TIME-FS: Joint Learning of Tensorial Incomplete Multi-View Unsupervised Feature Selection and Missing-View Imputation

Yanyong Huang[1], Minghui Lu[1]*, Wei Huang[2], Xiuwen Yi[3], Tianrui Li[4]

[1]Joint Laboratory of Data Science and Business Intelligence, Southwestern University of Finance and Economics
[2]College of Computer and Data Science, Fuzhou University
[3]JD Intelligent Cities Research and JD Intelligent Cities Business Unit
[4]School of Computing and Artificial Intelligence, Southwest Jiaotong University
huangyy@swufe.edu.cn, lu.m.h@foxmail.com, huangweifujian@126.com, xiuwenyi@foxmail.com, trli@swjtu.edu.cn

Abstract

Multi-view unsupervised feature selection (MUFS) has received considerable attention in recent years. Existing MUFS methods for processing unlabeled incomplete multi-view data, where some samples are missing in certain views, first impute the missing values and then perform feature selection on the completed dataset. However, treating imputation and feature selection as two separate processes overlooks their potential interactions. The graph-guided local structure gleaned from feature selection can aid in imputation, which in turn can enhance the feature selection performance. Additionally, most similarity graph-based MUFS methods suffer from high computational costs. To address these problems, we propose a novel MUFS method, termed Tensorial Incomplete Multi-view unsupErvised Feature Selection (TIME-FS). TIME-FS unifies missing value recovery, discriminative feature selection, and low-dimensional representation learning within a joint framework through matrix decomposition. Then, TIME-FS conducts CP decomposition on tensor data formed by the low-dimensional representations of different views to learn a consistent anchor graph across views and a view-preference weight matrix, both of which simultaneously guide missing view imputation and feature selection. Furthermore, an efficient algorithm with low time complexity and rapid convergence is proposed to solve TIME-FS. Extensive experimental results demonstrate the effectiveness and efficiency of TIME-FS over state-of-the-art methods.

Introduction

Multi-view data are common in real-world applications, where the same sample is described using heterogeneous features from different perspectives (Sun 2013; Zhao et al. 2017). Multi-view data are typically high-dimensional and unlabeled, which is attributable to the diverse sources of features and the labor-intensive process of labeling (Xie, Tao, and Wei 2016; Tang et al. 2022a). As an important unsupervised learning method, multi-view unsupervised feature selection (MUFS) aims to select informative features from unlabeled multi-view data, thereby achieving dimensionality reduction and enhancing the performance of downstream tasks (Li et al. 2017; Zhang et al. 2019).

*Corresponding Author

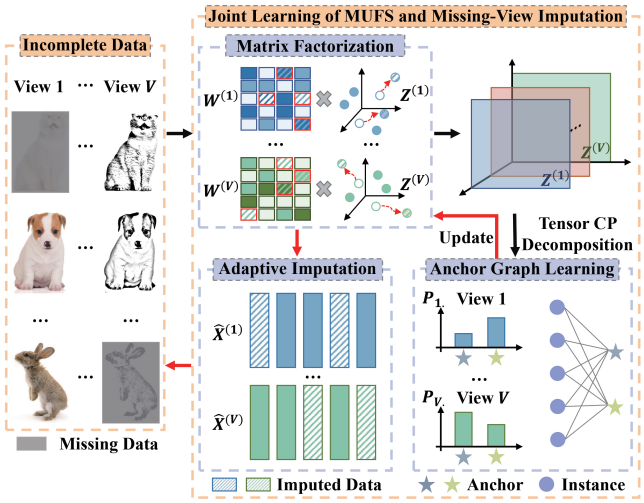


Figure 1: The framework of the proposed TIME-FS.

Existing MUFS methods are commonly classified into two categories. The first category involves concatenating multiple views into a single view and subsequently applying single-view feature selection methods. Typical methods in this category include JELSR (Hou et al. 2014), DUFS (Lim and Kim 2021) and HSL (Mi et al. 2024). In contrast to the first category, which combines multi-view data in a sequential manner, the second category directly conducts feature selection on multi-view data by leveraging potential correlations between views, exemplified by methods such as TLR-MUFS (Yuan et al. 2022), SDFS (Zhou et al. 2023), and JMVFG (Fang et al. 2024). TLR-MUFS employs tensor low-rank regularization on similarity graphs to capture cross-view consistency during the feature selection process. JMVFG integrates multi-view graph learning and multi-view feature selection into a unified framework.

