

Article

Multiview Data Clustering with Similarity Graph Learning Guided Unsupervised Feature Selection

Ni Li ^{1,*}, Manman Peng ^{2,†} and Qiang Wu ^{2,†}

¹ College of Information and Electronic Engineering, Hunan City University, Yiyang 413000, China

² College of Information and Engineer, Hunan University, Changsha 410082, China;

pengmanman@hnu.edu.cn (M.P.); wuqiang@hnu.edu.cn (Q.W.)

* Correspondence: lini@hncu.edu.cn; Tel.: +86-159-0737-8098

† These authors contributed equally to this work.

Abstract: In multiview data clustering, consistent or complementary information in the multiview data can achieve better clustering results. However, the high dimensions, lack of labeling, and redundancy of multiview data certainly affect the clustering effect, posing a challenge to multiview clustering. A clustering algorithm based on multiview feature selection clustering (MFSC), which combines similarity graph learning and unsupervised feature selection, is designed in this study. During the MFSC implementation, local manifold regularization is integrated into similarity graph learning, with the clustering label of similarity graph learning as the standard for unsupervised feature selection. MFSC can retain the characteristics of the clustering label on the premise of maintaining the manifold structure of multiview data. The algorithm is systematically evaluated using benchmark multiview and simulated data. The clustering experiment results prove that the MFSC algorithm is more effective than the traditional algorithm.

Keywords: multiview data clustering; unsupervised feature selection; similarity graph



Citation: Li, N.; Peng, M.; Wu, Q. Multiview Data Clustering with Similarity Graph Learning Guided Unsupervised Feature Selection. *Entropy* **2023**, *25*, 1606. <https://doi.org/10.3390/e25121606>

Academic Editors: Francisco J. Gallegos-Funes and Alberto J. Rosales Silva

Received: 25 October 2023
Revised: 18 November 2023
Accepted: 28 November 2023
Published: 30 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Various application types correspond to various network attributes that describe individuals and groups from different perspectives. These networks are represented as multiview feature spaces. For example, when uploading photos to Flickr, users are required to offer labels and related text. In other words, photos can be represented by three view feature spaces: photo content, label, and text description spaces.

Multiview data can integrate these view spaces and use correlation to obtain more accurate network representations. Currently, multiview data are usually described in the form of graphs, such as Gaussian function graphs, k nearest neighbor graphs [1], and graphs based on subspace clustering [2,3]. For the selection of the correct neighborhood size and the processing of points near the intersection of the subspace, subspace clustering based on self-representation is superior to other graph-based representation methods. Nie et al. developed a multiview clustering [4,5] algorithm that can perform spectral clustering of an information network of multiple views by constructing a multiview similarity matrix. The multiview clustering algorithm [6] proposed by Bickel et al. uses spherical k -means multiview clustering. Pu et al. advanced the multiview clustering algorithm [7] based on matrix decomposition, which regularizes the similarity matrix using multiview manifold regularization, to merge the inherent and nonlinear structure of the network in every view. The aforementioned methods provide an idea regarding the relationships between multiview data that improve clustering performance [8] by constructing multiview similarity matrix clustering. However, the redundancy of multiview data has not yet been resolved. In addition, the calculation for constructing a multiview similarity matrix is complicated and unsuitable for large-scale multiview data.

Feature selection [9] obtains the low-dimensional feature subspace representation of the network by selecting features as well as removing noisy, irrelevant, and redundant features to preserve the inherent data structure. This is an effective method for handling large-scale high-dimensional networks. Most existing feature selection methods are based on single-view networks. Recently, the focus of unsupervised feature selection research has been on the study of multiview data. Zhang et al. [10] propose a formulation that learns an adaptive neighbor graph for unsupervised multiview feature selection. This formulation collaborates multiview features and discriminates between different views. Fang et al. [11] propose a novel approach that incorporates both cluster structure and a similarity graph. Their method utilizes multiview feature selection and an orthogonal decomposition technique, which breaks down each objective matrix into a base matrix and a clustering index matrix for each view. Cao et al. [12] present a cluster learning guided multiview unsupervised feature selection, which unified subspace learning, cluster learning, and feature selection into a framework. Tang et al. [13] propose a feature selection method based on multiview data that aims to maintain diversity and enhance consensus learning by utilizing cross-view local structures. Liu et al. [14] propose a framework for guided unsupervised feature selection, which utilizes consensus clustering to generate pseudo cluster indexes for the purpose of feature selection.

There are two modes of feature selection in multiview networks. One is the serial mode, which is a feature selection method that serializes the connection multiview feature space into a feature space and then selects the features. The other is the parallel mode, which involves performing traditional feature selection on each view simultaneously. In more detail, the serial mode ignores the differences between heterogeneous feature spaces, so its performance is relatively poor. The parallel mode considers the correlation between multiple view spaces with relatively better performance. Research on the unsupervised feature selection of multiview data without labels poses a significant challenge. For the traditional unsupervised feature selection method, the feature distribution selected by the Laplacian score [15] method agrees with the sample distribution, which can perform a good regional classification and reflect the inherent manifold structure of data. However, the correlation between the features is not evaluated, resulting in the selection of redundant features. In the MFSC method, spectral analysis retains the internal structure and $L_{2,1}$ uses feature selection coefficients to select the best features. Therefore, the selected features retain the clustering structure of the data.

The MFSC algorithm proposed in this study makes the following contributions:

1. Compared with a single-view dataset that concatenates multiview data, the parallel use of multiview datasets from real-world social media sites significantly improves the accuracy of data representation.
2. In integrated subspace clustering and feature selection, the clustering label and representative coefficient matrix are flow regularizations. Furthermore, to obtain a more suitable feature selection matrix, the a priori of the manifold structure is embedded in the feature selection model.
3. In the construction of the parallel mode multiview feature selection algorithm, noisy, irrelevant, and redundant features are removed to preserve the inherent data structure and improve the efficiency and quality of feature selection based on clustering, which is more suitable for multiview data.

The rest of this paper is organized as follows. Section 2 introduces the basic studies related to the MFSC algorithm. Section 3 presents the MFSC model and its optimization iterative process in detail, and it theoretically proves the convergence and complexity of the algorithm. Section 4 reports the parameter sensitivity and performance analysis of MFSC on typical datasets, as well as the results of comparison experiments with some single-view or multiview feature selections. Section 5 presents the results of this study and the future work.

2. Related Studies

2.1. Multiview Subspace Representation

Let $X^{(v)}$ be the data sample node of the v -th view and $S^{(v)}$ be its representative coefficient matrix. Each data point in the subspace union can be reconstructed effectively by combining the other points in the dataset. Given the data X based on the group effect [16], for representation coefficients S_i and S_j of the samples, X_i and X_j are similar and so are S_i and S_j . The multiview representation of traditional sparse subspace clustering (MVSC) [17] is defined as follows:

$$\|X^{(v)} - X^{(v)}S^{(v)}\|_F^2, s.t. S^{(v)}I = I, S^{(v)}(i, i) = 0. \quad (1)$$

MVSC can well capture the self-representation matrix in the multisubspace k -nearest neighbor graph structure. Similar structure graph $Z^{(v)} = \frac{(S^{(v)})^T + S^{(v)}}{2}$ of the v -th view can learn the multisubspace structure when there are noise, abnormal values, and damaged or missing entries in the data.

2.2. Multiview Unsupervised Feature Selection

Most existing studies on multiview learning [18] assume that all views share the same label space and that these views are related to each other through the label space. It is well known that the main difficulty of unsupervised feature selection is the lack of class tags. Consequently, the concept of a pseudo-class label is introduced to guide the development of the framework using the relationship between views, which is defined as follows:

$$\|(X^{(v)})^T W^{(v)} - C\|_F^2 + \alpha \|W^{(v)}\|_{2,1}, \quad (2)$$

where the v -th view has a mapping matrix $W^{(v)}$ that assigns the pseudo-class label C to the data points. Based on the assumption that the view is associated with the shared label space, each pseudo-class label allocation matrix $(X^{(v)})^T W^{(v)}$ is approximated such that it is close to the pseudo-class label matrix. The $l_{2,1}$ norm [19] is added to W_i to ensure sparseness in the W_i row and feature selection. In addition, the $l_{2,1}$ norm is convex, making the optimization easier.

2.3. Multiview Manifold Structure

The greater the similarity value of the two data points, the more similar the clusters. A similar structure graph with k unconnected cluster subspaces can be directly learned and it is defined as follows:

$$\sum_{i,j=1}^n \|C_{i,:} - C_{j,:}\|_F^2 S_{ij}^{(v)} = \text{tr}(C^T L^{(v)} C), \quad (3)$$

where clustering label $C \in R^{n \times k}$ and Laplacian matrix $L^{(v)} = D^{(v)} - \frac{(S^{(v)})^T + S^{(v)}}{2}$. It is known that MHOAR [20] points out that the properties of the L matrix of nonnegative matrix S are shown in Theorem 1.

