multi-view spectral clustering (RMSC) to explicitly handle the noise in the transition probability matrices among views. Considering the diversity of different views, Zong et al. [7] employed the spectral perturbation to model the weights of views, which allowed the clustering results among views to approximate the consensus clustering result and smoothes the weights.

2.2. Graph-based MVC

Graph-based MVC achieves a fused graph by integrated graphs from multiple views. It can be noted that graph-based MVC and spectral MVC exhibit certain similarities; however, their distinction lies in the fact that spectral MVC typically identifies a low-dimensional embedding representation of the data, while graph-based MVC generates clusters on the constructed data graph rather than a new embedding representation [8]. Zhan et al. [16] pointed out most graph-based MVC need a predefined graph, the quality of which can significantly influence the clustering performance. Based on the concern, multiview clustering with graph learning (MVGL) [16] and multiview consensus graph clustering (MCGC) [17] methods were proposed to enhance the quality of the predefined graph. Different from MVGL, MCGC employed a specific disagreement cost for exploring consistency across different views. Moreover, Wang et al. [8] proposed a general graph-based MVC (GMC) method to construct the graph of each view and learn a fusion graph in a mutually reinforcing way. To reduce the computational complexity, Tang et al. [19] designed a parameter-free method and learnt a unified graph for MVC via cross-view graph diffusion.

2.3. Subspace-based MVC

Subspace-based MVC aims to obtain a shared low-dimensional space from different views, in which a consistent low-dimensional representation of multi-view data can be found for clustering. This method can reduce redundant information of data and has low computational complexity. Liu et al. [20] proposed multi-view subspace clustering algorithm, which preserves the locally consistent geometric relationship and ensure the consistency across different views. Wang et al. [9] pointed out that most subspace-based MVC methods suffer from cubic time complexity and are not suitable for dealing with large-scale datasets. Subsequently, a fast parameter-free multiview subspace clustering method with consensus anchor guidance (FPMVS-CAG) was proposed to automatically learn an optimal anchor subspace graph without any extra hyper-parameters, which was suitable for large-scale multi-view data clustering.

2.4. NMF-based MVC

To handle the high-dimensional data clustering, NMF-based MVC derives a representation in low-dimensional spaces by decomposing original data into two nonnegative low-rank matrices, and the clustering results can be obtained by K-means. In recent years, many extensions of NMF-based MVC have been proposed, such as constrained NMF-based MVC [32] with additional constraint terms, structured NMF-based MVC [38] by altering the loss function structure, and generalized NMF-based MVC [33] by extending the decomposition form [23]. Generally, the constrained NMF-based MVC has been widely studied, which imposes additional constraint terms such as graph regularizer and orthogonality constraint [31]. For instance, by integrating concept factorization, manifold regularization and the consistency constraint into a unified framework, the multi-view concept clustering (MVCC) [22] method was proposed to learn a consistent representation across different views. Moreover, Luong et al. [39] proposed a MVC framework of deep NMF by applying diversity constraint, orthogonal constraint, and cut-type constraint to enhance the learning of shared and complementary information among the views.

Compared to the spectral MVC, graph-based MVC, and subspace-based MVC mentioned above, NMF-based MVC have the following two advantages: 1) NMF-based MVC ensures that all elements in the generated decomposition matrix are nonnegative, making the results easier to interpret. 2) The computational complexity of NMF-based MVC is lower, especially on large-scale datasets. Inspired by the above, we further propose a new MVC framework that incorporates the inherent linear and nonlinear structures of multi-view data, taking into account the significance of different views and low-dimensional representations.

3. Method

In this paper, matrices are denoted as uppercase letters such as $\mathbf{X}$. We use $\mathbf{x}_i$ and x_{ij} to denote the i-th column and the ij-th entry of $\mathbf{X}$, respectively. The trace of $\mathbf{X}$ is denoted by $Tr(\mathbf{X})$. We use $\|\mathbf{X}\|_F$ and $\|\mathbf{X}\|_2$ to describe the Frobenius norm and the l_2 norm of $\mathbf{X}$, respectively. Assuming a multi-view dataset has d views, and $\mathbf{X}^v = \{\mathbf{x}_1^v, ..., \mathbf{x}_n^v\} \in \mathbb{R}^{m\times n}$ is the v-th view data, where m is the dimension of features and n denotes the number of instances. Table 1 presents a comprehensive list of mathematical symbols in this paper along with their corresponding descriptions.

Table 1
Some notations and definitions used in DA^{2}NMF.

Notations	Definitions
m	The dimension of features
n	The number of instances
c	The number of clusters
d	The number of views
k	The reduced feature dimensionality
$\mathbf{X}, \mathbf{x}_i, x_{ij}$	The matrix, the i-th column, the ij-th entry
$\mathbf{X}^v \in \mathbb{R}^{m\times n}$	The v-th view data
$\mathbf{U}^v \in \mathbb{R}^{m\times k}$	The v-th view basis matrix
$\mathbf{V}^v \in \mathbb{R}^{n\times k}$	The v-th view representation matrix
$\mathbf{S} \in \mathbb{R}^{\mathbf{n}\times\mathbf{n}}$	The fused graph
$Tr(\mathbf{X})$	The trace of $\mathbf{X}$
$\|\mathbf{X}\|_F$	The Frobenius norm of $\mathbf{X}$
$\|\mathbf{X}\|_2$	The l_2 norm of $\mathbf{X}$
$\boldsymbol{\sigma}^v$, $\boldsymbol{\omega}^v$	The auto-weighted parameters
α, β	The hyperparameters

3.1. The proposed DA^{2}NMF model

In this section, we present the DA^{2}NMF model and the optimization algorithm in detail. Our DA^{2}NMF model consists of three components: autoencoder-like NMF, adaptive graph learning, and dual auto-weighted strategy, which are unified within a framework, as shown in Fig. 1.

3.1.1. Autoencoder-like NMF

The core of NMF is to reconstruct the data $\mathbf{X}^v$ from basis matrix $\mathbf{U}^v$ and low-dimensional representation $\mathbf{V}^v$ [40], which can be seen as a decoding process. The objective function is

$$\begin{aligned} &\min_{\mathbf{U}^v,\mathbf{V}^v} \|\mathbf{X}^v - \mathbf{U}^v(\mathbf{V}^v)^T\|_F^2 \\ &s.t.\ \ \mathbf{U}^v \geq 0, \mathbf{V}^v \geq 0. \end{aligned} \tag{1}$$

However, Eq. (1) neglects the encoding process, which involves the transformation of $\mathbf{X}^v$ into a more optimized $\mathbf{V}^v$. Hence, we introduce an autoencoder-like NMF to jointly consider the reconstruction of original data and the learning of low-dimensional representations. The optimization problem is given by

$$\begin{aligned} &\min_{\mathbf{U}^v,\mathbf{V}^v} \sum_{v=1}^{d} (\|\mathbf{X}^v - \mathbf{U}^v(\mathbf{V}^v)^T\|_F^2 + \beta\|\mathbf{V}^v - (\mathbf{U}^v)^T\mathbf{X}^v\|_F^2) \\ &s.t.\ \ \mathbf{U}^v \geq 0, \mathbf{V}^v \geq 0, \end{aligned} \tag{2}$$

where the first term in Eq. (2) is decoder, which reconstructs the input matrices from low-dimensional representations. The encoder can directly project the input data into low-dimensional representations, as indicated in the second term of Eq. (2). According to Eq. (2), the linear low-dimensional representation of each view can be adequately learnt.

Remark 1. Autoencoder-like NMF refers to a variant of NMF that is inspired by the architecture and principles of the autoencoder, which comprises an encoder and a decoder. The encoder projects the original data into a low-dimensional representation, while the decoder reconstructs the original data based on this representation. In contrast to conventional NMF algorithms that primarily focus on the decoder, the autoencoder-like NMF can obtain the better latent representations in a low-dimensional space.

3.1.2. Adaptive graph learning

Despite addressing some drawbacks of traditional NMF, autoencoder-like NMF remains with two limitations: 1) it ignores the nonlinear structure within the original data, and 2) it lacks consistency information descriptions. To address the aforementioned limitations, inspired by [41], we incorporate the idea of adaptive graph learning into our model for preserving nonlinear structures. Adaptive graph learning obtains the similarity relationships between samples by optimizing the following problem:

$$\begin{aligned} &\min_{\mathbf{S}} \sum_{i,j}^{N} (\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 s_{ij} + \gamma s_{ij}^2) \\ &s.t. \forall i, \mathbf{s}_j^T\mathbf{1} = 1, 0 \leq s_{ij} \leq 1, \end{aligned} \tag{3}$$

where s_{ij} is the ij-th entry of a similarity matrix, and it can describe the distance between data points x_i and x_j, reflecting the relationship between the data points. Specifically, a smaller distance $\|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ implies that x_i is closer to x_j, resulting in a larger

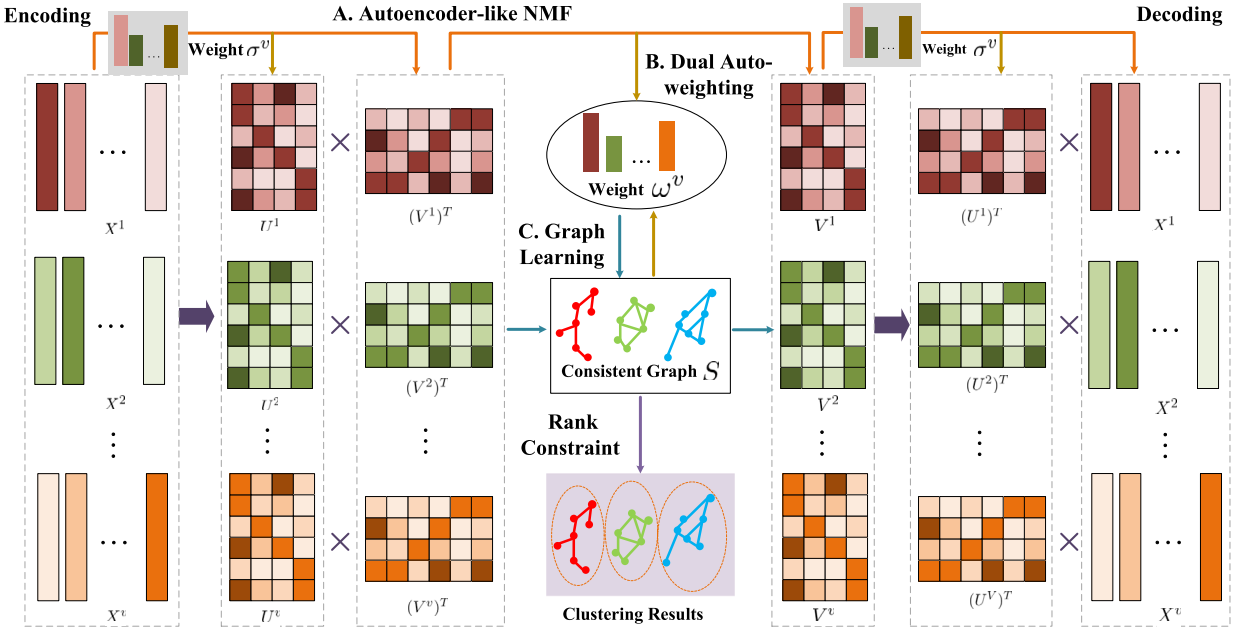


Fig. 1. Overview of the DA^{2}NMF. DA^{2}NMF comprises three crucial components: A. autoeoder-like NMF, B. graph learning, and C. dual auto-weighted strategy. Specifically, autoencoder-like NMF explores the linear latent representations $\{\mathbf{V}^1,...,\mathbf{V}^v\}$ of data $\{\mathbf{X}^1,...,\mathbf{X}^v\}$. Graph learning focuses on capturing the nonlinear structures within the low-dimensional representations and obtaining a consistent graph $\mathbf{S}$. And dual auto-weighted strategy dynamically allocates weight σ^v to each view $\mathbf{X}^v$. Simultaneously, when constructing the consistent graph, the auto-weight ω^v is assigned to each low-dimensional representation $\mathbf{V}^v$, enabling the generation of an enhanced consistent graph.