Although previous methods have demonstrated promising feature selection performance, they are predicated on the assumption that each sample appears in all views. This assumption does not always hold in real-world scenarios. For example, in e-commerce recommendation systems, many users only have purchase behavior features, with profile information often missing due to privacy concerns (Elkahky,

Song, and He 2015). There is limited research on unsupervised feature selection for incomplete multi-view data. CVFS (Xu et al. 2021) introduces a cross-view unsupervised feature selection method for handling incomplete multi-view data. C^2IMUFS (Huang et al. 2023) proposes an incomplete multi-view unsupervised feature selection method that utilizes complementary and consensus information across different views to construct the similarity graph. These two methods begin by imputing the missing data, followed by developing a multi-view feature selection approach on the completed dataset.

As mentioned above, the previously mentioned MUFS method is applied to incomplete data through two sequential steps: first, imputing the missing data, followed by performing feature selection on the imputed dataset. However, treating imputation and feature selection as two separate processes fails to leverage their potential synergy. The local structural information within and between views obtained from feature selection can guide imputation. In turn, the refined imputation can improve the effectiveness of feature selection. Moreover, most MUFS methods that use the similarity graph to characterize the local structure of data exhibit high time complexity due to the graph construction and the eigen-decomposition of the Laplacian matrix (Cao and Xie 2024b).

To address the above issues, we propose a novel MUFS method, called Tensorial Incomplete Multi-view unsupErvised Feature Selection (TIME-FS). Specifically, we first propose a joint learning framework that integrates missing-view imputation, feature selection, and low-dimensional representation learning through matrix factorization. Then, we perform CP decomposition on the tensor data formed by the low-dimensional representations of different views. This can learn a consistent anchor graph that captures the local structural information between views and a view-preference weight matrix that encapsulates view-specific information. The learned anchor graph and view weight matrix can simultaneously guide both missing-view imputation and feature selection. Furthermore, we develop an efficient optimization algorithm to solve TIME-FS with low computational time complexity and rapid convergence. Fig. 1 illustrates the framework of TIME-FS. The main contributions of this paper are summarized as follows:

- To the best of our knowledge, this is the first work to propose a joint learning framework for adaptive missing-view imputation, incomplete multi-view unsupervised feature selection, and tensor-based anchor graph learning. This framework facilitates seamless interaction among these modules, ultimately enhancing the performance of feature selection.
- TIME-FS adaptively learns a consistent anchor graph and a view-preference weight matrix by performing CP decomposition on the third-order tensor composed of the low-dimensional representations of views. By leveraging the learned view-consistent and view-specific local structural information, it can simultaneously guide both missing-view imputation and feature selection.
- An efficient iterative algorithm with low time complexity and good convergence is developed to solve the proposed TIME-FS. Comprehensive experiments on real-world datasets demonstrate the superiority of TIME-FS over other state-of-the-art (SOTA) methods in terms of clustering performance and running time.

The Propose Method

We first summarize the notations used throughout this paper and introduce the relevant definitions for matrix and tensor operations. We use bold calligraphic letters to denote tensors (e.g., $\boldsymbol{\mathcal{A}}$) and bold uppercase letters for matrices (e.g., $\boldsymbol{A}$). For any matrix $\boldsymbol{A}$, $\boldsymbol{A}_{i.}$, $\boldsymbol{A}_{.j}$, and A_{ij} represent its i-th row, j-th column, and (i,j)-th entry, respectively. The Frobenius norm of matrix $\boldsymbol{A} \in \mathbb{R}^{n\times r}$ is defined as $\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{r} A_{ij}^2}$, and the $\ell_{2,1}$-norm is given by $\|\boldsymbol{A}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{r} A_{ij}^2}$. $\mathrm{Tr}(\boldsymbol{A})$ and $\boldsymbol{A}^{\mathrm{T}}$ respectively denote the trace and transpose of matrix $\boldsymbol{A}$. Given two matrices $\boldsymbol{A} \in \mathbb{R}^{n\times r}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times r}$, the Khatri-Rao product of them is defined as $\boldsymbol{A} \odot \boldsymbol{B} = [\boldsymbol{A}_{.1} \otimes \boldsymbol{B}_{.1}, \ldots, \boldsymbol{A}_{.r} \otimes \boldsymbol{B}_{.r}] \in \mathbb{R}^{nm\times r}$, where $\otimes$ is the Kronecker product. For any third-order tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{n_1\times n_2\times n_3}$, the CP decomposition of $\boldsymbol{\mathcal{Z}}$ can be approximated by a sum of r rank-one tensors, expressed as $\boldsymbol{\mathcal{Z}} \approx [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!] \equiv \sum_{i=1}^{r}(\boldsymbol{A}_{i.} \circ \boldsymbol{B}_{i.} \circ \boldsymbol{C}_{i.})$, where $\circ$ denotes the vector outer product, and $\boldsymbol{A} \in \mathbb{R}^{n_1\times r}$, $\boldsymbol{B} \in \mathbb{R}^{n_2\times r}$, and $\boldsymbol{C} \in \mathbb{R}^{n_3\times r}$ (Carroll and Chang 1970).