Theorem 1. *The number of the eigenvalues 0 of normalized L is equal to the number of connected subspaces of S . Therefore, $\text{rank}(L) = n - k$. According to the Ky Fan theorem [21], using $\sigma_i(L)$ to represent the i -th smallest eigenvalue of L , then $\sigma_i(L) \geq 0$ and $\text{rank}(L) = n - k$. Therefore, $\sum_{i=1}^k \sigma_i(L) = \arg \min_{C \in R^{n \times k}, C^T C = I_k} \text{tr}(C^T L C)$.*

3. Proposed Model

This section contains an introduction to the MFSC model: an explanation of the iterative optimization implementation process, algorithm, proof of convergence, and analysis of algorithm complexity. An illustration of the MFSC model is shown in Figure 1. Multiview unsupervised feature selection, similarity graph learning, and clustering index learning are achieved in the parallel mode. MFSC reduces the redundancy and irrelevant influence of

multiview data and uses the clustering index as the feature selection standard to ensure that the clustering structure remains unchanged.

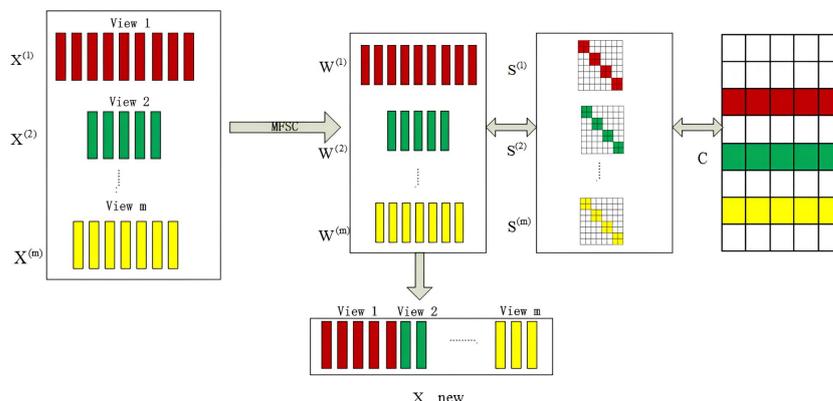


Figure 1. Overall framework based on the parallel mode in MFSC.

3.1. MFSC Model

Suppose the dataset $X = \{X^{(v)} \in R^{d^{(v)} \times n}\}_{v=1}^m$ denotes the data of the v -th view, $d^{(v)}$ denotes the feature number of the v -th view, and n denotes the number of data. The feature selection matrix is $W^{(v)} \in R^{d^{(v)} \times k}$, the clustering label is $C \in R^{n \times k}$, and the subspace representation coefficient $S^{(v)} \in R^{n \times n}$, where k denotes the cluster number. The MFSC model is defined as follows:

$$\begin{aligned} \arg \min_{S^{(v)}, C, W^{(v)}} & \sum_{v=1}^m u^{(v)} (\|X^{(v)T}W^{(v)} - C\|_F^2 \\ & + \|X^{(v)} - X^{(v)}S^{(v)}\|_F^2 + \alpha \text{tr}(C^T L^{(v)}C) \\ & + \beta \|W^{(v)}\|_{2,1}) \end{aligned} \tag{4}$$

$$s.t. C^T C = I_k, S^{(v)} I = I, S^{(v)}(i, i) = 0,$$

where $L^{(v)} = D^{(v)} - \frac{(S^{(v)})^T + S^{(v)}}{2}$.

The model independently learns the $S^{(v)}$ of each view instead of directly using the $S^{(v)}$ calculated by the kernel function. Using the similarity graph of self-representation learning based on the manifold structure, the multisubspace structure of the data can be effectively reflected. By integrating subspace similarity graph learning and feature selection, the pseudo-class label C can capture the relationship between the views to obtain a robust and clean pseudo-class label. Row sparsity is achieved by applying the $l_{2,1}$ [22] constraint to $W^{(v)}$. Figure 1 shows the feature selection based on the parallel mode that iteratively updates the similarity matrix $\{S^{(v)}\}_{v=1}^m$, the feature selection matrix $\{W^{(v)}\}_{v=1}^m$, and the pseudo label matrix C .

3.2. Optimization Calculation Process and Algorithm Representation

This section first introduces the effective implementation of the iterative method to solve the optimization calculation in Equation (4). In the implementation process, $W^{(v)}$, $S^{(v)}$, and C are updated iteratively to obtain the specific implementation process of the MFSC algorithm.

Update $W^{(v)}$:

To effectively calculate the feature selection matrix $W^{(v)}$, irrelevant items $S^{(v)}$ and C are fixed. The objection equation can be rewritten as follows:

$$J1(W^{(v)}) = \|(X^{(v)T}W^{(v)} - C)\|_F^2 + \beta \|W^{(v)}\|_{2,1}. \tag{5}$$

Given that this equation is nondifferentiable [19], the equation is transformed into:

$$J1(W^{(v)}) = \|(X^{(v)})^T W^{(v)} - C\|_F^2 + \beta \text{tr}((W^{(v)})^T D^{(v)} W^{(v)}), \tag{6}$$

where $D^{(v)}$ denotes a diagonal matrix and the j -th diagonal element is $D^{(v)}(j, j) = \frac{1}{2\|W_j^{(v)}\|_2}$.

Calculation process:

$$\begin{aligned} \frac{\partial J1(W^{(v)})}{\partial W^{(v)}} &= 2X^{(v)}(X^{(v)})^T W^{(v)} - 2X^{(v)}C \\ &+ 2\beta D^{(v)}W^{(v)} = 0, \end{aligned} \tag{7}$$

The updated rules for $W^{(v)}$ are as follows:

$$W^{(v)} = (X^{(v)}(X^{(v)})^T + \beta D^{(v)})^{-1} X^{(v)}C. \tag{8}$$

Update $S^{(v)}$:

Theorem 2. Given $X = XZ, W = \frac{Z+Z^T}{2}$, and $L = D - W$, then

$$\text{tr}(F^T L F) = \frac{1}{2} \text{tr}(W P), \tag{9}$$

with $P_{ij} = \|f_i - f_j\|_2^2$, where f_i is the i -th row vector of matrix F .

To effectively calculate the clustering label C , irrelevant items $W^{(v)}$ and $S^{(V)}$ are fixed. The objection equation can be rewritten as follows:

$$\begin{aligned} J2(S^{(v)}) &= \|X^{(v)} - X^{(v)}S^{(v)}\|_F^2 + \alpha \text{tr}(C^T L^{(v)} C) \\ &+ \rho \|I^T - I^T S^{(v)}\|_F^2 \\ \text{s.t. } &S^{(v)}(i, i) = 0. \end{aligned} \tag{10}$$

Based on the properties of the matrix trace $\text{tr}(X^T Y) = \text{tr}(X Y^T)$ and Theorem 2, it is known that P is a symmetric matrix; then,

$$\begin{aligned} \text{tr}(C^T L^{(v)} C) &= \frac{1}{2} \text{tr}(W P) \\ &= \frac{1}{2} \left(\frac{1}{2} \text{tr}(S^{(v)} P^T) + \frac{1}{2} \text{tr}((S^{(v)})^T P) \right) \\ &= \frac{1}{2} \text{tr}((S^{(v)})^T P), \end{aligned} \tag{11}$$

where $P_{ij} = \|C_i - C_j\|_2^2$. According to

$$\begin{aligned} &\|X^{(v)} - X^{(v)}S^{(v)}\|_F^2 + \rho \|I^T - I^T S^{(v)}\|_F^2 \\ &= \|[(X^{(v)})^T, p * I]^T - [(X^{(v)})^T, p * I]^T S^{(v)}\|_F^2, \end{aligned} \tag{12}$$

suppose $X_1^{(v)} = [(X^{(v)})^T, p * I]^T$, so $J2(S^{(v)})$ is equivalently expressed as follows:

$$\begin{aligned} J2(S^{(v)}) &= \|X_1^{(v)} - X_1^{(v)}S^{(v)}\|_F^2 + \frac{\alpha}{2} \text{tr}((S^{(v)})^T P) \\ \text{s.t. } &S^{(v)}(i, i) = 0. \end{aligned} \tag{13}$$