value of s_{ij}. Based on Eq. (3), we perform adaptive graph learning on view-specific representations and obtain a consistent graph. The objective function is

$$\begin{aligned}&\min_{\mathbf{U}^v,\mathbf{V}^v,\mathbf{S}} \sum_{v=1}^{d}(\|\mathbf{X}^v-\mathbf{U}^v(\mathbf{V}^v)^T\|_F^2+\beta\|\mathbf{V}^v-(\mathbf{U}^v)^T\mathbf{X}^v\|_F^2)+\sum_{v=1}^{d}\sum_{i,j}^{n}\|\mathbf{v}_i^v-\mathbf{v}_j^v\|_2^2 s_{ij}+\alpha\|\mathbf{S}\|_F^2 \\ &s.t. \quad \mathbf{U}^v\geq 0,\mathbf{V}^v\geq 0,\mathbf{s}_j^T\mathbf{1}=1,0\leq s_{ij}\leq 1.\end{aligned} \tag{4}$$

The last two terms in Eq. (4) represent the adaptive graph learning, whose basic idea is that a large probability value s_{ij} should be assigned when the distance between two data points is small.

Remark 2. Adaptive graph learning is employed to explore nonlinear relationships within view-specific low-dimensional representation $\mathbf{V}^v$. By automatically updating the edges for each data point (i.e., vertex v_i), a similarity graph $\mathbf{S}$ is dynamically constructed to capture more flexible nonlinear relationships.

3.1.3. Dual auto-weighted strategy

Furthermore, there is diversity among each view, so it is essential to distinguish the importance of different views in multi-view learning [42]. We introduce the concept of auto-weighting to design a dual auto-weighted strategy. Eq. (4) can be transformed into

$$\begin{aligned}&\min_{\mathbf{U}^v,\mathbf{V}^v,\mathbf{S},\mathbf{F},\sigma^v,\omega^v} \sum_{v=1}^{d}\sigma^v(\|\mathbf{X}^v-\mathbf{U}^v(\mathbf{V}^v)^T\|_F^2+\beta\|\mathbf{V}^v-(\mathbf{U}^v)^T\mathbf{X}^v\|_F^2) \\ &\qquad +\sum_{v=1}^{d}\omega^v\sum_{i,j}^{n}\|\mathbf{v}_i^v-\mathbf{v}_j^v\|_2^2 s_{i,j}+\alpha\|\mathbf{S}\|_F^2 \\ &s.t. \qquad \mathbf{U}^v\geq 0,\mathbf{V}^v\geq 0,\mathbf{s}_j^T\mathbf{1}=1,0\leq s_{ij}\leq 1,\end{aligned} \tag{5}$$

where σ^v and ω^v represent auto-weighted parameters, σ^v is assigned to balance the importance of each view to learn the discriminative low-dimensional representation, and ω^v is allocated to measure the significance of each latent low-dimensional representation when constructing the consistent graph.

Remark 3. The dual auto-weighted strategy is crucial as it takes into account the significance of different views and different low-dimensional representations, learning a discriminative consistency graph. Specifically, σ^v is automatically assigned to different views

with the aim of obtaining reliable view-specific low-dimensional representations. Similarly, ω^v is allocated adaptively to measure the importance of different low-dimensional representations.

3.1.4. Objective function

Eq. (5) is capable of adequately capturing both linear structures and nonlinear relationships within data. Building upon this foundation, we introduce a rank constraint on the Laplacian matrix of the consistent graph. This constraint ensures that the learned consistent graph contains c connected components, corresponding to c clusters, thereby directly obtaining the clustering results and avoiding post-processing steps. The objective function is

$$\begin{aligned}\min_{\mathbf{U}^v,\mathbf{V}^v,\mathbf{S},\mathbf{F},\sigma^v,\omega^v} & \sum_{v=1}^{d}\sigma^v(\|\mathbf{X}^v-\mathbf{U}^v(\mathbf{V}^v)^T\|_F^2+\beta\|\mathbf{V}^v-(\mathbf{U}^v)^T\mathbf{X}^v\|_F^2)\\ &+\sum_{v=1}^{d}\omega^v\sum_{i,j}^{n}\|\mathbf{v}_i^v-\mathbf{v}_j^v\|_2^2 s_{i,j}+\alpha\|\mathbf{S}\|_F^2\\ s.t.\quad & \mathbf{U}^v\geq 0,\mathbf{V}^v\geq 0,\mathbf{s}_j^T\mathbf{1}=1,0\leq s_{ij}\leq 1,rank(\mathbf{L}_s)=n-c,\end{aligned} \tag{6}$$

where $\mathbf{L}_s=\mathbf{D}-(\mathbf{S}+\mathbf{S}^T)/2$ is the graph Laplacian matrix, $\mathbf{D}\in\mathbb{R}^{n\times n}$ is a diagonal matrix with the ith diagonal element calculated by $d_{ii}=\sum_j(s_{ij}+s_{ji})/2$. However, Eq. (6) is difficult to solve. Based on the Ky Fan's Theorem [43], the rank constraint $rank(\mathbf{L}_s)=n-c$ can be equivalently replaced by minimizing $Tr(\mathbf{F}^T\mathbf{L}_s\mathbf{F})$ subject to $\mathbf{F}\in\mathbb{R}^{c\times n}$ and $\mathbf{F}^T\mathbf{F}=\mathbf{I}$. Mathematically, the overall objective function of DA^{2}NMF is

$$\begin{aligned}\min_{\mathbf{U}^v,\mathbf{V}^v,\mathbf{S},\mathbf{F},\sigma^v,\omega^v} & \sum_{v=1}^{d}\sigma^v(\|\mathbf{X}^v-\mathbf{U}^v(\mathbf{V}^v)^T\|_F^2+\beta\|\mathbf{V}^v-(\mathbf{U}^v)^T\mathbf{X}^v\|_F^2)\\ &+\sum_{v=1}^{d}\omega^v\sum_{i,j}^{n}\|\mathbf{v}_i^v-\mathbf{v}_j^v\|_2^2 s_{i,j}+\alpha\|\mathbf{S}\|_F^2+2\lambda Tr(\mathbf{F}^T\mathbf{L}_s\mathbf{F})\\ s.t.\quad & \mathbf{U}^v\geq 0,\mathbf{V}^v\geq 0,\mathbf{s}_j^T\mathbf{1}=1,0\leq s_{ij}\leq 1,\mathbf{F}^T\mathbf{F}=\mathbf{I},\end{aligned} \tag{7}$$

where $\mathbf{F}=[f_1,...,f_n]$ is the clustering indicator matrix.

In summary, our DA^{2}NMF model exhibits four advantages: 1) DA^{2}NMF utilizes autoencoder-like NMF by integrating both an encoder and a decoder, thereby comprehensively learning the latent linear low-dimensional representations in data. 2) DA^{2}NMF incorporates graph learning to capture the nonlinear features of data. 3) DA^{2}NMF designs a dual auto-weighted strategy, which is capable of measuring the importance of different views as well as different low-dimensional representations, thereby obtaining a discriminative graph. 4) DA^{2}NMF yields clustering results directly without the post-processing, and the clustering indicator matrix contributes to updating of the consistent graph and learning of low-dimensional representations.

3.2. Optimization algorithm

To solve the optimization problem in Eq. (7), an iterative update procedure is designed based on MUR. The updating rules, complexity analysis, and convergence analysis are presented below.

3.2.1. Updating rules

The updating rules of Eq. (7) can be divided into six steps. That is updating $\mathbf{U}^v$, $\mathbf{V}^v$, $\mathbf{S}$, $\mathbf{F}$, $\boldsymbol{\sigma}^v$, and $\boldsymbol{\omega}^v$.