Joint Learning of MUFS and Adaptive Imputation

Given an incomplete multi-view dataset $\boldsymbol{\mathcal{X}} = \{\boldsymbol{X}^{(v)}\}_{v=1}^{V}$, where $\boldsymbol{X}^{(v)} = [\boldsymbol{X}_{.1}^{(v)}, \ldots, \mathring{\boldsymbol{X}}_{.j}^{(v)}, \ldots, \boldsymbol{X}_{.n}^{(v)}] \in \mathbb{R}^{d_v\times n}$ is the v-th view's data matrix with n samples and d_v features, and $\mathring{\boldsymbol{X}}_{.j}^{(v)}$ denotes that the j-th sample in the v-th view is missing. To select discriminative features from the given multi-view dataset, many matrix factorization-based unsupervised feature selection methods have been widely developed (Mairal et al. 2010; Huang et al. 2023; Mi et al. 2024). However, when using existing methods for feature selection on incomplete data, it is necessary to first impute the missing values with predefined entries to form a complete dataset, after which feature selection is conducted. Such a "two-stage" way fails to account for the potential interactions between data imputation and feature selection. In this paper, we propose a "one-stage" method using matrix factorization that simultaneously performs adaptive missing-view imputation and feature selection. Moreover, we automatically assign weights to each view based on their importance. This can be formulated as follows:

$$\begin{aligned}\min_{\substack{\boldsymbol{E}^{(v)},\boldsymbol{W}^{(v)},\\ \boldsymbol{Z}^{(v)},\alpha_v}} & \sum_{v=1}^{V}\alpha_v^{\gamma}\{\|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \lambda\|\boldsymbol{W}^{(v)}\|_{2,1}\} \\ s.t.\ & \hat{\boldsymbol{X}}^{(v)} = \boldsymbol{X}^{(v)} + \boldsymbol{E}^{(v)}\boldsymbol{G}^{(v)}, \boldsymbol{Z}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}} = \boldsymbol{I}, \\ & \sum_{v=1}^{V}\alpha_v = 1, \alpha_v \geq 0, v = 1, 2, \ldots, V,\end{aligned} \tag{1}$$

where $\boldsymbol{E}^{(v)} \in \mathbb{R}^{d_v \times n_v}$ is the data matrix in $\boldsymbol{X}^{(v)}$ requiring imputation of missing samples, n_v is the number of missing samples in the v-th view, $\boldsymbol{W}^{(v)} \in \mathbb{R}^{d_v \times c}$ denotes the feature selection matrix, $\boldsymbol{Z}^{(v)} \in \mathbb{R}^{c \times n}$ represents the low-dimensional representation of the v-th view, α_v is the weight of the v-th view, and γ and λ are two regularization parameters. $\boldsymbol{G}^{(v)} \in \mathbb{R}^{n_v \times n}$ serves as an indicator matrix for the missing instances, defined as

$$G_{ij}^{(v)} = \begin{cases} 1 & \text{if } \mathring{\boldsymbol{X}}_{\cdot j}^{(v)} \text{ is the } i\text{-th missing instance} \\ & \text{in the } v\text{-th view;} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

By imposing the constraint $\hat{\boldsymbol{X}}^{(v)} = \boldsymbol{X}^{(v)} + \boldsymbol{E}^{(v)}\boldsymbol{G}^{(v)}$, the missing data in the original matrix $\boldsymbol{X}^{(v)}$ is filled using $\boldsymbol{E}^{(v)}$, while the existing data remains unchanged. The orthogonal constraint on $\boldsymbol{Z}^{(v)}$ in Eq. (1) ensures that the learned low-dimensional representation is more compact and avoids redundancy. Furthermore, The $\ell_{2,1}$-norm is imposed on the feature selection matrix $\boldsymbol{W}^{(v)}$ to promote sparsity, thereby driving the weights of non-essential features toward zero.