Given that $tr((S^{(v)})^T P) = \sum_i S^{(v)}(i,:)P(:,i)$, where $S^{(v)}(i,:)$ denotes the i -th row vector of $S^{(v)}$ and $P(:,i)$ denotes the i -th column vector of P . Then,

$$\begin{aligned} & \|X_1^{(v)} - X_1^{(v)}S^{(v)}\|_F^2 = \\ & tr(X_1^{(v)} - X_1^{(v)}S^{(v)})^T(X_1^{(v)} - X_1^{(v)}S^{(v)}) \\ & = \sum_i (X_1^{(v)}(i,:) - X_1^{(v)}S^{(v)}(i,:)) \\ & (X_1^{(v)}(i,:) - X_1^{(v)}S^{(v)}(i,:))^T. \end{aligned} \tag{14}$$

Suppose $X_s^{(v)} = X_1^{(v)} - (X_1^{(v)}S^{(v)} - X_1^{(v)}(:,i)S^{(v)}(i,:))$; then,

$$\begin{aligned} & \|X_1^{(v)} - X_1^{(v)}S^{(v)}\|_F^2 = \|X_s^{(v)} - X_1^{(v)}(:,i)S^{(v)}(i,:)\|_F^2 \\ & = tr((X_s^{(v)} - X_1^{(v)}(:,i)S^{(v)}(i,:))^T(X_s^{(v)} - X_1^{(v)}(:,i)S^{(v)}(i,:))) \\ & = tr((X_s^{(v)})^T X_s^{(v)} - (S^{(v)}(i,:))^T(X_1^{(v)}(:,i))^T X_s^{(v)} \\ & - (X_s^{(v)})^T X_1^{(v)}(:,i)S^{(v)}(i,:)) \\ & + (S^{(v)}(i,:))^T(X_1^{(v)}(:,i))^T X_1^{(v)}(:,i)S^{(v)}(i,:)) \\ & = tr((X_s^{(v)})^T X_s^{(v)}) - 2tr(S^{(v)}(i,:)(X_s^{(v)})^T X_1^{(v)}(:,i)) \\ & + tr((X_1^{(v)}(:,i))^T X_1^{(v)}(:,i)S^{(v)}(i,:)(S^{(v)}(i,:))^T). \end{aligned} \tag{15}$$

Subsequently, the objective vector expression for $S^{(v)}(i,:)$ is obtained as follows:

$$\begin{aligned} J2(S^{(v)}(i,:)) & = (X_1^{(v)}(:,i))^T X_1^{(v)}(:,i)S^{(v)}(i,:)(S^{(v)}(i,:))^T \\ & - 2S^{(v)}(i,:)(X_s^{(v)})^T X_1^{(v)}(:,i) \\ & + \frac{\alpha}{2}S^{(v)}(i,:)P(:,i). \end{aligned} \tag{16}$$

Similarly, the objective vector expression can also be expressed in the following form. There is only a constant difference between the two forms:

$$\begin{aligned} J2(S^{(v)}(i,:)) & = \|(S^{(v)}(i,:))^T - \frac{(X_s^{(v)})^T X_1^{(v)}(:,i)}{(X_1^{(v)}(:,i))^T X_1^{(v)}(:,i)}\|_2^2 \\ & + \frac{\alpha}{2}S^{(v)}(i,:)P(:,i) \\ & s.t. S^{(v)}(i,:)(i) = 0. \end{aligned} \tag{17}$$

Let $\frac{(X_s^{(v)})^T X_1^{(v)}(:,i)}{(X_1^{(v)}(:,i))^T X_1^{(v)}(:,i)} = Q^{(v)}(:,i)$. Suppose the subscript of the vector is k ; then, if $k = i$, $(S^{(v)}(i,:))^T(k) = 0$, otherwise solve the following equation:

$$\begin{aligned} J2(S^{(v)}(i,:)(k)) & = ((S^{(v)}(i,:))^T(k) - Q^{(v)}(:,i)(k))^2 \\ & + \frac{\alpha}{2}S^{(v)}(i,:)(k)P(:,i)(k), \end{aligned} \tag{18}$$

According to $\frac{dJ2(S^{(v)}(i,:)(k))}{dS^{(v)}(i,:)(k)} = 0$, the solution is as follows:

$$(S^{(v)}(i,:))^T(k) = \begin{cases} Q^{(v)}(:,i)(k) - \frac{\alpha P^{(:,i)}(k)}{4}, & \text{if } Q^{(v)}(:,i)(k) > \frac{\alpha P^{(:,i)}(k)}{4}; \\ Q^{(v)}(:,i)(k) + \frac{\alpha P^{(:,i)}(k)}{4}, & \text{if } Q^{(v)}(:,i)(k) < \frac{\alpha P^{(:,i)}(k)}{4}; \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

Update C:

To effectively calculate the clustering label C, $W^{(v)}$ and $S^{(V)}$ are fixed, and irrelevant items are ignored. The optimization formula can be rewritten as follows:

$$J3(C) = \sum_{v=1}^m u^{(v)} (\| (X^{(v)})^T W^{(v)} - C \|_F^2 + \alpha \text{tr}(C^T L^{(v)} C)), \tag{20}$$

$$s.t. C^T C = I_k, C \geq 0.$$

To remove the orthogonal constraint, a penalty term $\rho \|C^T C - I_k\|_F^2$ is added to function (20). The following optimization functions are available:

$$\sum_{v=1}^m u^{(v)} (\| (X^{(v)})^T W^{(v)} - C \|_F^2 + \alpha \text{tr}(C^T L^{(v)} C)) \tag{21}$$

$$+ \rho \|C^T C - I_k\|_F^2, s.t. C \geq 0.$$

The Lagrangian operator ϕ is introduced to remove the inequality constraints and the following Lagrangian function is obtained:

$$\omega(C, \phi) = \sum_{v=1}^m u^{(v)} (\| (X^{(v)})^T W^{(v)} - C \|_F^2 + \alpha \text{tr}(C^T L^{(v)} C)) \tag{22}$$

$$+ \rho \|C^T C - I_k\|_F^2 - \text{tr}(\phi^T C).$$

Take $\omega(C, \phi)$ to the derivative of C, then:

$$\frac{\partial \omega(C, \phi)}{\partial C} = \sum_{v=1}^m u^{(v)} (-2(X^{(v)})^T W^{(v)} + 2C + 2\alpha L^{(v)} C) \tag{23}$$

$$+ 4\rho C(C^T C - I_k) - \phi = 0.$$

Thus, ϕ is obtained as follows:

$$\phi = \sum_{v=1}^m u^{(v)} (-2(X^{(v)})^T W^{(v)} + 2C + 2\alpha L^{(v)} C) \tag{24}$$

$$+ 4\rho C(C^T C - I_k).$$

Based on the Karush–Kuhn–Tucker condition [23] $\phi_{ij} C_{ij} = 0$, the following equation is obtained:

$$\left(\sum_{v=1}^m u^{(v)} (-2(X^{(v)})^T W^{(v)} + 2C + 2\alpha L^{(v)} C) \right. \tag{25}$$

$$\left. + 4\rho C(C^T C - I_k) \right)_{ij} C_{ij} = 0.$$

The following update formulas are obtained:

$$C_{ij} = \frac{(\sum_{v=1}^m u^{(v)} ((X^{(v)})^T W^{(v)}) + 2\rho C)_{ij}}{(\sum_{v=1}^m u^{(v)} (C + \alpha L^{(v)} C) + 2\rho C C^T C)_{ij}} C_{ij}. \tag{26}$$

After updating C , C must be regularized to ensure that it satisfies the following constraint: $C^T C = I_k$.