A. Updating $\mathbf{U}^v$. Fixing the other variables, we update $\mathbf{U}^v$. The calculation of $\mathbf{U}^v$ for the v-th view, when the other variables are fixed, does not depend on $\mathbf{U}^{v'}$ or $\mathbf{V}^{v'}$ $(v'\neq v)$. For a more refined depiction, $\mathbf{X},\mathbf{U}$, and $\mathbf{V}$ are used instead of $\mathbf{X}^v$, $\mathbf{U}^v$, and $\mathbf{V}^v$, respectively. Hence, the objective function for the v-th view can be simplified as

$$\mathcal{L}_1=\|\mathbf{X}-\mathbf{U}\mathbf{V}^T\|_F^2+\beta\|\mathbf{V}^T-\mathbf{U}^T\mathbf{X}\|_F^2. \tag{8}$$

We transform Eq. (8) into the form of a trace

$$\begin{aligned}\mathcal{L}_1=&Tr[\mathbf{X}^T\mathbf{X}-\mathbf{X}^T\mathbf{U}\mathbf{V}^T-\mathbf{V}\mathbf{U}^T\mathbf{X}+\mathbf{V}\mathbf{U}^T\mathbf{U}\mathbf{V}^T]\\ &+\beta Tr[\mathbf{V}\mathbf{V}^T-\mathbf{V}^\mathbf{T}\mathbf{U}^\mathbf{T}\mathbf{X}-\mathbf{X}^\mathbf{T}\mathbf{U}\mathbf{V}^\mathbf{T}+\mathbf{X}^\mathbf{T}\mathbf{U}\mathbf{U}^\mathbf{T}\mathbf{X}].\end{aligned} \tag{9}$$

Utilizing the Lagrange multiplier matrix Ψ as a nonnegative constraint on $\mathbf{U}$ and eliminating the irrelevant terms of $\mathbf{U}$, the sub-problem of Eq. (9) can be formulated as

$$\begin{aligned}\mathcal{L}_1=&Tr[-2\mathbf{X}^T\mathbf{U}\mathbf{V}^T+\mathbf{V}\mathbf{U}^\mathbf{T}\mathbf{U}\mathbf{V}^\mathbf{T}]\\ &+\beta Tr[-2\mathbf{X}^\mathbf{T}\mathbf{U}\mathbf{V}^\mathbf{T}+\mathbf{X}^\mathbf{T}\mathbf{U}\mathbf{U}^\mathbf{T}\mathbf{X}]+Tr[\Psi\mathbf{U}^T].\end{aligned} \tag{10}$$

Taking the derivative of problem (10) with respect of $\mathbf{U}$, we can obtain

$$\frac{\partial \mathcal{L}1}{\partial \mathbf{U}} = -2\mathbf{XV} + 2\mathbf{UV^T V} + 2\beta(-\mathbf{XV} + \mathbf{X^T U}) + \Psi. \tag{11}$$

According to the Karush Kuhn-Tucker (KKT) conditions [44], $\Psi \odot \mathbf{U} = \mathbf{0}$ and the update of $\mathbf{U}$ can be obtained as follows

$$\mathbf{U} \leftarrow \mathbf{U} \odot \frac{(1+\beta)\mathbf{XV}}{\mathbf{UV^T V} + \beta \mathbf{XX^T U}}, \tag{12}$$

where $\odot$ denotes the element-wise product.

B. Updating $\mathbf{V}^v$. Fixing the other variables, and we can obtain the objective function of $\mathbf{V}^v$

$$\mathcal{L}_2 = \sigma(\|\mathbf{X} - \mathbf{UV}^T\|_F^2 + \beta\|\mathbf{V} - \mathbf{U}^T\mathbf{X}\|_F^2) + 2\omega Tr(\mathbf{VL}_s\mathbf{V}^T). \tag{13}$$

The trace form can be formulated as

$$\begin{aligned}\mathcal{L}_2 = \sigma(&Tr[\mathbf{X}^T\mathbf{X} - \mathbf{X}^T\mathbf{UV}^T - \mathbf{VU}^T\mathbf{X} + \mathbf{VU}^T\mathbf{UV}^T] \\ &+ \beta Tr[\mathbf{VV}^T - \mathbf{V^T U^T X} - \mathbf{X^T UV^T} + \mathbf{X^T UU^T X}]) + 2\omega Tr(\mathbf{VL}_s\mathbf{V}^T).\end{aligned} \tag{14}$$

Using the Lagrange multiplier matrix Φ as a non-negative constraint on $\mathbf{V}$ and removing the irrelevant terms of $\mathbf{V}$, the sub-problem of Eq. (14) can be formulated as

$$\begin{aligned}\mathcal{L}_2 = \sigma(&Tr[-2\mathbf{X}^T\mathbf{UV}^T + \mathbf{VU^T UV^T}] \\ &+ \beta Tr[\mathbf{VV^T} - \mathbf{2X^T UV^T}]) + 2\omega Tr(\mathbf{VL}_s\mathbf{V}^T).\end{aligned} \tag{15}$$

Taking the derivative of problem (15) with respect of $\mathbf{V}$, we can obtain

$$\frac{\partial \mathcal{L}_2}{\partial \mathbf{V}} = \sigma(-2\mathbf{X}^T\mathbf{U} + 2\mathbf{VU^T U} + 2\beta(\mathbf{V} - \mathbf{X^T U})) + 2\omega\mathbf{L}_s\mathbf{V} + \Psi. \tag{16}$$

According to the KKT conditions, $\Phi \odot \mathbf{V} = 0$, the update of $\mathbf{V}$ can be obtained as follows

$$\mathbf{V} \leftarrow \mathbf{V} \odot \frac{\sigma(1+\beta)\mathbf{X^T U}}{\sigma(\mathbf{VU^T U} + \beta\mathbf{V}) + \omega\mathbf{L}_s\mathbf{V}}. \tag{17}$$

C. Updating S. Fixing $\mathbf{U}$, $\mathbf{V}$, $\mathbf{F}$, σ^v, ω^v, and updating $\mathbf{S}$, we can get the objective function

$$\begin{aligned}\mathcal{L}_3 &= \sum_{v=1}^{d}\omega^v \sum_{i,j}^{n}\|\mathbf{v}_i^v - \mathbf{v}_j^v\|_2^2 s_{i,j} + \alpha\|\mathbf{S}\|_F^2 + 2\lambda Tr(\mathbf{F}^T\mathbf{L}_s\mathbf{F}) \\ &= \sum_{i,j}^{n}(\sum_{v=1}^{d}\omega^v\|\mathbf{v}_i^v - \mathbf{v}_j^v\|_2^2 s_{i,j} + \alpha s_{ij}^2 + \lambda\|\mathbf{f}_i - \mathbf{f}_j\|_2^2 s_{ij}),\end{aligned} \tag{18}$$

we denote $g_{ij} = \sum_{v=1}^{d}\omega^v\sum_{i,j}^{n}\|\mathbf{v}_i^v - \mathbf{v}_j^v\|_2^2 + \lambda\|\mathbf{f}_i - \mathbf{f}_j\|_2^2$, and the problem turns as follows:

$$\begin{aligned}&\mathcal{L}_3 = \sum_{i,j}^{n}(\alpha(s_{ij} + \frac{1}{2\alpha}g_{ij})^2 - \frac{1}{4\alpha}g_{ij}^2) = \sum_{j}^{n}\|\mathbf{s}_j + \frac{1}{2\alpha}\mathbf{g}_j\|_2^2 - \sum_{i,j}^{n}\frac{1}{4\alpha}g_{ij}^2 \\ &s.t. \quad \mathbf{s}_j^T\mathbf{1} = 1, 0 \le s_{ij} \le 1.\end{aligned} \tag{19}$$

The optimization problem mentioned above becomes

$$\begin{aligned}&\min_{\mathbf{S}} \sum_{j}^{n}\|\mathbf{s}_j + \frac{1}{2\alpha}\mathbf{g}_j\|_2^2 \\ &s.t. \quad \mathbf{s}_j^T\mathbf{1} = 1, 0 \le s_{ij} \le 1,\end{aligned} \tag{20}$$

where $\mathbf{S}$ can be solved column-by-column, and the detailed solution of Eq. (20) can be referred to [45].

D. Updating F. Fixing $\mathbf{U}$, $\mathbf{V}$, $\mathbf{S}$, ω^v, σ^v, and updating $\mathbf{F}$, we can get the objective function

$$\begin{aligned}&\min_{\mathbf{F}} \ Tr(\mathbf{F}^T\mathbf{L}_s\mathbf{F}) \\ &s.t. \ \mathbf{F}^T\mathbf{F} = \mathbf{I}.\end{aligned} \tag{21}$$

This problem can be solved by the eigen-decomposition. And the optimal solution of $\mathbf{F}$ can be formed by the c eigenvectors corresponding to the c smallest eigenvalues of $\mathbf{L}_s$.

E. Updating σ^v. Fixing $\mathbf{U}$, $\mathbf{V}$, $\mathbf{S}$, $\mathbf{F}$, ω^v, and updating σ^v, we can get the objective function. σ^v is updated by

$$\sigma^v = \frac{1}{2\sqrt{\|\mathbf{X}^v - \mathbf{U}^v(\mathbf{V}^v)^T\|_F^2 + \beta\|\mathbf{V}^v - (\mathbf{U}^v)^T\mathbf{X}^v\|_F^2)}}. \tag{22}$$

F. Updating ω^v. Fixing $\mathbf{U}$, $\mathbf{V}$, $\mathbf{S}$, $\mathbf{F}$, σ^v, and updating ω^v, we can get the objective function. ω^v is updated by

$$\omega^v = \frac{1}{2\sqrt{\sum_{ij}^{n} \|v_i^v - v_j^v\|_2^2 s_{ij}}}. \tag{23}$$

Algorithm 1 illustrates the entire solving and updating processes for problem Eq. (7). Note that the parameter α is determined by the value of the nearest neighbor parameter and does not need to be specified as an input [45].

Algorithm 1 The optimization algorithm for DA^{2}NMF.

Input: Multi-view data $\mathbf{X}^1$,...,$\mathbf{X}^v$, parameters β, λ.
1: **Initialize:** Standardized for each view $\mathbf{X}^v$. Using NMF to initialize $\mathbf{U}^v$ and $\mathbf{V}^v$. Initialize $\mathbf{S}$, $\sigma = 1/d$, $\omega = 1/d$, and $\mathbf{F}$.
2: **while** not **converge do**
3: Update $\mathbf{U}^v$ by Eq. (12).
4: Update $\mathbf{V}^v$ by Eq. (17).
5: Update $\mathbf{S}$ by Eq. (20).
6: Update $\mathbf{F}$ by Eq. (21). Here the c smallest eigenvalues.
7: Update σ^v by Eq. (22).
8: Update ω^v by Eq. (23).
9: Check the convergence conditions.
10: **end while**
Output: The similarity matrix $\mathbf{S}$ with exact c connected components.

3.2.2. Complexity analysis

Based on Algorithm 1, the most computationally intensive part of the optimization process is the update step for $\mathbf{U}$, $\mathbf{V}$, $\mathbf{S}$, and $\mathbf{F}$. Specifically, the complexity of updating $\mathbf{U}^v$ in Eq. (12) is $O(mnk + mk^2 + m^2k)$ for the vth view, and the complexity of updating $\mathbf{V}^v$ in Eq. (17) is $O(nmk + nk^2 + n^2k)$ for the vth view. Considering that $m \gg k$ and $n \gg k$, the complexity of updating $\mathbf{U}$, $\mathbf{V}$ is $O(mnk + m^2k + n^2k)$. The complexity of updating these two processes for multi-view data with d views is $O(d(mnk + m^2k + n^2k))$. The complexity of updating $\mathbf{S}$ in Eq. (20) primarily involves the computation of $\mathbf{G}$, which is $O(n^2(d + c))$. The complexity of updating $\mathbf{F}$ in Eq. (21) involves the computation of the eigenvectors of the Laplacian matrix and has a complexity of $O(n^2c)$. Over T iterations, the overall complexity of our DA^{2}NMF algorithm is $O(T(dmnk + dkm^2 + dkn^2 + dn^2 + cn^2) + n^2c)$.