In Eq. (1), the missing data $\boldsymbol{E}^{(v)}$ is treated as a variable and is optimized simultaneously with the feature selection matrix $\boldsymbol{W}^{(v)}$ and the low-dimensional representation $\boldsymbol{Z}^{(v)}$. This strategy guarantees that the imputed data is reconstructed from the discriminative features identified by $\boldsymbol{W}^{(v)}$ and the compact low-dimensional representation, thereby avoiding the influence of redundant features in the original data. In turn, the adaptive imputation process also boosts the effectiveness of feature selection. Therefore, the joint learning of adaptive missing-view imputation, incomplete multi-view unsupervised feature selection, and low-dimensional representation learning can facilitate seamless interaction among these components and improve feature selection performance.

Tensor-Based Anchor Graph Learning

Previous studies have demonstrated that leveraging local structural information in multi-view data is beneficial to improve the feature selection performance (Huang et al. 2023; Liang et al. 2023). However, most conventional MUFS methods use similarity graphs to describe the local structure of data, often incurring high computational costs. To address this issue, we use tensor CP decomposition on a third-order tensor formed from the low-dimensional representations of different views to obtain a consistent anchor graph and a view-preference weight matrix. By learning a set of anchors that are much fewer than the total number of samples, the resulting anchor graph effectively reduces the computational burden while capturing the consistent local structure between the original samples and these anchors across views. Additionally, by combining the view-preference weight matrix and anchor graph, we can capture the view-specific local structural information. This can be formulated as follows:

$$\begin{aligned} &\min_{\boldsymbol{A},\boldsymbol{M},\boldsymbol{P}} \|\boldsymbol{\mathcal{Z}} - [\![\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{P}]\!]\|_F^2 + \eta\|\boldsymbol{M}\|_F^2 \\ &\quad s.t.\ \ \boldsymbol{M}\mathbf{1}_r = \mathbf{1}_n, \boldsymbol{M} \geq 0, \end{aligned} \tag{3}$$

where $\boldsymbol{\mathcal{Z}} = \Phi\{\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)}, \ldots, \boldsymbol{Z}^{(V)}\} \in \mathbb{R}^{c \times n \times V}$ is a third-order tensor constructed by stacking V low-dimensional representations $\{\boldsymbol{Z}^{(v)}\}_{v=1}^{V}$, $\boldsymbol{A} \in \mathbb{R}^{c \times r}$ denotes the consensus anchor set consisting of r ($r \ll n$) anchors, $\boldsymbol{M} \in \mathbb{R}^{n \times r}$ represents a consistent anchor graph, $\boldsymbol{P} \in \mathbb{R}^{V \times r}$ is a view-preference weight matrix, η is a trade-off parameter, and $\mathbf{1}_r$ denotes the r-dimensional column vector with all entries being one. In Eq. (3), each row of $\boldsymbol{P}$ represents the differing preference levels that an anchor assigns to each view. By integrating each row of $\boldsymbol{P}$ with the consistent anchor graph $\boldsymbol{M}$, one can capture the local structural information specific to each view. Hence, Eq. (3) simultaneously utilizes shared information between views and view-specific information to capture the local data structure, which aids in recovering missing data and enhances feature selection performance.

By combining Eqs. (1) and (3), the final objective function of TIME-FS can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{\Theta}} &\sum_{v=1}^{V} \alpha_v^{\gamma}[\|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \lambda\|\boldsymbol{W}^{(v)}\|_{2,1}] \\ &+ \|\boldsymbol{\mathcal{Z}} - [\![\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{P}]\!]\|_F^2 + \eta\|\boldsymbol{M}\|_F^2 \\ s.t.\ &\hat{\boldsymbol{X}}^{(v)} = \boldsymbol{X}^{(v)} + \boldsymbol{E}^{(v)}\boldsymbol{G}^{(V)}, \boldsymbol{Z}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}} = \boldsymbol{I}, \\ &\boldsymbol{\mathcal{Z}} = \Phi\{\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)}, \ldots, \boldsymbol{Z}^{(V)}\}, \boldsymbol{M}\mathbf{1}_r = \mathbf{1}_n, \\ &\boldsymbol{M} \geq 0, \sum_{v=1}^{V} \alpha_v = 1, \alpha_v \geq 0, v = 1, \ldots, V, \end{aligned} \tag{4}$$