3.3. Convergence

Theorem 3. *The iterative optimization process $J1(W^{(v)})$ automatically reduces the objective function value until it converges.*

Proof. The first term of $J1(W^{(v)}) : \Omega(X^{(v)}, W^{(v)}, C) = \|(X^{(v)})^T W^{(v)} - C\|_F^2$ and its Hessian matrix value is:

$$\frac{\partial^2 \Omega(X^{(v)}, W^{(v)}, C)}{\partial (W^{(v)})^2} = 2X^{(v)}(X^{(v)})^T \geq 0. \tag{27}$$

Therefore, $J1(W^{(v)})$ is convex; that is,

$$\Omega(X^{(v)}, W_{(t+1)}^{(v)}, C) \leq \Omega(X^{(v)}, W_t^{(v)}, C). \tag{28}$$

The second term of $J1(W^{(v)}) : \Phi(W^{(v)}) = \|W^{(v)}\|_{2,1} = \sum_{i=1}^{l^{(v)}} \sqrt{\sum_{j=1}^k (W_{ij}^{(v)})^2} = \sum_{i=1}^{l^{(v)}} \|W_i^{(v)}\|_2$ and its Hessian matrix value is

$$\frac{\partial^2 \text{tr}((W^{(v)})^T D^{(v)} W^{(v)})}{\partial (W^{(v)})^2} = 2D^{(v)} \geq 0. \tag{29}$$

Therefore,

$$\Phi(W_{t+1}^{(v)}) \leq \Phi(W_t^{(v)}). \tag{30}$$

Then, $J1(W_{(t+1)}^{(v)}) \leq J1(W_t^{(v)})$. The proof is completed. \square

Theorem 4. *The iterative optimization process of Algorithm 1 automatically reduces the value of the objective function (4) until it converges.*

Proof. Other variables are fixed such that the objective function $J1(W^{(v)})$ is related to $W^{(v)}$. Theorem 3 proves that, under the update rule, the objective value of $J1(W^{(v)})$ is automatically reduced:

$$J(S^{(v)}, C, W_{t+1}^{(v)}) \leq J(S^{(v)}, C, W_t^{(v)}). \tag{31}$$

Given the other fixed variables, the objective function $J2(W^{(v)})$ is related to $S^{(v)}$. Then, the Hessian matrix of $J2(W^{(v)})$ is $\frac{\partial^2 J2(S^{(v)})}{\partial (S^{(v)})^2} = 2(X_1^{(v)})^T X_1^{(v)} \geq 0$, which is a positive semi-definite matrix. Therefore,

$$J(S_{i+1}^{(v)}, C, W^{(v)}) \leq J(S_i^{(v)}, C, W^{(v)}). \tag{32}$$

Fix other variables and update C ; the Hessian matrix of the objective function $J3(C)$ is $2\widetilde{L}^{(v)} \geq 0$, where $\widetilde{L}^{(v)} = aL^{(v)} + I$. Thus,

$$J(S^{(v)}, C_{t+1}, W^{(v)}) \leq J(S^{(v)}, C_t, W^{(v)}). \tag{33}$$

The proof is complete. \square

Algorithm 1 MFSC FOR CLUSTERING

Require: $\{X^{(v)}\}_{v=1}^m, \{u^{(v)}\}_{v=1}^m, k, \alpha, \beta$
Ensure: ACC, NMI

```

for v=1 to m do
  initialize  $\{S^{(v)}\}_{v=1}^m, \{W^{(v)}\}_{v=1}^m$ , and C
end for
while not convergence do
  for v=1 to m do
    update  $\{W^{(v)}\}_{v=1}^m$  according to Equation (8)
    update  $\{S^{(v)}\}_{v=1}^m$  according to Equation (19)
    update C according to Equation (26)
  end for
end while
for v = 1 to m do
  Sort each feature for  $X^{(v)}$  according to  $\|W^{(v)}\|_F^2$  in descending order and select the
  top- $f$  ranked ones;
   $X_{new} = [X_{new}, X_f^{(v)}]$ 
end for
kmeans clustering for  $X_{new}$  ;
Calculate ACC and NMI

```

3.4. Complexity Analysis

In this section, the time complexity of the three subproblems in the optimization model is calculated:

In subproblem $J1(W^{(v)})$, term $X^{(v)}(X^{(v)})^T$ requires $O(n(d^{(v)})^2)$ and its inverse matrix requires $O(n(d^{(v)})^3)$. The time complexity of term $(X^{(v)}(X^{(v)})^T + \beta D^{(v)})^{-1}X^{(v)}C$ is $O(d^{(v)} \times n \times k)$. Therefore, the total time complexity of the subproblem is $O(\sum_{v=1}^m (n(d^{(v)})^2 + n(d^{(v)})^3 + d^{(v)} \times n \times k))$.

In subproblem $J2(S^{(v)})$, each row of $S^{(v)}$ requires matrix multiplication and the time complexity is $O(n^2 d^{(v)})$. Therefore, the total time complexity of the subproblem is $O(\sum_{v=1}^m n^3 d^{(v)})$.

In subproblem $J3(C)$, the calculation of term $(X^{(v)})^T W^{(v)}$ requires $O(d^{(v)} \times n \times k)$, and the calculation of terms $L^{(v)}C$ and $CC^T C$ requires $O(n^2 k)$. The total time complexity is $O(\sum_{v=1}^m (d^{(v)} \times n \times k + n^2 \times k))$.

4. Experiment

This section conducts an evaluation experiment on the MFSC algorithm using some kinds of benchmark multiview datasets and its performance is compared with those of other related algorithms.

4.1. Dataset

The evaluation experiment of the MFSC algorithm was conducted on 5 real multiview datasets: news dataset 3sources, paper dataset Cora, information retrieval and research dataset CiteSeer, website dataset BBCSport, and blog website dataset BlogCatalog. Table 1 summarizes the 5 datasets. In addition, the specific information is as follows:

- 3sources** The news dataset comes from three online news sources: BBC, Reuters, and Guardian. All articles are placed within the text. Out of a total of 948 articles from three sources, 169 are adopted. It is noteworthy that each article in the dataset has a main theme.
- Cora** The paper dataset contains a total of 2708 sample points, which are divided into 7 categories. Each sample point is a scientific paper. A paper comprises a 1433-dimensional word vector.

3. **CiteSeer** The papers in the information retrieval and research dataset are divided into six categories, containing a total of 3312 papers, and records the citation or citation information between the papers. Through sorting, 3703 unique words are obtained.
4. **BBCSport** The website dataset comes from 544 dataset points of the BBC sports website, including sports news related to 5 subject areas (athletics, cricket, football, rugby, and tennis) and 2 related views.
5. **BlogCatalog** BlogCatalog is the social blog directory which manages the bloggers and their blogs. The data consists of 10,312 articles, divided into 6 categories, each article with two views: blog content and its related tags.

Table 1. Statistical table of typical datasets.

	3sources			Cora		CiteSeer		BBCSport		BlogCatalog	
the number of vertexes	169			2708		3312		544		10,312	
the number of classes	6			7		6		5		6	
the number of views	3			2		2		2		2	
view	V1	V2	V3	V1	V2	V1	V2	V1	V2	V1	V2
the number of features	3560	3631	3068	1433	2708	3703	3312	3183	3203	6115	5764

4.2. Benchmark Method

MFSC is compared with the following algorithms.

1. LapScore (the Laplacian score function) selects features with strong separability, where the distribution of feature vector values is consistent with the sample distribution, thereby reflecting the inherent manifold structure of the data.
2. Relief is a multiclass feature selection algorithm. The larger the weight of the feature, the stronger the classification ability of the feature. Features with weights less than a certain threshold are removed.
3. MCFS [24] (a multiclustering feature selection) algorithm uses the spectral method to preserve the local manifold topology and selects features using a method that can preserve the clustering topology.
4. PRMA [7] (probabilistic robust matrix approximation) is a multiview clustering algorithm with robust regularization matrix approximation. Powerful norm and manifold regularization are used for regularization matrix factorization, making the model more distinguishable in multiview data clustering.
5. GMNMF [17] (graph-based multiview nonnegative matrix factorization) is a multiview nonnegative matrix decomposition clustering algorithm involving intrinsic structure information among multiview graphs.
6. SCFS [3] (subspace clustering-based feature selection) is an unsupervised feature selection method based on subspace clustering that maintains a similarity relation by learning the representation of the low-dimensional subspace of samples.
7. JMVFG [11] (joint multiview unsupervised feature selection and graph learning) proposed a unified objective function that can simultaneously learn clustering structure, and global and local similarity graphs.
8. CCSFS [12] (consensus cluster structure guided multiview unsupervised feature selection) unifies subspace learning, clustering learning, consensus learning, and unsupervised feature selection into one optimization framework for mutual optimization.

4.3. Evaluation Metrics

ACC (Accuracy) is used to compare the obtained cluster labels $cluster_label_i$ with the real cluster labels $truth_label_i$. The ACC is defined as follows:

$$ACC = \frac{\sum (cluster_label_i == truth_label_i)}{m}, \quad (34)$$

where m denotes the total number of data samples.

NMI (Normalized Mutual Information) is the mutual information entropy between the obtained and real cluster labels; it is defined as follows:

$$NMI = \frac{\sum_{i=1}^K \sum_{j=1}^K n_{i,j} \log\left(\frac{n_{i,j}}{n_i n_j}\right)}{\sqrt{(\sum_{i=1}^K n_i \log\left(\frac{n_i}{n}\right)) (\sum_{j=1}^K n_j \log\left(\frac{n_j}{n}\right))}}, \quad (35)$$

where n_i denotes the sample number of cluster C_i ($1 \leq i \leq K$) and $n_{i,j}$ denotes the sample number in both cluster C_i and category C_j .