3.2.3. Convergence analysis

According to problem (7), our objective function is not a jointly convex problem. Consequently, seeking the global optimum is a challenging task. Next, we demonstrate that problem (7) achieves local optimum within the framework of Algorithm 1 at the iterations. For convenience, we define the objective function of the optimization problem as follows:

$$\begin{aligned}\mathcal{L} = &\sum_{v=1}^{d} \sigma^v (\|\mathbf{X}^v - \mathbf{U}^v(\mathbf{V}^v)^T\|_F^2 + \beta\|\mathbf{V}^v - (\mathbf{U}^v)^T\mathbf{X}^v\|_F^2) \\ &+ \sum_{v=1}^{d} \omega^v Tr(\mathbf{V}\mathbf{L}_s\mathbf{V}^T) + \alpha\|\mathbf{S}\|_F^2 + 2\lambda Tr(\mathbf{F}^T\mathbf{L}_s\mathbf{F}),\end{aligned} \tag{24}$$

and we have following theorem:

Theorem 1. *Under the updating rules of Algorithm 1, the problem (7) can achieve local optimum.*

To prove the Theorem 1, it is necessary to establish the convergence of the six subproblems within problem (7). When other variables are held constant, the computation of $\mathbf{U}^v$ for the v-th view remains independent of $\mathbf{U}^{v'}$ or $\mathbf{V}^{v'}$ (where $v' \neq v$). Here, $\mathbf{X}, \mathbf{U}$, and $\mathbf{V}$ are used in place of $\mathbf{X}^v, \mathbf{U}^v$, and $\mathbf{V}^v$, respectively. As the updates for $\mathbf{U}$ and $\mathbf{V}$ are relatively complex, we provide the following clarifications before proceeding with the proof:

The update steps for $\mathbf{U}$ and $\mathbf{V}$ are similar to NMF [46]. Considering each element v_{ij} in $\mathbf{V}$, we use F_{ij} to denote the part of $\mathcal{L}$, which is only relevant to v_{ij}. We will obtain that

$$F'_{ij} = (\frac{\partial \mathcal{L}}{\partial \mathbf{V}_{ij}}) = [\sigma(-2\mathbf{X}^T\mathbf{U} + 2\mathbf{V}\mathbf{U}^\mathbf{T}\mathbf{U} + 2\beta(\mathbf{V} - \mathbf{X}^\mathbf{T}\mathbf{U})) + 2\omega\mathbf{L}_s\mathbf{V}]_{ij}, \tag{25}$$

$$F''_{ij} = 2\sigma(\mathbf{U}^\mathbf{T}\mathbf{U})_{ii} + 2\sigma\beta\mathbf{I}_{jj} + 2\omega(\mathbf{L}_s)_{ij}. \tag{26}$$

To prove the convergence of $\mathbf{V}$-subproblem, we need to demonstrate that each F_{ij} remains nonincreasing during the update described in Eq. (17). Next, we will prove the nonincreasing property of F_{ij} by utilizing an auxiliary function [46]. The following two definitions are introduced:

Definition 1. [46] $G(v, v')$ is an auxiliary function of $F(v)$ if the following conditions

$$G(v, v') \geq F(v), \quad G(v, v) = F(v), \tag{27}$$

are satisfied.

Definition 2. [46] If G is an auxiliary function of F, then F is nonincreasing under the update

$$v^{(t+1)} = \arg\min_{v} \ G(v, v^{(t)}). \tag{28}$$

Definition 1 and Definition 2 are utilized to demonstrate that by introducing an appropriate auxiliary function, the update for $\mathbf{V}$ in Eq. (17) can be transformed into the update form of Eq. (28). Additionally, we introduce the following lemma to construct auxiliary function.

Lemma 1. *Function*

$$\begin{aligned} G(v, v_{ij}^{(t)}) = & F_{ij}(v_{ij}^{(t)}) + F'_{ij}(v_{ij}^{(t)})(v - (v_{ij}^{(t)}) \\ & + \frac{[\sigma(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U} + \beta\mathbf{V}) + \omega\mathbf{L}_s\mathbf{V}]_{ij}}{(v_{ij}^{(t)})}(v - (v_{ij}^{(t)})^2, \end{aligned} \tag{29}$$

is an auxiliary function of F_{ij}.

Proof of Lemma 1. It is straightforward to observe that $G(v, v) = F_{ij}(v)$. Before proving $G(v, v_{ij}^{(t)}) \geq F_{ij}(v)$, we need to consider the Taylor series expansion of $F_{ij}(v)$.

$$\begin{aligned} F_{ij}(v) = & F_{ij}(v_{ij}^{(t)}) + F'_{ij}(v_{ij}^{(t)})(v - (v_{ij}^{(t)}) \\ & + [\sigma(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj} + \sigma\beta\mathbf{I}_{jj} + \omega(\mathbf{L}_s)_{ii}](v - (v_{ij}^{(t)})^2. \end{aligned} \tag{30}$$

By Eq. (29) and Eq. (30), we can observe that proving $G(v, v_{ij}^{(t)}) \geq F_{ij}(v)$ is essentially equivalent to proving

$$\frac{\sigma[(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U} + \beta\mathbf{V}) + \omega\mathbf{L}_s\mathbf{V}]_{ij}}{(v_{ij}^{(t)})} \ \geq \ \sigma(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj} + \sigma\beta\mathbf{I}_{jj} + \omega(\mathbf{L}_s)_{ii}. \tag{31}$$

Eq. (31) can be divided into three parts for proof, as follows:

$$\begin{cases} \sigma\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U}_{ij} = \sigma\sum_{l}^{k} v_{il}^{(t)}\mathbf{U}^{\mathbf{T}}\mathbf{U}_{lj} \geq \sigma v_{ij}^{(t)}(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj}, \\ \sigma\beta\mathbf{V}_{ij} = \sigma\beta\sum_{l}^{k} v_{il}^{(t)} \geq \sigma\beta v_{ij}^{(t)}\mathbf{I}_{jj}, \\ \omega\mathbf{L}_s\mathbf{V}_{ij} = \omega\sum_{l}^{q}(\mathbf{L}_s)_{il}v_{lj}^{(t)} \geq \omega(\mathbf{L}_s)_{ii}v_{ij}^{(t)}. \end{cases} \tag{32}$$

In accordance with Eq. (32), the validity of Eq. (31) is established, thereby ensuring $G(v, v_{ij}^{(t)}) \geq F_{ij}(v)$. Furthermore, the above proof satisfies Definition 1, demonstrating that $G(v, v_{ij}^{(t)})$ is an auxiliary function for $F_{ij}(v)$.

Next, we will present the proof of Theorem 1.

Proof of Theorem 1. It is imperative to demonstrate the convergence of each subproblem in Algorithm 1. Next, we present the convergence of each subproblem systematically in the following.

(1). Update $\mathbf{U}$ and $\mathbf{V}$ in Eq. (12) and Eq. (17). We replace $G(v, v_{ij}^{(t)})$ in Eq. (28) with Eq. (29) by introducing the Newton's method [44], and we obtain the following update rule:

$$\begin{aligned} v_{ij}^{(t+1)} &= v_{ij}^{(t)} - v_{ij}^{(t)}\frac{F'_{ij}(v_{ij}^{(t)})}{2[\sigma(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U} + \beta\mathbf{V}) + \omega\mathbf{L}_s\mathbf{V}]_{ij}} \\ &= v_{ij}^{(t)}\frac{[\sigma(1+\beta)\mathbf{X}^{\mathbf{T}}\mathbf{U}]_{ij}}{[\sigma(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U} + \beta\mathbf{V}) + \omega\mathbf{L}_s\mathbf{V}]_{ij}}. \end{aligned} \tag{33}$$

The aforementioned proof satisfies the conditions of Definition 2 and F_{ij} is nonincreasing under the update rule of Eq. (33). A similar conclusion can also be extended to the updating of $\mathbf{U}$ using Eq. (12), indicating that auxiliary function related to the subproblem of

U is also nonincreasing. Additionally, we have made slight adjustments to the update procedures of **U** and **V**, as described in [47], to ensure the convergence of the subproblems of Problem (7) concerning **U** and **V**.

(2). Update **S** in Eq. (18). In order to discuss the update process of **S**, we rewrite $\mathcal{L}$ as

$$\mathcal{L} = \sum_{v=1}^{d} \sqrt{\sum_{i,j}^{n} \|\mathbf{v}_i^v - \mathbf{v}_j^v\|_2^2 s_{ij}} + \alpha \|\mathbf{S}\|_F^2. \tag{34}$$

Next, we proceed to demonstrate that the updated **S** will reduce the objective function value of Eq. (34) until convergence. We assume that after each iteration, **S** becomes $\tilde{\mathbf{S}}$, and then we can obtain the following

$$\sum_v \frac{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 \tilde{s}_{ij}}{2\sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}}} + \alpha \|\tilde{\mathbf{S}}\|_F^2 \le \sum_v \frac{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}}{2\sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}}} + \alpha \|\mathbf{S}\|_F^2. \tag{35}$$

According to [45], we have: $\sqrt{x} - \frac{x}{2\sqrt{y}} \le \sqrt{y} - \frac{y}{2\sqrt{y}}$ for any positive real number x and y. Based on this, we can obtain

$$\begin{aligned} &\sum_v \sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 \tilde{s}_{ij}} - \sum_v \frac{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 \tilde{s}_{ij}}{2\sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}}} \\ \le &\sum_v \sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}} - \sum_v \frac{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}}{2\sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}}}. \end{aligned} \tag{36}$$

Combining Eq. (35) and Eq. (36), we have

$$\sum_v \sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 \tilde{s}_{ij}} + \alpha \|\tilde{\mathbf{S}}\|_F^2 \le \sum_v \sqrt{\sum_{ij}^n \|v_i^v - v_j^v\|_2^2 s_{ij}} + \alpha \|\mathbf{S}\|_F^2, \tag{37}$$

which can demonstrates that the problem (18) is convergent.

(3). Update **F** in Eq. (21). According to [8], considering the update of **F**, we introduce the Hessian matrix of the Eq. (21) as follows,

$$\frac{\partial^2 Tr(\mathbf{F}^T \mathbf{L}_s \mathbf{F})}{\partial \mathbf{F} \mathbf{F}^T} = \mathbf{L}_s + \mathbf{L}_s^T, \tag{38}$$

where $\mathbf{L}_s$ is positive semi-definite, Eq. (21) is also positive semi-definite. Therefore, problem (21) is a convex function, which is convergent.

(4). Update σ and ω in Eq. (22) and Eq. (23). As Eq. (22) and Eq. (23) are both convex functions, and we have provided a closed-form solutions for σ and ω, respectively. Therefore, the subproblems concerning σ and ω of problem (7) are convergent.