where $\boldsymbol{\Theta} = \{\{\boldsymbol{W}^{(v)}, \boldsymbol{Z}^{(v)}, \boldsymbol{E}^{(v)}, \alpha_v\}_{v=1}^{V}, \boldsymbol{A}, \boldsymbol{M}, \boldsymbol{P}\}$.

By integrating feature selection, adaptive missing-view imputation, and tensor-based anchor graph learning within a unified framework, Time-FS is capable of leveraging both view-consistent and view-specific local structural information to enhance the performance of feature selection and facilitate the imputation of missing data. Furthermore, acquiring more discriminative features and reliable imputed data aids in accurately characterizing the local structure of the data. These three components can synergistically enhance one another to recover missing values and achieve better feature selection performance.

Optimization

Since the objective function in Eq. (4) is non-convex to all variables simultaneously, we propose an alternative iterative algorithm to solve this problem.

Update $\boldsymbol{W}^{(v)}$ by Fixing Others. When other variables are fixed, $\boldsymbol{W}^{(v)}$ can be updated by solving the following problem:

$$\min_{\boldsymbol{W}^{(v)}} \|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \lambda\|\boldsymbol{W}^{(v)}\|_{2,1} \tag{5}$$

According to (Nie et al. 2010), $\|\boldsymbol{W}^{(v)}\|_{2,1}$ can be rewritten as $\mathrm{Tr}((\boldsymbol{W}^{(v)})^{\mathrm{T}}\boldsymbol{\Lambda}^{(v)}\boldsymbol{W}^{(v)})$, where $\boldsymbol{\Lambda}^{(v)}$ is a diagonal matrix, with the i-th diagonal element given by $1/(2\|\boldsymbol{W}_{i\cdot}^{(v)}\|_2 + \epsilon)$, and ϵ is set to 2×10^{-16} to avoid division by zero. Then, Eq. (5) is further transformed as follows:

$$\min_{\boldsymbol{W}^{(v)}} \|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \lambda\,\mathrm{Tr}((\boldsymbol{W}^{(v)})^{\mathrm{T}}\boldsymbol{\Lambda}^{(v)}\boldsymbol{W}^{(v)}) \tag{6}$$

By taking the derivative of Eq. (6) w.r.t. $\boldsymbol{W}^{(v)}$ and setting it to 0, we obtain the optimal solution for $\boldsymbol{W}^{(v)}$:

$$\boldsymbol{W}^{(v)} = (\boldsymbol{I} - \lambda\boldsymbol{\Lambda}^{(v)})^{-1}\hat{\boldsymbol{X}}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}}, \tag{7}$$

where $\boldsymbol{\Lambda}^{(v)}$ is initially derived from $\boldsymbol{W}^{(v)}$, and both $\boldsymbol{\Lambda}^{(v)}$ and $\boldsymbol{W}^{(v)}$ are updated in turns.

Update $\boldsymbol{Z}^{(v)}$ by Fixing Others. After fixing other variables, the objective function w.r.t. $\boldsymbol{Z}^{(v)}$ is reduced to:

$$\begin{aligned}&\min_{\boldsymbol{Z}^{(v)}} \alpha_v^{\gamma}\|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \|\boldsymbol{\mathcal{Z}} - [\![\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{P}]\!]\|_F^2\\ &s.t.\ \boldsymbol{Z}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}} = \boldsymbol{I}, \boldsymbol{\mathcal{Z}} = \Phi\{\boldsymbol{Z}^{(1)}, \ldots, \boldsymbol{Z}^{(V)}\}.\end{aligned} \tag{8}$$

According to the definition of CP decomposition (Carroll and Chang 1970), Eq. (8) can be transformed into:

$$\begin{aligned}&\min_{\boldsymbol{Z}^{(v)}} \alpha_v^{\gamma}\|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \|\boldsymbol{Z}^{(v)} - \boldsymbol{A}diag(\boldsymbol{P}_{v\cdot})\boldsymbol{M}^{\mathrm{T}}\|_F^2\\ &s.t.\ \boldsymbol{Z}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}} = \boldsymbol{I}.\end{aligned} \tag{9}$$