4.4. Parameter Setting

The MFSC algorithm has two main parameters α and β . In the experiment, the parameter range of α is set to $\{10^{-3}, 10^{-2}, 0.1, 1, 10, 10^2, 10^4\}$ and that of β is set to $\{10^{-6}, 10^{-3}, 10^{-2}, 1, 10, 10^2, 10^4\}$. The correlation coefficient of the 3sources data is $u = \{0.3, 0.3, 0.4\}$ and the other data views are the two-view data, with the correlation coefficient defined as follows $u = \{0.5, 0.5\}$. The value range of *feature#* (feature selection number) is set to $\{100, 200, 300, 400, 500\}$. Due to the large scale of BlogCatalog dataset, its range is $feature\# \in \{500, 1000, 1500, 2000, 2500\}$. Considering that the clustering method k-means usually converges to a local minimum, it is necessary to repeat each experiment 20 times and report the average performance.

4.5. Results of Multiview Clustering

Tables 2 and 3 show the ACC and NMI values of the different feature selection and multiview clustering methods. To determine the impact of the benchmark feature selection method on clustering, this experiment first merges the results of multiview feature selection into new data and then executes k-means. The final value is the average value of the clustering of different feature selection values. Based on the experimental results, MFSC performs well on both ACC and NMI, which proves the effectiveness of the algorithm.

Table 2. ACC of different methods on typical datasets.

	3sources	Cora	CiteSeer	BBCSport	BlogCatalog
LapScore	0.36	0.3	0.32	0.39	0.55
ReliefF	0.42	0.27	0.31	0.42	0.59
MCFS	0.55	0.29	0.27	0.47	0.54
PRMA	0.54	0.26	0.36	0.37	0.56
GMNMF	0.46	0.25	0.26	0.45	0.57
SCFS	0.65	0.4	0.34	0.45	0.56
JMVFG	0.64	0.35	0.44	0.42	0.52
CCSFS	0.54	0.3	0.4	0.45	0.5
MFSC	0.69	0.39	0.38	0.48	0.54

Table 3. NMI of different methods on typical datasets.

	3sources	Cora	CiteSeer	BBCSport	BlogCatalog
LapScore	0.18	0.27	0.3	0.15	0.32
ReliefF	0.35	0.25	0.5	0.32	0.4
MCFS	0.48	0.27	0.28	0.24	0.29
PRMA	0.49	0.24	0.34	0.42	0.31
GMNMF	0.4	0.29	0.56	0.51	0.33
SCFS	0.54	0.35	0.4	0.6	0.47
JMVFG	0.45	0.3	0.51	0.57	0.42
CCSFS	0.31	0.34	0.41	0.56	0.4
MFSC	0.58	0.3	0.42	0.78	0.45

4.6. Parameter Analysis

To achieve peak clustering performance, we tune parameters α , β , and $feature\#$. Thus, we alter their values to see how they affect the ACC and NMI of clustering for 3sources data, Cora data, CiteSeer data, BBCSport data and BlogCatalog data.

Figures 2–4 show the clustering experiment results of parameters α , β , and $feature\#$ in the 3sources dataset.

Figure 2 shows the change description of α , β , and the clustering indexes ACC and NMI in 3source. The average value is taken as the final result. Based on the ACC and NMI results in 3source, the MFSC algorithm is sensitive to parameters α and β . When parameter α is small, the performance of ACC is relatively high. When parameter β is large, the performance of NMI is relatively high.

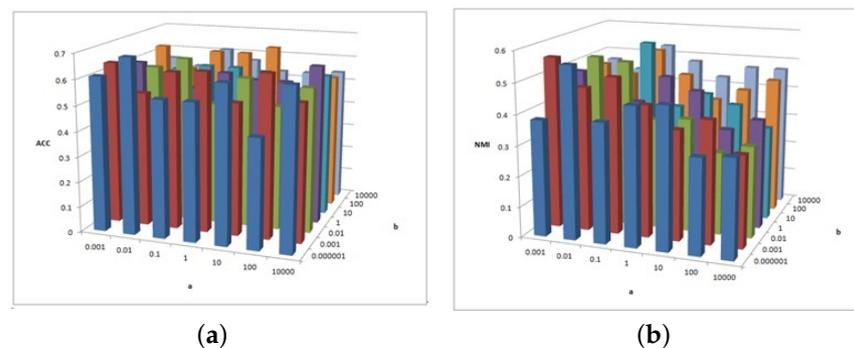


Figure 2. (a) ACC values of parameter α and parameter β for 3sources data. (b) NMI values of parameter α and parameter β for 3sources data.

Figure 3 shows the change description of parameter α and $feature\#$ and the values of clustering indexes ACC and NMI from 3source. In most cases, when the parameter $\alpha = 0.001$, the ACC and NMI of the MFSC exhibit better performance in the feature selection dimension, which shows the importance of capturing the multiview manifold structure and embedding it into the feature selection model.

Figure 4 shows the change description of parameter β and feature number and the clustering performance ACC and NMI values from 3source. It can be concluded that the MFSC is sensitive to the selected feature number. As the value of feature selection increases, the ACC and NMI increase. In most cases, when the parameter $\beta = 10,000$, the ACC and NMI of the MFSC exhibit better performance. To ensure the sparsity of matrix W , the larger the value of the feature selection, the greater the importance and the stronger the clustering performance.

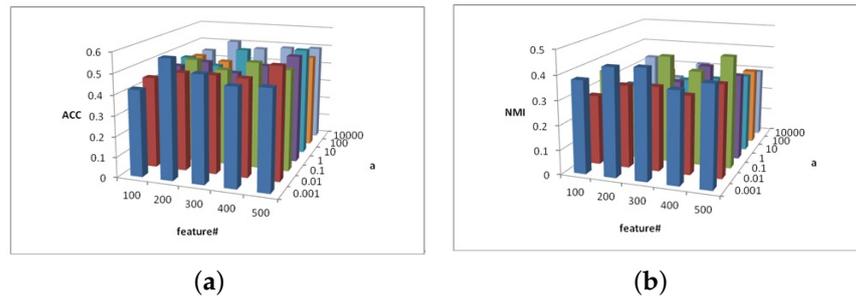


Figure 3. (a) ACC values of parameter α and parameter $feature\#$ for 3sources data. (b) NMI values of parameter α and parameter $feature\#$ for 3sources data.

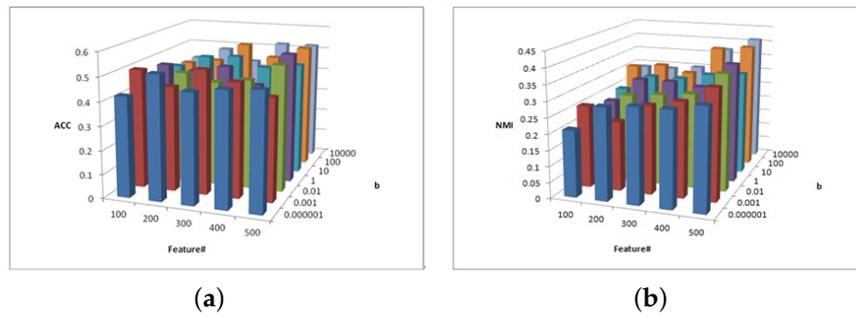


Figure 4. (a) ACC values of parameter β and parameter $feature\#$ for 3sources data. (b) NMI values of parameter β and parameter $feature\#$ for 3sources data.

Figures 5–7 show the clustering results of parameters α , β , and $feature\#$ in the Cora dataset.

Figure 5a shows that the ACC is insensitive to parameter α and insensitive to parameter β on interval $\beta \in [0.000001, 0.001]$ or $\beta \in [0.01, 1, 10, 100, 10000]$. Figure 5b shows that the NMI is insensitive to parameter α but is sensitive to parameter β . When $\beta \geq 1$, the NMI value is larger.

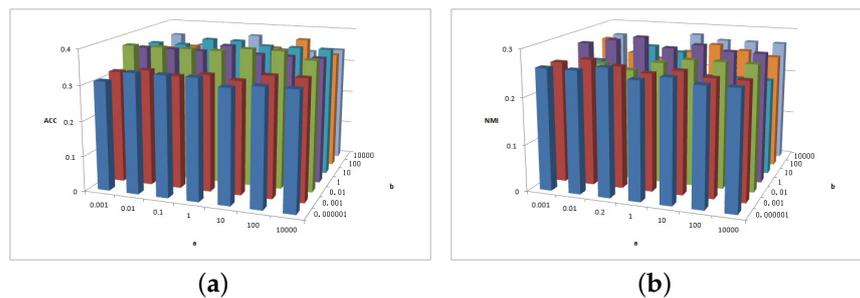


Figure 5. (a) ACC values of parameter α and parameter β for Cora data. (b) NMI values of parameter β and parameter $feature\#$ for Cora data.