In summary, it can be demonstrated that within Algorithm 1, each subproblem converges. Therefore, Problem (7) achieves local minimization.

4. Experiments

In this section, extensive experiments are performed to evaluate the performance of DA^{2}NMF. All the experiments are conducted on Matlab 2016. And the three metrics accuracy (ACC), normalized mutual information (NMI), and Purity are used to measure the clustering performance. ACC represents the proportion of correctly clustered samples, NMI measures the difference between predicted and true labels, while Purity indicates the percentage of correctly clustered samples within the total samples. These metrics range from 0 to 1, with higher values denoting better results. The specific details can be referred to in [17].

4.1. Datasets

The following five classical datasets are used in MVC: **MSRCv1**[1]: The MSRCv1 dataset is an image dataset consisting of 210 scene recognition images, which can be categorized into 7 classes. And each image is described by using 5 distinct image features. **BBCSport**[2]: This dataset consists of 544 documents obtained from the BBCSport website, with a total of 544 instances. It is composed of 2 views and categorized into 5 classes. **3Sources**[3]: Consisting of 169 news documents reported by online news organizations,

[1] http://research.microsoft.com/en-us/projects/objectclassrecognition/.
[2] http://mlg.ucd.ie/datasets/segment.html.
[3] http://mlg.ucd.ie/datasets/3sources.html.

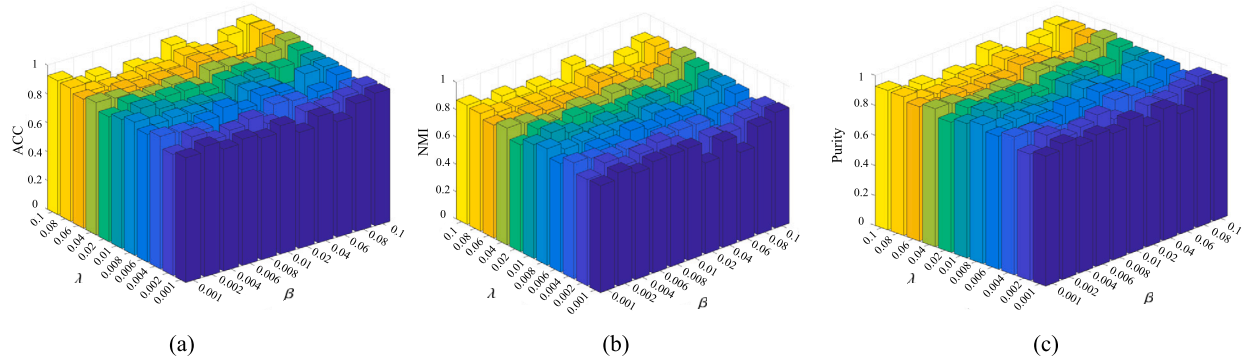


Fig. 2. Parameter analysis of DA^{2}NMF in MSRCv1 dataset. (a), (b), and (c) represent the ACC, NMI, and Purity under different parameter settings.

this dataset comprises 3 views and can be categorized into 6 classes. **HandWritten**[4]: With 2000 data points representing 0–9 digit classes, the HandWritten dataset includes 6 types of features for each data point. **WebKb**[5]: The WebKb dataset comprises a total of 203 web pages, each having 3 distinct views. This dataset can be divided into 4 different categories.

4.2. Comparison methods

To test the performance of the DA^{2}NMF, we compare it against a single-view algorithm and 15 multi-view algorithms. Furthermore, we categorize the 15 MVC methods into four groups: spectral MVC, NMF-based MVC, graph-based MVC, and subspace-based MVC.

A. spectral MVC: The 5 models, namely **Co-Regularized Spectral Clustering (CoReg)**, [12], **Affinity Aggregation for Spectral Clustering (AASC)** [13], **Robust Multi-view Spectral Clustering (RMSC)** [14], **MVC via Adaptively Weighted Procrustes (AWP)** [15], and **Weighted Multi-view Spectral Clustering based on spectral perturbation (WMSC)** [7], are categorized as spectral MVC methods.

B. NMF-based MVC: The NMF-based MVC contains the following, **Multi-View NMF (MultiNMF)** [21] and **Multi-view Concept Clustering (MVCC)** [22].

C. graph-based MVC: The graph-based MVC contains the following, **Graph Learning for MVC (MVGL)** [16], **Multi-view Consensus Graph Clustering (MCGC)** [17], **Graph-based System (GBS)** [18], **Graph-Based MVC (GMC)** [8], and **MVC via Cross-View Graph Diffusion (CGD)** [19].

D. Subspace-based MVC: The subspace-based MVC contains the following, **Consensus Graph Constrained Multi-view Subspace Clustering (CGMSC)** [20] and **Fast Parameter-free Multi-view Subspace Clustering with Consensus Anchor Guidance (FPMVS-CAG)** [9].

4.3. Parameters analysis

In accordance with the details provided in Section 3, the impact of hyperparameters β and λ on clustering performance of DA^{2}NMF model are investigated. Taking the MSRCv1 dataset as an example, we analyze the sensitivity of each hyperparameter considering the range of values: [0.001, 0.002, 0.004, 0.006, 0.008, 0.01, 0.02, 0.04, 0.06, 0.08, 0.1]. Fig. 2 depicts the experimental results for each parameter setting, providing a more comprehensible analysis of parameter sensitivity. From Fig. 2, we can observe that the DA^{2}NMF model is sensitive to both parameters β and λ. Additionally, compared to parameter β, DA^{2}NMF model exhibits higher sensitivity to λ. Based on these observations, we set λ to 0.04 and β to 0.1.

4.4. Comparison with the single-view clustering algorithm

To validate the effectiveness of the DA^{2}NMF method, we conduct experimental analysis on single-view clustering using the aforementioned five datasets. Specifically, we use the DA^{2}NMF method and the traditional NMF-based [48] clustering method separately for each view. To evaluate the significance of auto-weighted strategy, additional experiments are conducted to compare the equal-weighted approach with the auto-weighted method. The clustering performance is presented in Table 2 and Fig. 3, and the best results are bolded in Table 2. Based on our observation of the results, we have the following analyses.

1). Our DA^{2}NMF exhibits strong clustering performance within each view across multiple datasets. Especially in the third view of the HandWritten dataset, DA^{2}NMF surpasses NMF method by 57.8% based on the ACC metric. This is due to the fact that our DA^{2}NMF extracts features from each view by simultaneously considering both linear and nonlinear structures of multi-view data.

[4] https://archive.ics.uci.edu/ml/datasets/Multiple+Features.
[5] https://linqs.soe.ucsc.edu/data.

Table 2
Clustering performance comparison of each view in different datasets.

Method		BBCSport			3Sources			WebKb			MSRCv1			HandWritten		
		ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
NMF-based clustering	View1	45.96	32.47	53.49	35.92	28.85	49.82	64.83	39.32	77.04	37.62	25.27	38.62	35.41	29.70	36.99
	View2	44.22	26.51	52.13	52.07	42.36	60.36	50.79	4.76	54.43	65.00	53.42	65.86	60.70	58.59	62.30
	View3	-	-	-	38.22	31.03	52.31	73.40	46.07	80.30	67.90	61.97	70.43	10.05	3.50	10.45
	View4	-	-	-	-	-	-	-	-	-	44.67	36.63	47.57	29.27	36.09	33.57
	View5	-	-	-	-	-	-	-	-	-	56.71	47.48	59.57	65.33	61.43	67.39
	View6	-	-	-	-	-	-	-	-	-	-	-	-	48.96	44.93	51.43
DA^{2}NMF	View1	77.02	63.24	77.02	36.69	35.46	58.58	71.43	36.85	72.91	41.90	34.26	42.86	58.95	62.82	64.05
	View2	60.11	50.43	72.79	53.25	52.89	72.78	51.72	8.47	54.68	70.48	60.56	70.48	57.10	61.16	63.15
	View3	-	-	-	49.11	41.24	62.72	59.61	9.75	61.08	68.57	58.11	68.57	67.85	60.96	67.85
	View4	-	-	-	-	-	-	-	-	-	53.81	45.78	57.14	32.10	38.03	32.80
	View5	-	-	-	-	-	-	-	-	-	58.57	52.55	61.90	77.25	73.13	77.25
	View6	-	-	-	-	-	-	-	-	-	-	-	-	58.90	61.72	63.65
	Equal	88.79	74.94	88.79	63.91	48.47	69.82	66.01	17.21	67.49	76.19	75.98	79.05	91.10	85.65	91.10
	Auto	**91.54**	**78.58**	**91.54**	**81.66**	**65.64**	**81.66**	**81.28**	**51.82**	**82.27**	**95.24**	**90.29**	**95.24**	**94.95**	**90.62**	**94.65**

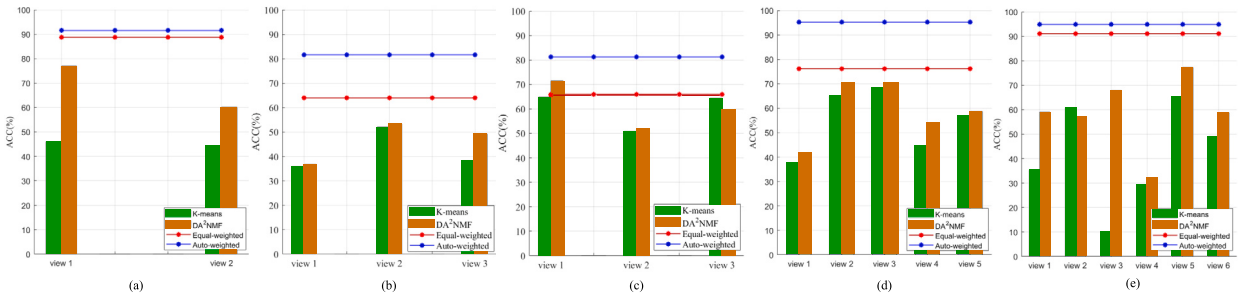


Fig. 3. The ACC of the single-view algorithm on (a) BBCSport, (b) 3Sources, (c) WebKb, (d) MSRCv1, (e) HandWritten.

2). Clustering based solely on individual views fails to leverage the full diversity present in the data. Notably, in View3 of the HandWritten dataset, the traditional NMF method shows poor performance. Recognizing the complementarity of multiple views, we assign equal weights to all views for clustering. It can be observed that our multi-view DA^{2}NMF significantly improves the clustering performance compared to single-view clustering results by integrating features from multiple views.