By means of the property of matrix trace, we can reformulate Eq. (9) as follows:

$$\min_{\boldsymbol{Z}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}}=\boldsymbol{I}} \|\boldsymbol{Z}^{(v)} - \boldsymbol{Q}^{(v)}\|_F^2 \tag{10}$$

where $\boldsymbol{Q}^{(v)} = \alpha_v^{\gamma}\boldsymbol{W}^{(v)}\hat{\boldsymbol{X}}^{(v)} + \boldsymbol{A}diag(\boldsymbol{P}_{v\cdot})\boldsymbol{M}^{\mathrm{T}}$. According to the orthogonal Procrustes problem in (Viklands 2006), the optimal solution of Eq. (10) can be obtained as follows:

$$\boldsymbol{Z}^{(v)} = \boldsymbol{U}^{(v)}(\boldsymbol{V}^{(v)})^{\mathrm{T}}, \tag{11}$$

where $\boldsymbol{U}^{(v)} \in \mathbb{R}^{c\times c}$, $\boldsymbol{V}^{(v)} \in \mathbb{R}^{c\times n}$ denote the left and right singular matrix of $\boldsymbol{Q}^{(v)}$, respectively.

Update A by Fixing Others. With other variables fixed, we can update $\mathbf{A}$ by solving the following problem:

$$\min_{\boldsymbol{A}} \|\boldsymbol{\mathcal{Z}} - [\![\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{P}]\!]\|_F^2 \tag{12}$$

According to (Kolda and Bader 2009), Eq. (12) can be rewritten as follows:

$$\min_{\boldsymbol{A}} \|\boldsymbol{\mathcal{Z}}_{(1)} - \boldsymbol{A}(\boldsymbol{P} \odot \boldsymbol{M})^{\mathrm{T}}\|_F^2 \tag{13}$$

where $\boldsymbol{\mathcal{Z}}_{(1)} \in \mathbb{R}^{c\times nV}$ is the mode-1 unfolding of tensor $\boldsymbol{\mathcal{Z}}$, and $\odot$ stands for Khatri-Rao product. By taking the derivative of Eq. (13) w.r.t. $\boldsymbol{A}$ and setting it to 0, we get the following closed-form solution:

$$\boldsymbol{A} = \boldsymbol{\mathcal{Z}}_{(1)}(\boldsymbol{P} \odot \boldsymbol{M})[(\boldsymbol{P} \odot \boldsymbol{M})^{\mathrm{T}}(\boldsymbol{P} \odot \boldsymbol{M})]^{-1} \tag{14}$$

Update $\boldsymbol{M}$ by Fixing Others. When other variables are fixed, updating $\boldsymbol{M}$ is equal to solve the following problem:

$$\begin{aligned}&\min_{\boldsymbol{M}}\ \|\boldsymbol{\mathcal{Z}} - [\![\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{P}]\!]\|_F^2 + \eta\|\boldsymbol{M}\|_F^2\\ &\ s.t.\ \boldsymbol{M}\mathbf{1}_r = \mathbf{1}_n, \boldsymbol{M} \geq 0,\end{aligned} \tag{15}$$

Base on (Kolda and Bader 2009), Eq. (15) can be formulated as follows:

$$\begin{aligned}&\min_{\boldsymbol{M}}\ \|\boldsymbol{\mathcal{Z}}_{(2)} - \boldsymbol{M}(\boldsymbol{P} \odot \boldsymbol{A})^{\mathrm{T}}\|_F^2 + \eta\,\mathrm{Tr}(\boldsymbol{M}\boldsymbol{M}^{\mathrm{T}})\\ &\ s.t.\ \boldsymbol{M}\mathbf{1}_r = \mathbf{1}_n, \boldsymbol{M} \geq 0,\end{aligned} \tag{16}$$

where $\boldsymbol{\mathcal{Z}}_{(2)} \in \mathbb{R}^{n\times cV}$ is the mode-2 unfolding of tensor $\boldsymbol{\mathcal{Z}}$.