Figure 6 shows the clustering results of parameter α and $feature\#$ in Cora dataset. As depicted in Figure 6a, when parameters α and $feature\#$ increase, the ACC value increases. In Figure 6b, $feature\#$ is sensitive to NMI value, while the overall relative value of $feature\#$ is larger and the NMI value is larger.

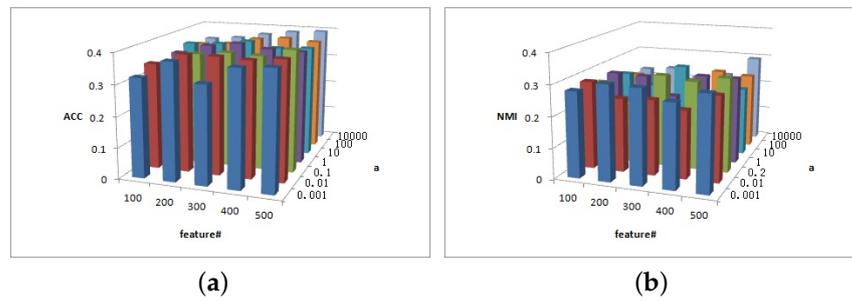


Figure 6. (a) ACC values of parameter α and parameter $feature\#$ for Cora data. (b) NMI values of parameter β and parameter $feature\#$ for Cora data.

Figure 7 depicts the clustering results of parameters β and $feature\#$ in the Cora dataset. As depicted in Figure 7a, for $\beta \leq 0.01$, the ACC increases as $feature\#$ increases; otherwise, for $\beta > 0.01$ the ACC value remains basically unchanged. Figure 7b shows that, when $feature\# = 100$ and 500 , the NMI value is larger.

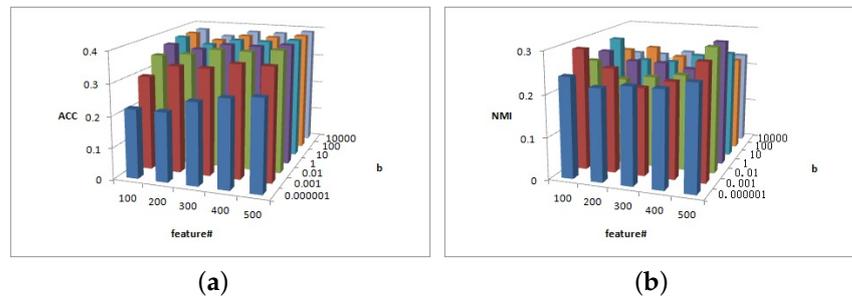


Figure 7. (a) ACC values of parameter β and parameter $feature\#$ for Cora data. (b) NMI values of parameter β and parameter $feature\#$ for Cora data.

Figures 8–10 show the clustering results of parameters α , β , and $feature\#$ in CiteSeer dataset. Figure 8a shows that the ACC is insensitive to parameters α and β , but Figure 8b shows that the NMI is insensitive to parameters $\alpha \in \{0.001, 0.01, 1, 10, 100\}$ and $\beta \in \{0.01, 0.1, 1, 100\}$; when the two parameters are larger or smaller, the NMI value exhibits a small fluctuation.

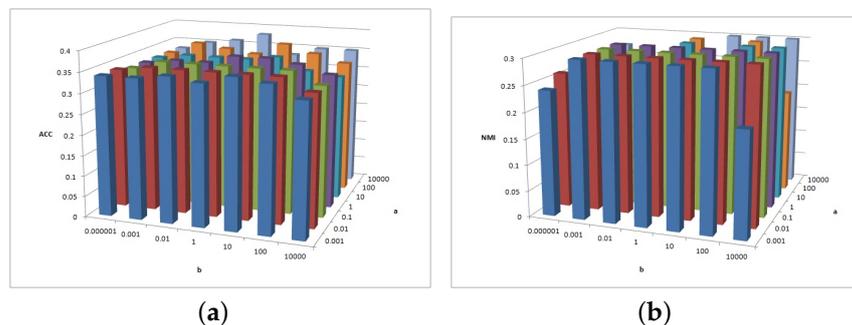


Figure 8. (a) ACC values of parameter α and parameter β for CiteSeer data. (b) NMI values of parameter α and parameter β for CiteSeer data.

Figure 9 shows the clustering results of parameters α and $feature\#$ in the CiteSeer dataset. As illustrated in Figure 9a, the magnitude of the ACC value difference is 0.05 and the ACC is insensitive to parameter α . For $feature\# = 300$, the ACC has better performance. As shown in Figure 9b, the NMI is slightly sensitive to parameters α and β . For $\alpha = 1$, when $feature\# = 200$, a larger NMI is achieved.

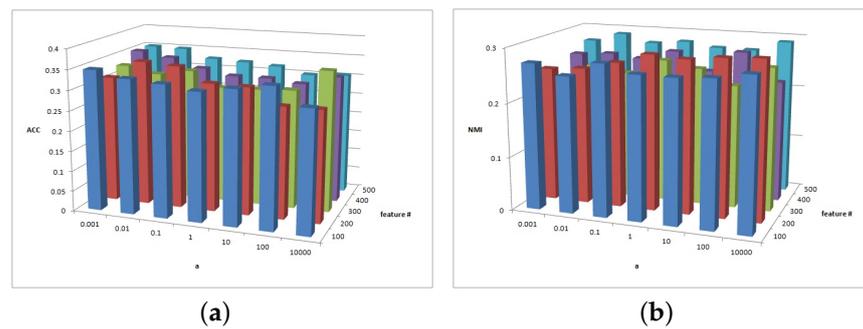


Figure 9. (a) ACC values of parameter α and parameter $feature\#$ for CiteSeer data. (b) NMI values of parameter α and parameter $feature\#$ for CiteSeer data.

Figure 10 shows the clustering results of parameters β and $feature\#$ in the CiteSeer dataset. As demonstrated in Figure 10a, the magnitude of the ACC value difference is 0.05 and the ACC is insensitive to parameter β . In general, the NMI performance is better when $\beta \in \{0.001, 0.01, 1\}$. The ACC performance is stabler when $feature\# > 200$. Figure 10b shows that the NMI results are almost insensitive to parameter β and $feature\#$ in the CiteSeer dataset. When $\beta = 0.001$, the NMI result is greater.

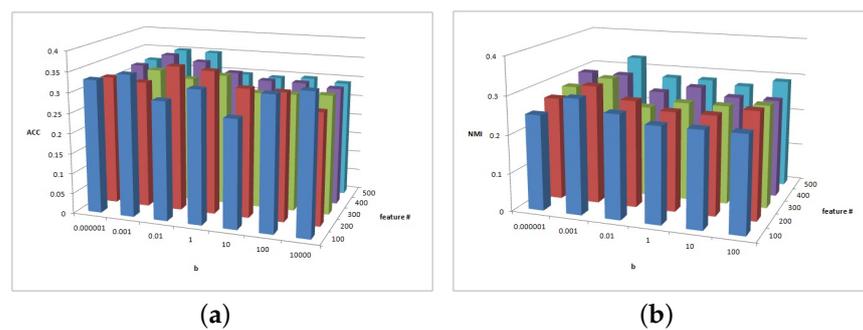


Figure 10. (a) ACC values of parameter β and parameter $feature\#$ for CiteSeer data. (b) NMI values of parameter β and parameter $feature\#$ for CiteSeer data.

Figures 11–13 show the clustering experiment results of parameters α , β , and $feature\#$ in the BBCSport dataset.

Figure 11a shows that the ACC is insensitive to parameters α and β in the BBCSport dataset. Figure 11b shows that the NMI is insensitive to parameter α but the NMI changes slightly when $\beta \geq 1$. However, the NMI results have a peak when $\beta = 1$.

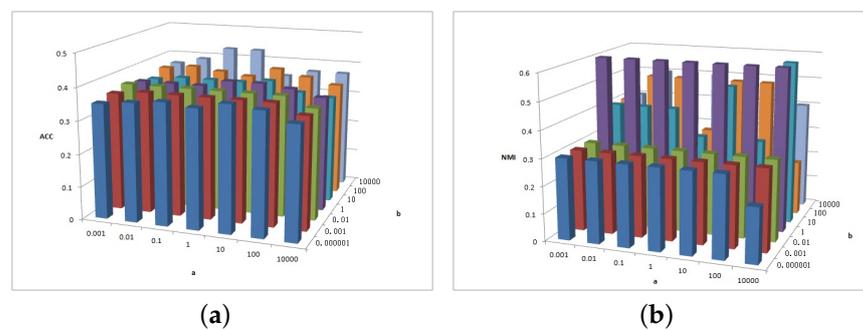


Figure 11. (a) ACC values of parameter α and parameter β for BBCSport data. (b) NMI values of parameter α and parameter β for BBCSport data.