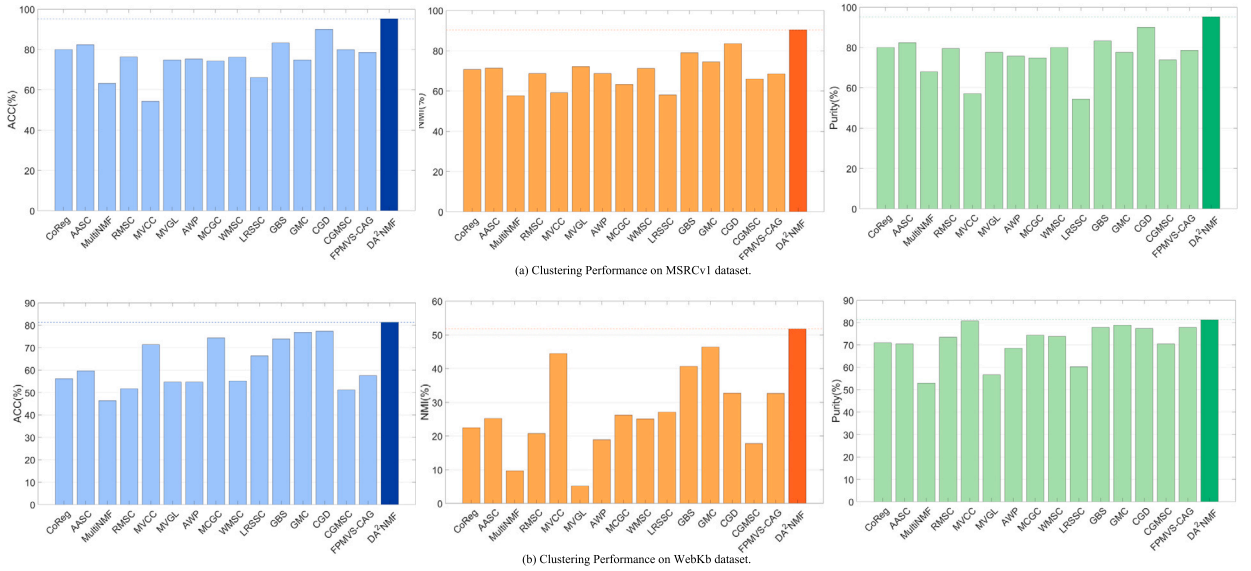
3). While assigning equal weights to various views can leverage complementary information across multiple perspectives, there are cases, such as in the 3Sources dataset, where the clustering performance of using only the third view surpasses the clustering performance of averaging all views. Conversely, the dual auto-weighted DA^{2}NMF outperforms all clustering methods. This demonstrates that our dual auto-weighted DA^{2}NMF adaptively allocates weights based on the importance of different views. Moreover, this method dynamically assigns weights to low-dimensional representations during the graph fusion process, resulting in a more representative graph that ensures consistency across views.

4.5. Comparison with the multi-view clustering algorithms

To further evaluate the effectiveness of the proposed MVC framework DA^{2}NMF, comparative experiments are conducted against other MVC methods. Tables 3–7 display the clustering results of different on MSRCv1, 3Sources, HandWritten, BBCSport, and WebKb datasets, respectively. In each metric, the best clustering result is indicated in bold, while the second best is underlined. From these results, we derive the following observations:

1). From the above Tables 3–7, DA^{2}NMF algorithm has shown promising experimental results across the aforementioned five classical multi-view datasets, outperforming the comparison methods, including spectral clustering, the NMF-based clustering, the graph-based clustering, and the subspace-based clustering. Specifically, compared to NMF-based methods like MultiNMF and MVCC, DA^{2}NMF achieves the highest performance metrics on the datasets mentioned above. This could be attributed to the fact that DA^{2}NMF not only utilizes a decoder for data reconstruction but also incorporates an encoder, which explores and mines the underlying data structure. DA^{2}NMF integrates an encoder and a decoder into a unified framework, enabling comprehensive learning of linear low-dimensional representations from the data.

2). Moreover, DA^{2}NMF surpasses graph learning, particularly DA^{2}NMF outperforms the CGD (the second highest) algorithm, with improvements of 5.24%, 6.81%, and 5.24% in the ACC, NMI, and Purity metrics in MSRCv1 dataset. This could be attributed to the fact that DA^{2}NMF incorporates adaptive graph learning, dynamically constructing a consistent graph based on the structure of the low-dimensional representations. By constructing graph to learn nonlinear relationships, DA^{2}NMF becomes possible to better capture the underlying structure and relational information within the view-specific representations.



(a) Clustering Performance on MSRCv1 dataset.

(b) Clustering Performance on WebKb dataset.

Fig. 4. Comparison results of different methods on (a) MVSRCv1 and (b) WebKb datasets.

Table 3
Clustering performance compared to multi-view algorithms on MSRCv1.

Metric	CoReg	AASC	MultiNMF	RMSC	MVCC	MVGL	AWP	MCGC
ACC(%)	80.48	82.38	66.67	74.29	72.38	74.76	75.24	74.29
NMI(%)	71.62	71.38	58.88	68.72	62.54	72.14	68.70	63.25
Purity(%)	80.48	82.38	66.67	79.52	72.38	77.62	75.71	74.76

Metric	WMSC	LRSSC	GBS	GMC	CGD	CGMSC	FPMVS-CAG	DA²NMF
ACC(%)	76.19	66.12	83.33	74.76	90.00	71.90	78.57	**95.24**
NMI(%)	71.24	58.02	79.06	74.53	83.48	65.96	68.56	**90.29**
Purity(%)	80.00	54.39	83.33	77.62	90.00	73.81	78.57	**95.24**

Table 4
Clustering performance compared to multi-view algorithms on 3Sources.

Metric	CoReg	AASC	MultiNMF	RMSC	MVCC	MVGL	AWP	MCGC
ACC(%)	45.56	37.28	51.01	43.20	72.78	72.78	42.60	74.56
NMI(%)	47.48	30.63	46.14	42.89	68.12	56.17	37.90	58.72
Purity(%)	68.05	50.03	61.84	61.54	73.96	76.33	59.76	78.11

Metric	WMSC	LRSSC	GBS	GMC	CGD	CGMSC	FPMVS-CAG	DA²NMF
ACC(%)	42.60	61.15	69.23	69.23	78.70	66.27	34.32	**81.66**
NMI(%)	41.98	51.70	62.73	62.73	**69.86**	60.13	12.27	65.64
Purity(%)	60.36	61.57	74.56	74.56	**84.02**	76.92	46.15	81.66

Table 5
Clustering performance compared to multi-view algorithms on HandWritten.

Metric	CoReg	AASC	MultiNMF	RMSC	MVCC	MVGL	AWP	MCGC
ACC(%)	92.60	82.35	10.05	19.85	52.15	85.30	87.35	59.15
NMI(%)	89.10	86.50	3.50	21.56	50.29	88.99	88.07	70.30
Purity(%)	92.60	85.75	10.45	28.90	54.35	88.05	87.65	59.30

Metric	WMSC	LRSSC	GBS	GMC	CGD	CGMSC	FPMVS-CAG	DA²NMF
ACC(%)	84.15	75.75	88.10	88.30	85.45	69.00	82.30	**94.95**
NMI(%)	85.96	72.49	90.12	**90.74**	88.77	81.94	79.24	90.62
Purity(%)	86.70	66.21	88.10	88.30	87.90	77.55	82.30	**94.95**

Table 6
Clustering performance compared to multi-view algorithms on BBCSport.

Metric	CoReg	AASC	MultiNMF	RMSC	MVCC	MVGL	AWP	MCGC
ACC(%)	60.48	62.32	63.13	51.47	86.03	72.61	63.42	59.93
NMI(%)	47.12	39.11	47.21	36.09	74.14	70.83	44.26	31.35
Purity(%)	67.28	67.10	67.52	61.40	86.03	76.29	70.40	61.95

Metric	WMSC	LRSSC	GBS	GMC	CGD	CGMSC	FPMVS-CAG	DA^{2}NMF
ACC(%)	58.82	52.60	80.70	80.70	81.70	54.41	49.82	**91.54**
NMI(%)	47.60	36.43	76.10	76.10	**81.07**	32.86	33.18	78.58
Purity(%)	68.57	45.80	84.38	84.38	84.74	54.78	60.11	**91.54**

Table 7
Clustering performance compared to multi-view algorithms on WebKb.

Metric	CoReg	AASC	MultiNMF	RMSC	MVCC	MVGL	AWP	MCGC
ACC(%)	56.16	59.61	46.35	51.72	71.43	54.68	54.68	74.38
NMI(%)	22.46	25.25	9.68	20.77	44.46	5.22	18.85	26.21
Purity(%)	70.94	70.44	52.91	73.40	80.79	56.65	68.47	74.38

Metric	WMSC	LRSSC	GBS	GMC	CGD	CGMSC	FPMVS-CAG	DA^{2}NMF
ACC(%)	55.17	66.38	73.89	76.85	77.36	51.23	57.64	**81.28**
NMI(%)	25.04	27.09	40.65	46.43	32.70	17.75	32.68	**51.82**
Purity(%)	73.89	60.20	77.83	78.82	77.34	70.44	77.83	**81.28**

3). DA^{2}NMF outperforms auto-weighted methods such as GMC. This could be attributed to the fact that GMC only utilizes an auto-weighted allocation, whereas DA^{2}NMF employs a novel approach of dual auto-weighted strategy, resulting in a more representative fusion graph.

Furthermore, to facilitate a more visual comparison, we present bar charts, as illustrated in Fig. 4. By observing the Fig. 4, we can derive the same results as those shown in the Tables 3–7, indicating that our DA^{2}NMF outperforms the comparison methods.

4.6. Visualization analysis

To illustrate the effectiveness of DA^{2}NMF to consolidate the structural information across diverse views, the standardized t-SNE tool [49] is utilized to visualize the original data matrices of each view and the learned consistent graph of the MSRCv1 dataset. The visualization results are depicted in Fig. 5, where data points of the same color denote samples from identical categories. In Fig. 5, for view1, view2, view3, view4, and view5, the classes are intermingled, making it challenging to discern the category for each data point. However, in the visualizations of the DA^{2}NMF, clear class boundaries and distinctive separations are observed, indicating the framework's capacity to effectively grasp the underlying data structures from diverse perspectives. This ability facilitates the fusion of multi-view information, thereby promoting better clustering.

Furthermore, as depicted in Fig. 6, we present confusion matrices to illustrate the clustering outcomes of DA^{2}NMF and other MVC methods applied to the MSRCv1 dataset. A clearer block-diagonal structure in the confusion matrix signifies superior clustering performance. For a visually intuitive display, areas with notably higher error rates are marked using red circles. It's apparent that, compared to the other methods, our DA^{2}NMF distinctly shows a clear block-diagonal pattern in the confusion matrix. Additionally, DA^{2}NMF does not misassign the second class to different clusters, which is common in most comparison methods. Overall, compared to other MVC algorithms, DA^{2}NMF integrates multi-view information, resulting in a significantly improved clustering performance.