Eq. (16) can be solved row-by-row by optimizing the following problem:

$$\begin{aligned}&\min_{\boldsymbol{M}_{i\cdot}}\ \frac{1}{2}\|\boldsymbol{M}_{i\cdot} + \boldsymbol{Y}_{i\cdot}\|_2^2\\ &\ s.t.\ \boldsymbol{M}_{i\cdot}\mathbf{1}_r = 1, \boldsymbol{M}_{i\cdot} \geq 0,\end{aligned} \tag{17}$$

where $\boldsymbol{Y} = -\boldsymbol{\mathcal{Z}}_{(2)}(\boldsymbol{P} \odot \boldsymbol{A})[\eta\boldsymbol{I}_r + (\boldsymbol{P} \odot \boldsymbol{A})^{\mathrm{T}}(\boldsymbol{P} \odot \boldsymbol{A})]^{-1}$. Eq. (17) can be solved using the same procedure as in (Wang, Yang, and Liu 2019). Then, we have

$$\boldsymbol{M}_{i\cdot} = (\frac{1}{r}(1 + \sum_{j=1}^{r} Y_{ij})\mathbf{1}_r^{\mathrm{T}} - \boldsymbol{Y}_{i\cdot})_+, \tag{18}$$

where $(y)_+ = max(0, y)$ indicates the max function to ensure that y is non-negative.

Update $\boldsymbol{P}$ by Fixing Others. Similar to the process of solving Eq. (12), we can obtain the following equivalent optimization problem w.r.t. $\boldsymbol{P}$ after fixing other variables:

$$\min_{\boldsymbol{P}} \|\boldsymbol{\mathcal{Z}}_{(3)} - \boldsymbol{P}(\boldsymbol{M} \odot \boldsymbol{A})^{\mathrm{T}}\|_F^2 \tag{19}$$

where $\boldsymbol{\mathcal{Z}}_{(3)} \in \mathbb{R}^{V\times cn}$ is the mode-3 unfolding of tensor $\boldsymbol{\mathcal{Z}}$. By taking the derivative of Eq. (19) w.r.t. $\boldsymbol{P}$ and setting it to 0, we can obtain:

$$\boldsymbol{P} = \boldsymbol{\mathcal{Z}}_{(3)}(\boldsymbol{M} \odot \boldsymbol{A})[(\boldsymbol{M} \odot \boldsymbol{A})^{\mathrm{T}}(\boldsymbol{M} \odot \boldsymbol{A})]^{-1} \tag{20}$$

Update $\boldsymbol{E}^{(v)}$ by Fixing Others. After fixing the other variables, the optimization problem for updating $\boldsymbol{E}^{(v)}$ becomes:

$$\min_{\boldsymbol{E}^{(v)}} \|\boldsymbol{X}^{(v)} + \boldsymbol{E}^{(v)}\boldsymbol{G}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 \tag{21}$$

According to Eq. (2), we have $\boldsymbol{X}^{(v)}(\boldsymbol{G}^{(v)})^{\mathrm{T}} = 0$. Then, the optimal solution $\boldsymbol{E}^{(v)}$ can be obtained in closed form: $\boldsymbol{E}^{(v)} = \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}(\boldsymbol{G}^{(v)})^{\mathrm{T}}$.

Update α_v by Fixing Others. By fixing the other variables, the objective function w.r.t. α_v becomes:

$$\min_{\alpha_v} \sum_{v=1}^{V} \alpha_v^{\gamma} d^{(v)}, s.t.\ \sum_{v=1}^{V} \alpha_v = 1, \alpha_v \geq 0, v = 1, \ldots, V, \tag{22}$$

where $d^{(v)} = \|\hat{\boldsymbol{X}}^{(v)} - \boldsymbol{W}^{(v)}\boldsymbol{Z}^{(v)}\|_F^2 + \lambda\|\boldsymbol{W}\|_{2,1}$. By employing the Lagrange multiplier method, we can obtain the optimal solution of α_v as follows:

$$\alpha_v = \frac{(d^{(v)})^{\frac{1}{1-\gamma}}}{\sum_{v=1}^{V}(d^{(v)})^{\frac{1}{1-\gamma}}} \tag{23}$$

The entire optimization procedure of TIME-FS is summarized in Algorithm 1. In this algorithm, the columns of $\{\boldsymbol{E}^v\}_{v=1}^V$ are initialized as the mean of the observed instances for each view. $\{\boldsymbol{Z}^{(v)}\}_{v=1}^V$ is initialized randomly to ensure row orthogonality, and both $\boldsymbol{A}$ and $\boldsymbol{P}$ are also randomly initialized. $\boldsymbol{W}^{(v)}$ is initially set to $\hat{\boldsymbol{X}}^{(v)}(\boldsymbol{Z}^{(v)})^{\mathrm{T}}$. $\boldsymbol{M}$ and $\{\alpha_v\}_{v=1}^V$ are initialized according to Eqs. (18) and (23), respectively.