Figure 12a shows the clustering ACC of parameters α and $feature\#$ in the BBCSport dataset. The magnitude of the ACC value difference in this figure is 0.05, and the ACC is

insensitive to parameter α and $feature\#$ in BBCSport dataset. Comparatively, it has high ACC with $feature\# \geq 400$ and $0.1 \leq a \leq 10$. Figure 12b shows the clustering NMI of parameters α and $feature\#$ in the BBCSport dataset. When $feature\# \leq 200$, the results of NMI are insensitive to parameters α and $feature\#$. NMI increases first and then decreases with parameter α . When $feature\# > 200$, NMI has a greater value when $a = 0.1$ and $feature\# = 300$ and 400 .

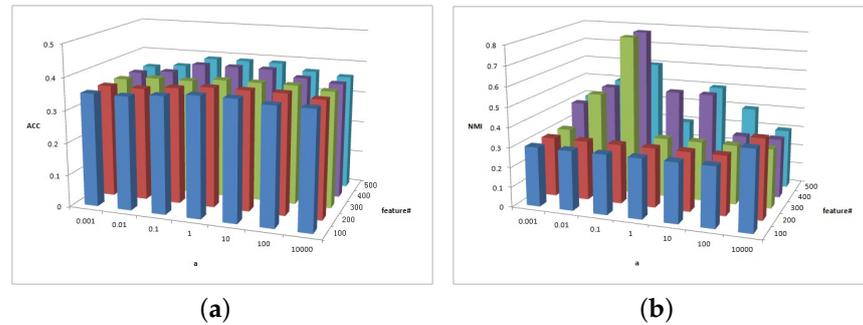


Figure 12. (a) ACC values of parameter α and parameter $feature\#$ for BBCSport data. (b) NMI values of parameter α and parameter $feature\#$ for BBCSport data.

Figure 13a shows the clustering ACC of parameters b and $feature\#$ in the BBCSport dataset. The magnitude of ACC value difference in this figure is 0.05, and the ACC is insensitive to parameters β and $feature\#$ in the BBCSport dataset. Comparatively, it has high ACC with $feature\# \geq 300$ and $a = 10,000$. Figure 13b shows the clustering NMI of parameters b and $feature\#$ in the BBCSport dataset. NMI is sensitive to $feature\#$, and NMI has a greater value when $feature\# \geq 300$ and $b \geq 100$.

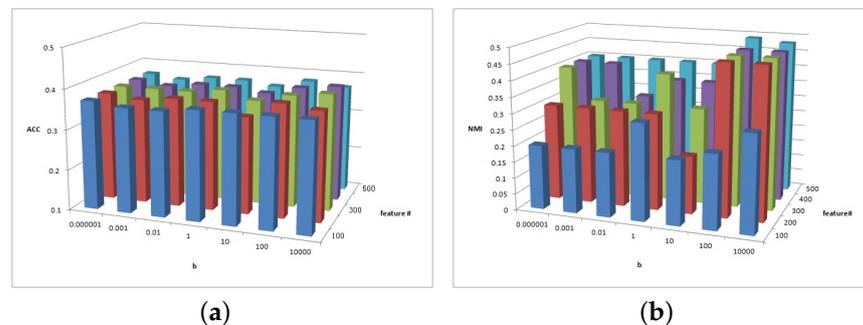


Figure 13. (a) ACC values of parameter β and parameter $feature\#$ for BBCSport data. (b) NMI values of parameter β and parameter $feature\#$ for BBCSport data.

Figures 14–16 show the clustering results of parameters α , β , and $feature\#$ in BlogCatalog dataset.

Figure 14a shows that ACC is insensitive to α but, when $\beta = 1000$, its performance is better. Figure 14b shows the NMI performance for parameters α and β . NMI is not very sensitive to α and β , and, when β is larger and α is smaller, NMI is relatively larger.

Figure 15a shows the ACC performance with parameters α and $feature\#$. When $feature\# = 1500$ and $\alpha > 10$, the ACC performance is better. Figure 15b shows the NMI decrease with parameter $feature\#$ and NMI is not sensitive to α . Figure 15 indicates that higher $feature\#$ is not necessarily better for BlogCatalog data.

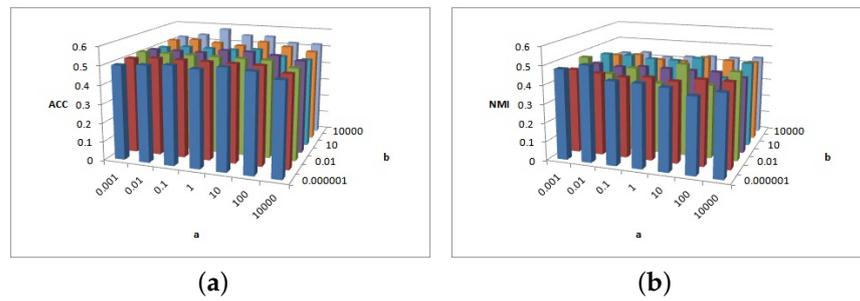


Figure 14. (a) ACC values of parameter α and parameter β for BlogCatalog data. (b) NMI values of parameter α and parameter β for for BlogCatalog data.

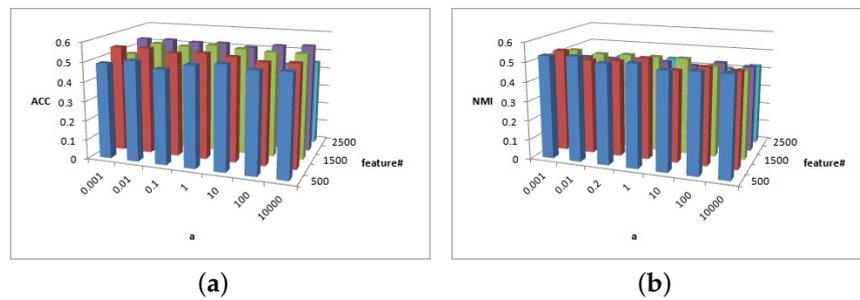


Figure 15. (a) ACC values of parameter α and parameter $feature\#$ for BlogCatalog data. (b) NMI values of parameter α and parameter $feature\#$ for BlogCatalog data.

Figure 16 shows that β is sensitive to ACC and NMI, and $feature\#$ is sensitive to clustering performance. In Figure 16a, when $feature\# \geq 1500$ and $\beta \geq 1$, the ACC is better. In Figure 16b, NMI increases with parameter β ; when $\beta \geq 10$, NMI has a greater value.

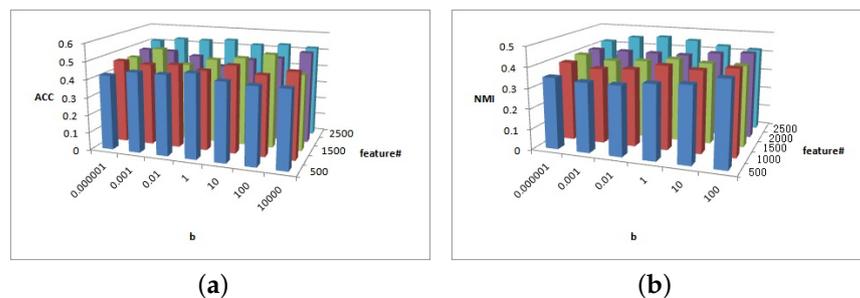


Figure 16. (a) ACC values of parameter β and parameter $feature\#$ for BlogCatalog data. (b) NMI values of parameter β and parameter $feature\#$ for BlogCatalog data.

Parameter sensitivity remains a challenging and unsolved problem in feature selection. This experiment analyzes the sensitivity of parameters α , β , and $feature\#$. We performed similar parameter sensitivity analyses for the data sources. The results show that MFSC is almost insensitive to parameters α and β for ACC performance. This shows the importance of capturing the multiview manifold structure embedded in the feature selection model. However, the MFSC is sensitive to $feature\#$. This is because the network size affects the number of feature selections.

4.7. Convergence Analysis

The convergence effects of the datasets are shown in Figure 17. In addition, the convergence effect of the remaining data is similar. Based on the experimental results, the convergence effect is relatively good. The objective function increases as the number of

convergences increases and quickly reaches a constant convergence value regardless of the initial objective value.