5. Discussion

To individually evaluate the impacts of autoencoder-like NMF, adaptive graph learning, and dual-weighted strategy, we conduct several ablation experiments. Additionally, for further validation, we create a baseline model by removing autoencoder-like module, adaptive graph learning module, and dual auto-weighted strategy from DA^{2}NMF, denoting it as Multi-view NMF (MvNMF). The settings for ablation experiments are as follows: 1) Removal of adaptive graph learning module and dual auto-weighted strategy, named as Multi-view Autoencoder-like NMF (MvANMF). 2) Removal of autoencoder-like NMF and dual auto-weighted strategy, named as Multi-view Graph-based NMF (MvGNMF). 3) Removal of dual auto-weighted strategy, named as Multi-view Autoencoder-like NMF with Graph learning (MvAGNMF). 4) Removal of auto-weight ω, named as A^{2}NMF-ω. 5) Removal of auto-weight σ, named as A^{2}NMF-σ. The specific results of the ablation experiments are presented in Table 8. According to Table 8, we have the following observations and analyses:

1. It is evident that autoencoder-like NMF, adaptive graph learning, and the dual auto-weighted strategy effectively enhance the performance of the DA^{2}NMF. Specifically, the experimental performance of MvANMF is superior to that of MvNMF on both

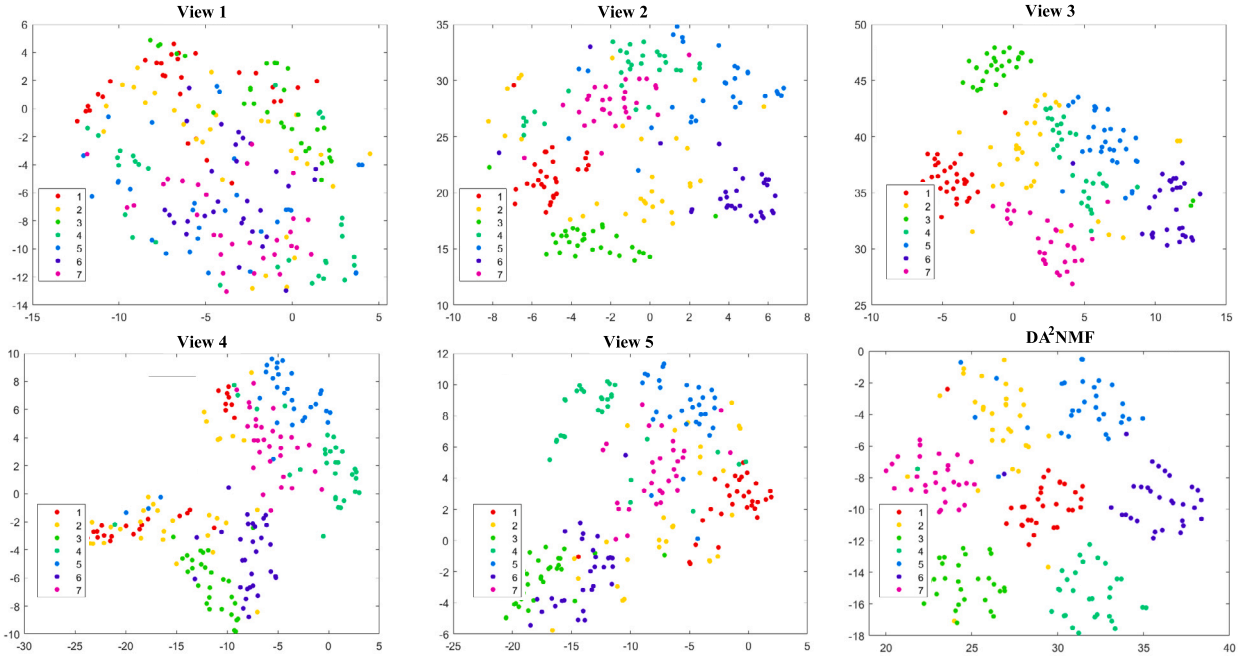


Fig. 5. Visualization of the different view and the consistent graph in the MSRCv1 dataset, where each category is represented by a distinct color.

Table 8
Comparison between DA^{2}NMF and ablative methods.

Dataset	Methods	ACC	NMI	Purity
MSRCv1	MvNMF	63.19	57.65	67.93
	MvANMF	83.81	72.49	83.81
	MvGNMF	89.52	82.20	89.52
	MvAGNMF	91.43	83.83	91.43
	A^{2}NMF-ω	93.33	87.27	93.33
	A^{2}NMF-σ	92.38	86.43	92.38
	DA^{2}NMF	**95.26**	**90.29**	**95.24**
WebKb	MvNMF	46.35	9.68	52.91
	MvANMF	49.26	10.30	53.69
	MvGNMF	53.69	35.66	77.36
	MvAGNMF	65.52	17.87	66.50
	A^{2}NMF-ω	67.00	34.44	74.88
	A^{2}NMF-σ	70.44	36.81	73.89
	DA^{2}NMF	**81.28**	**51.82**	**81.28**

datasets. Furthermore, MvANMF achieves a significant improvement of 20.62% in ACC compared to the MvNMF on the MSRCv1 dataset. In comparison to MvNMF, MvANMF incorporates an additional encoder to learn low-dimensional representations. This further emphasizes the effectiveness of the autoencoder-like module, demonstrating that the comprehensive integration of encoder and decoder is beneficial for obtaining linear view-specific low-dimensional representations.

2. Compared to MvNMF, MvGNMF exhibits a 26.33% improvement in ACC metric. Similar effects are observed on the WebKb dataset as well. This shows the significance of adaptive graph learning. By capturing the nonlinear structures between data, a satisfied clustering result can be obtained.

3. In both MSRCv1 and WebKb datasets, adaptively changing the weights of σ and ω leads to improvements in the experimental results. Specifically, on the MSRCv1 dataset, A^{2}NMF-ω yields better experimental results compared to A^{2}NMF-σ. However, the opposite conclusion is observed in the WebKb dataset. This indicates that simultaneously adjusting the dual weights is necessary. Therefore, the dual auto-weighted strategy employed in DA^{2}NMF has proven to be effective.

6. Conclusion

In this work, we proposed a Dual Auto-weighted multi-view clustering model by Autoencoder-like NMF (DA^{2}NMF), which can fully explore the linear and nonlinear structures of data. Specifically, autoencoder-like NMF learns latent representations and reconstructs the multi-view data simultaneously. Furthermore, adaptive graph learning comprehensively explores the nonlinear structures

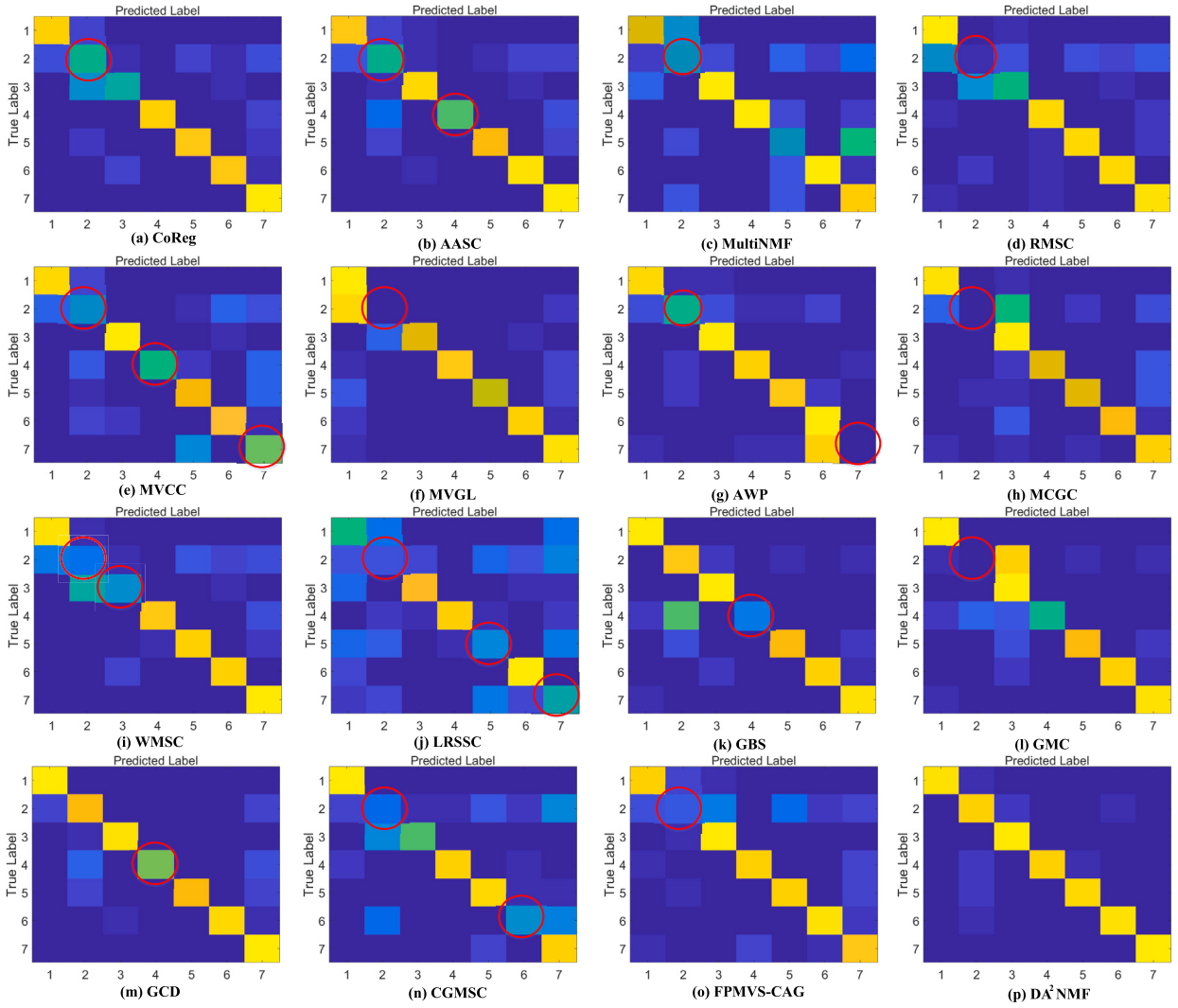


Fig. 6. Visualization of the confusion matrices in the MSRCv1 dataset.

within the view-specific representations. By designing a dual auto-weighted strategy, DA^{2}NMF can highlight the significance of different views and the contribution of each low-dimensional representation. An iterative algorithm based on MUR is developed to solve the optimization problem with a theoretical convergence guarantee. Experimental results demonstrate that the proposed DA^{2}NMF outperforms the state-of-the-art multi-view algorithms.