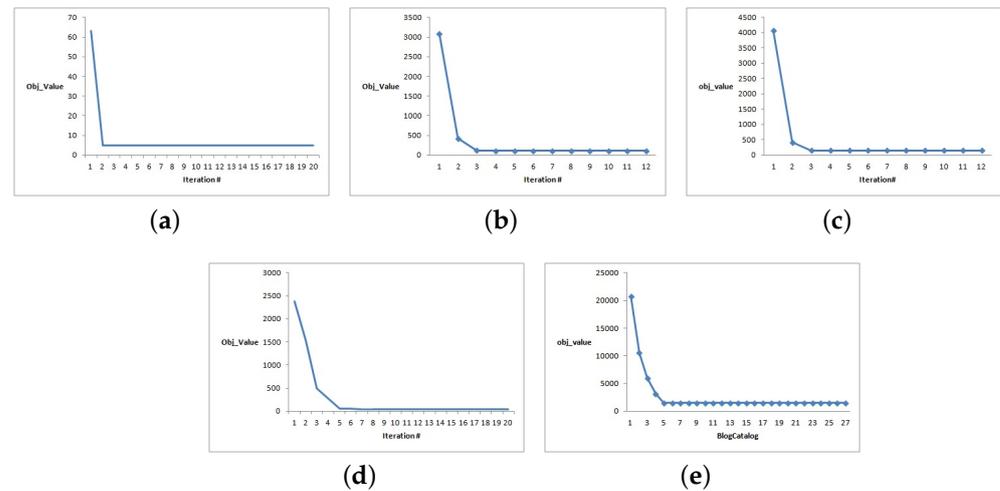


Figure 17. MFSC convergence curves. (a) 3sources; (b) Cora; (c) CiteSeer; (d) BBCSport; (e) BlogCatalog.

5. Conclusions and Future Work

This study proposes a multiview clustering-guided feature selection algorithm for multiview data, which integrates subspace learning and feature selection, and embeds the norm of manifold regularization. This feature selection algorithm reduces the influence of redundancy and the irrelevant matrix of the multiview data. In addition, clustering is used as the standard for feature selection. This algorithm can perform feature selection to ensure that the clustering structure remains unchanged. It is noteworthy that the complementary contribution of each view is fully considered. The optimization process is calculated and theoretically analyzed, and experiments are performed using a multiview dataset. It can be concluded that the algorithm is effective and superior to many existing feature selection algorithms or multiview clustering algorithms.

Although our method achieves good clustering performance, on the one hand, we mainly consider social network data, while other types of multimodal data graph structures are not considered. On the other hand, some parameters need to be manually adjusted. Recently, deep learning has demonstrated excellent feature extraction capabilities in multiview data, such as images and natural languages. In the future, we will study how to integrate deep learning and the MFSC model to process multiview data and accurately describe semantic information.

Author Contributions: N.L.: Conceptualization, Methodology, Software, Investigation, Formal Analysis, Writing—Original Draft, Writing—review and editing, Funding Acquisition; M.P.: Visualization, Validation, Data Curation, Supervision, Funding Acquisition; Q.W.: Investigation, Formal Analysis, Resources. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National key R&D Program of China (Grant 2017YFB0202901, 2017YFB0202905), the Hunan Natural Science Foundation Project (No. 2023JJ50353).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. He, X. Locality preserving projections. *Adv. Neural Inf. Process. Syst.* **2003**, *16*, 186–197.
2. Dong, W.; Wu, X.J.; Li, H.; Feng, Z.H.; Kittler, J. Subspace Clustering via Joint Unsupervised Feature Selection. In Proceedings of the 25th International Conference on Pattern Recognition, Milan, Italy, 10–15 January 2021; pp. 3861–3870.
3. Parsa, M.G.; Zare, H.; Ghatte, M. Unsupervised feature selection based on adaptive similarity learning and subspace clustering. *Eng. Appl. Artif. Intell.* **2020**, *95*, 103855. [[CrossRef](#)]
4. Nie, F.; Zhu, W.; Li, X. Structured Graph Optimization for Unsupervised Feature Selection. *IEEE Trans. Knowl. Data Eng.* **2019**, *33*, 1210–1222. [[CrossRef](#)]
5. Ma, X.; Yan, X.; Liu, J.; Zhong, G. Simultaneous multi-graph learning and clustering for multiview data. *Inf. Sci.* **2022**, *593*, 472–487. [[CrossRef](#)]
6. Bickel, S.; Scheffer, T. Multi-view clustering. In Proceedings of the IEEE International Conference on Data Mining, Brighton, UK, 1–4 November 2004; pp. 234–243.
7. Pu, J.; Qian, Z.; Zhang, L.; Bo, D.; You, J. Multiview clustering based on Robust and Regularized Matrix Approximation. In Proceedings of the International Conference on Pattern Recognition, Cancun, Mexico, 4–8 December 2016; pp. 2550–2555.
8. Yang, Y.; Wang, H. Multi-view Clustering: A Survey. *Big Data Min. Anal.* **2018**, *1*, 83–107.
9. Venkatesh, B.; Anuradha, J. A Review of Feature Selection and Its Methods. *Cybern. Inf. Technol.* **2019**, *19*, 3. [[CrossRef](#)]
10. Zhang, H.; Wu, D.; Nie, F.; Wang, R.; Li, X. Multilevel projections with adaptive neighbor graph for unsupervised multi-view feature selection. *Inf. Fusion* **2021**, *70*, 129–140. [[CrossRef](#)]
11. Fang, S.G.; Huang, D.; Wang, C.D.; Tang, Y. Joint Multi-view Unsupervised Feature Selection and Graph Learning. *Comput. Vis. Pattern Recognit.* **2022**, *98*, 1–18. [[CrossRef](#)]
12. Cao, Z.; Xie, X.; Sun, F.; Qian, J. Consensus cluster structure guided multi-view unsupervised feature selection. *Knowl.-Based Syst.* **2023**, *271*, 110578–110590. [[CrossRef](#)]
13. Tang, C.; Zheng, X.; Liu, X.; Zhang, W.; Zhang, J.; Xiong, J.; Wang, L. Cross-View Locality Preserved Diversity and Consensus Learning for Multi-View Unsupervised Feature Selection. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 4705–4716. [[CrossRef](#)]
14. Liu, H.; Shao, M.; Fu, Y. Feature Selection with Unsupervised Consensus Guidance. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 2319–2331. [[CrossRef](#)]
15. He, X.; Cai, D.; Niyogi, P. Laplacian Score for Feature Selection. In Proceedings of the Neural Information Processing Systems, NIPS 2005, Vancouver, BC, Canada, 5–8 December 2005; pp. 2547–2556.
16. Xu, Y.; Zhong, A.; Yang, J.; Zhang, D. LPP solution schemes for use with face recognition. *Pattern Recognit.* **2010**, *43*, 4165–4176. [[CrossRef](#)]
17. Hao, W.; Yan, Y.; Li, T. Multi-view Clustering via Concept Factorization with Local Manifold Regularization. In Proceedings of the IEEE International Conference on Data Mining, Barcelona, Spain, 12–15 December 2016; pp. 1245–1250.
18. Livescu, K. Multiview Clustering via Canonical Correlation Analysis. In Proceedings of the International Conference on Machine Learning, Montreal, QC, Canada, 20–21 June 2008; pp. 129–136.
19. Nie, F.; Huang, H.; Cai, X.; Ding, C.H.Q. Efficient and Robust Feature Selection via Joint $l_{2,1}$ -Norms Minimization. In Proceedings of the International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 6–9 December 2010; pp. 1813–1821.
20. Newman, M.W.; Libraty, N.; On, O.; On, K.A.; On, K.A. The Laplacian spectrum of graphs. *Graph Theory Comb. Appl.* **1991**, *18*, 871–898.
21. Fan, K. On a Theorem of Weyl Concerning Eigenvalues of Linear Transformations: II. *Proc. Natl. Acad. Sci. USA* **1950**, *35*, 652–655. [[CrossRef](#)] [[PubMed](#)]
22. Nie, F.; Xu, D.; Tsang, W.H.; Zhang, C. Flexible Manifold Embedding: A Framework for Semi-Supervised and Unsupervised Dimension Reduction. *IEEE Trans. Image Process.* **2010**, *19*, 1921–1932. [[PubMed](#)]
23. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004; pp. 43–365.
24. Cai, D.; Zhang, C.; He, X. Unsupervised feature selection for multi-cluster data. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; pp. 333–342.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.