For MVC, as the DA^{2}NMF involves multiple variables, one remaining challenge is to effectively perform clustering on large-scale datasets. Moreover, since the proposed method focuses on handling complete multi-view datasets, it may have limitations when extended to the incomplete MVC tasks. We will attempt to address the remaining challenges in our future work.

CRediT authorship contribution statement

Si-Jia Xiang: Writing – original draft, Methodology, Formal analysis, Conceptualization. **Heng-Chao Li:** Writing – review & editing, Validation, Supervision. **Jing-Hua Yang:** Writing – review & editing, Validation, Supervision, Methodology. **Xin-Ru Feng:** Writing – review & editing, Validation, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62271418, and in part by the Natural Science Foundation of Sichuan Province under Grants 2023NSFSC0030 and 2024NSFSC7075, Postdoctoral Fellowship Program of CPSF under Grant GZC20232198.

References

[1] Z. Cao, X. Xie, Y. Li, Multi-view unsupervised feature selection with consensus partition and diverse graph, Inf. Sci. 661 (2024) 120178.
[2] G. Haseli, R. Ranjbarzadeh, M. Hajiaghaei-Keshteli, S.J. Ghoushchi, A. Hasani, M. Deveci, W. Ding, HECON: weight assessment of the product loyalty criteria considering the customer decision's halo effect using the convolutional neural networks, Inf. Sci. 623 (2023) 184–205.
[3] Y.B. Özçelik, A. Altan, Overcoming nonlinear dynamics in diabetic retinopathy classification: a robust AI-based model with chaotic swarm intelligence optimization and recurrent long short-term memory, Fractal Fract. 7 (8) (2023) 598.
[4] R. Ranjbarzadeh, S.J. Ghoushchi, S. Anari, S. Safavi, N.T. Sarshar, E.B. Tirkolaee, M. Bendechache, A deep learning approach for robust, multi-oriented, and curved text detection, Cogn. Comput. (2022) 1–13.
[5] J. Li, X. Zhang, J. Wang, X. Wang, Z. Tan, H. Sun, Projection-based coupled tensor learning for robust multi-view clustering, Inf. Sci. 632 (2023) 664–677.
[6] M. Yin, X. Liu, L. Wang, G. He, Learning latent embedding via weighted projection matrix alignment for incomplete multi-view clustering, Inf. Sci. 634 (2023) 244–258.
[7] L. Zong, X. Zhang, X. Liu, H. Yu, Weighted multi-view spectral clustering based on spectral perturbation, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 4621–4629.
[8] H. Wang, Y. Yang, B. Liu, GMC: graph-based multi-view clustering, IEEE Trans. Knowl. Data Eng. 32 (2020) 1116–1129.
[9] S. Wang, X. Liu, X. Zhu, P. Zhang, Y. Zhang, F. Gao, E. Zhu, Fast parameter-free multi-view subspace clustering with consensus anchor guidance, IEEE Trans. Image Process. 31 (2021) 556–568.
[10] J. Yang, C. Chen, H. Dai, L. Fu, Z. Zheng, A structure noise-aware tensor dictionary learning method for high-dimensional data clustering, Inf. Sci. 612 (2022) 87–106.
[11] M. Liu, Y. Wang, V. Palade, Z. Ji, Multi-view subspace clustering network with block diagonal and diverse representation, Inf. Sci. 626 (2023) 149–165.
[12] H. Xu, X. Zhang, W. Xia, Q. Gao, X. Gao, Low-rank tensor constrained co-regularized multi-view spectral clustering, Neural Netw. 132 (2020) 245–252.
[13] H. Huang, Y. Chuang, C. Chen, Affinity aggregation for spectral clustering, in: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 773–780.
[14] R. Xia, Y. Pan, L. Du, J. Yin, Robust multi-view spectral clustering via low-rank and sparse decomposition, in: Proceedings of the 28th AAAI Conference on Artificial Intelligence, 2014, pp. 2149–2155.
[15] F. Nie, L. Tian, X. Li, Multiview clustering via adaptively weighted procrustes, in: Proceedings of the 24th ACM SIGKDD International Conference on Edge Discovery and Data Mining, 2018, pp. 2022–2030.
[16] K. Zhan, C. Zhang, J. Guan, J. Wang, Graph learning for multiview clustering, IEEE Trans. Cybern. 48 (2018) 2887–2895.
[17] K. Zhan, F. Nie, J. Wang, Y. Yang, Multiview consensus graph clustering, IEEE Trans. Image Process. 28 (2019) 1261–1270.
[18] H. Wang, Y. Yang, B. Liu, H. Fujita, A study of graph-based system for multi-view clustering, Knowl.-Based Syst. 163 (2019) 1009–1019.
[19] C. Tang, X. Liu, X. Zhu, E. Zhu, Z. Luo, L. Wang, W. Gao, CGD: multi-view clustering via cross-view graph diffusion, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020, pp. 5924–5931.
[20] X. Liu, G. Pan, M. Xie, Multi-view subspace clustering with adaptive locally consistent graph regularization, Neural Comput. Appl. 33 (2021) 15397–15412.
[21] J. Gao, J. Han, J. Liu, C. Wang, Multi-view clustering via joint nonnegative matrix factorization, in: Proceedings of the 13th SIAM International Conference on Data Mining, 2013, pp. 252–260.
[22] H. Wang, Y. Yang, T. Li, Multi-view clustering via concept factorization with local manifold regularization, in: Proceedings of the IEEE 16th International Conference on Data Mining, 2016, pp. 1245–1250.
[23] X.-R. Feng, H.-C. Li, R. Wang, Q. Du, X. Jia, A. Plaza, Hyperspectral unmixing based on nonnegative matrix factorization: a comprehensive review, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 15 (2022) 4414–4436.
[24] S. Wang, L. Chen, N. Zheng, L. Li, F. Peng, J. Lu, Shared and individual representation learning with feature diversity for deep multiview clustering, Inf. Sci. 647 (2023) 119426.
[25] S.Z. Selim, M.A. Ismail, K-means-type algorithms: a generalized convergence theorem and characterization of local optimality, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1984) 81–87.
[26] S. Peng, J. Yin, Z. Yang, B. Chen, Z. Lin, Multiview clustering via hypergraph induced semi-supervised symmetric nonnegative matrix factorization, IEEE Trans. Circuits Syst. Video Technol. 33 (2023) 5510–5524.
[27] C.F. Yang, M. Ye, J. Zhao, Document clustering based on nonnegative sparse matrix factorization, in: Proceedings of the International Conference on Computing, Networking and Communications, vol. 3611, 2005, pp. 557–563.
[28] S.M. Wild, J. Curry, A. Dougherty, Improving non-negative matrix factorizations through structured initialization, Pattern Recognit. 37 (11) (2004) 2217–2232.
[29] D. Kuang, H. Park, C.H.Q. Ding, Symmetric nonnegative matrix factorization for graph clustering, in: Proceedings of the 12th SIAM International Conference on Data Mining, 2012, pp. 106–117.
[30] J. Liu, Y. Jiang, Z. Li, Z. Zhou, H. Lu, Partially shared latent factor learning with multiview data, IEEE Trans. Neural Netw. Learn. Syst. 26 (2015) 1233–1246.
[31] H. Cai, B. Liu, Y. Xiao, L. Lin, Semi-supervised multi-view clustering based on orthonormality-constrained nonnegative matrix factorization, Inf. Sci. 536 (2020) 171–184.
[32] X. Zhang, H. Gao, G. Li, J. Zhao, J. Huo, J. Yin, Y. Liu, L. Zheng, Multi-view clustering based on graph-regularized nonnegative matrix factorization for object recognition, Inf. Sci. 432 (2018) 463–478.
[33] X. Liu, S. Ding, X. Xu, L. Wang, Deep manifold regularized semi-nonnegative matrix factorization for multi-view clustering, Appl. Soft Comput. 132 (2023) 109806.
[34] Z. Kang, C. Peng, Q. Cheng, Twin learning for similarity and clustering: a unified kernel approach, in: Proceedings of the 31st AAAI Conference on Artificial Intelligence, 2017, pp. 2080–2086.
[35] Y. Zhang, W. Quan, T. Akutsu, L. Liu, H. Cai, B. Zhang, Accurate multi-view clustering to seek the cross-viewed yet uniform sample assignment via tensor feature matching, Inf. Sci. 664 (2024) 120305.
[36] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (2007) 395–416.

[37] R. Wang, H. Chen, Y. Lu, Q. Zhang, F. Nie, X. Li, Discrete and balanced spectral clustering with scalability, IEEE Trans. Pattern Anal. Mach. Intell. 45 (12) (2023) 14321–14336.
[38] W. Shao, L. He, P.S. Yu, Multiple incomplete views clustering via weighted nonnegative matrix factorization with $l_{2,1}$ regularization, in: Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2015, pp. 318–334.
[39] K. Luong, R. Nayak, T. Balasubramaniam, M.A. Bashar, Multi-layer manifold learning for deep non-negative matrix factorization-based multi-view clustering, Pattern Recognit. 131 (2022) 108815.
[40] X.-R. Feng, H.-C. Li, S. Liu, H. Zhang, Correntropy-based autoencoder-like NMF with total variation for hyperspectral unmixing, IEEE Geosci. Remote Sens. Lett. 19 (2022) 1–5.
[41] T. Chen, Y. Zeng, H. Yuan, G. Zhong, L.L. Lai, Y.Y. Tang, Multi-level regularization-based unsupervised multi-view feature selection with adaptive graph learning, Int. J. Mach. Learn. Cybern. 14 (5) (2023) 1695–1709.
[42] C. Chen, H. Qian, W. Chen, Z. Zheng, H. Zhu, Auto-weighted multi-view constrained spectral clustering, Neurocomputing 366 (2019) 1–11.
[43] K. Fan, On a theorem of Weyl concerning eigenvalues of linear transformations: II, Proc. Natl. Acad. Sci. USA 36 (1) (1949) 31–35.
[44] S.P. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University, 2004.
[45] F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in: Proceedings of the 31st AAAI Conference on Artificial Intelligence, 2017, pp. 2408–2414.
[46] D.D. Lee, H.S. Seung, Algorithms for non-negative matrix factorization, in: Proceedings of the Neural Information Processing Systems 13, 2000, pp. 556–562.
[47] C. Lin, Convergence of multiplicative update algorithms for non-negative matrix factorization, IEEE Trans. Neural Netw. 18 (2007) 1589–1596.
[48] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.
[49] L. van der Maaten, G.E. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (2008) 2579–2605.