Algorithm 1: Iterative Algorithm of TIME-FS

Input:Incomplete multi-view data $\boldsymbol{\mathcal{X}} = \{\boldsymbol{X}^{(v)}\}_{v=1}^{V}$; the parameters γ, η, and λ.

Initialize $\{\boldsymbol{E}^{(v)}, \boldsymbol{Z}^{(v)}, \boldsymbol{W}^{(v)}, \alpha_v\}_{v=1}^{V}$, $\boldsymbol{A}$, $\boldsymbol{P}$, and $\boldsymbol{M}$.
while not convergent **do**
  Update $\{\boldsymbol{W}^{(v)}\}_{v=1}^{V}$ via Eq. (7);
  Update $\{\boldsymbol{Z}^{(v)}\}_{v=1}^{V}$ via Eq. (11);
  Update $\boldsymbol{A}$ via Eq. (14);
  Update $\boldsymbol{M}$ via Eq. (18);
  Update $\boldsymbol{P}$ via Eq. (20);
  Update $\{\boldsymbol{E}^{(v)}\}_{v=1}^{V}$ by solving Eq. (21);
  Update $\{\alpha_v\}_{v=1}^{V}$ via Eq. (23);
end while

Output:Sorting the ℓ_2-norm of the rows of $\{\mathbf{W}^{(v)}\}_{v=1}^{V}$ in descending order and selecting the top k features from $\boldsymbol{\mathcal{X}}$.

Complexity and Convergence Analysis

Time Complexity Analysis. In Algorithm 1, seven variables are updated alternatively. In each iteration, updating $\boldsymbol{W}^{(v)}$ costs $\mathcal{O}(d_v nc)$. By performing economic SVD (Brand 2006) on $\boldsymbol{Q}^{(v)}$ in Eq. (11), updating $\boldsymbol{Z}^{(v)}$ costs $\mathcal{O}(d_v nc + c^2 n)$. The computational cost for updating $\boldsymbol{A}$, $\boldsymbol{M}$, and $\boldsymbol{P}$ is $\mathcal{O}(c^2 nV)$ for each. Updating $\boldsymbol{E}^{(v)}$ costs $\mathcal{O}(d_v n_v c + c n_v)$, which is less than $\mathcal{O}(d_v nc + c^2 n)$ because $n_v < n$. Updating α_v involves only element-based operations, which can be disregarded. Hence, the total time complexity of Algorithm 1 is $\mathcal{O}(dnc + c^2 nV)$, where $d = \sum_{v=1}^{V} d_v$.

Convergence Analysis. Algorithm 1 solves the resulting optimization problem through an iterative process that decomposes it into seven sub-problems, optimizing each sequentially. Since all seven sub-problems are convex and have closed-form solutions, the objective function value of Eq. (4) will decrease monotonically.

Experiment

Experimental Settings

Datasets. We use six real-world datasets for evaluation, including one text dataset (WebKB[1]), one face image dataset (ORL_mtv[2]), one handwritten digit image dataset (Mfeat[2]), and three object image datasets (COIL20[3], Caltech101[2], and Aloi[2]). Table 1 summarizes the detailed information of six datasets. In addition, we construct the incomplete multi-view data following the method in (Lin et al. 2022), by removing certain ratios of samples from each view as missing data. The missing ratio is set within the range of {10%, 20%, 30%, 40%, 50%} in the experiment.

Comparison Methods. To validate the effectiveness and efficiency of the proposed TIME-FS, we compare it with several SOTA methods, including TLR-MUFS (Yuan et al. 2022), TRCA-CGL (Liang et al. 2023), SDFS (Zhou et al. 2023), C^{2}IMUFS (Huang et al. 2023), CDMvFS (Cao, Xie, and Li 2024), JMVFG (Fang et al. 2024), SCMvFS (Cao and Xie 2024b), CDNMF (Duan et al. 2024), HSL (Mi et al. 2024), and CFSMO (Cao and Xie 2024a). Besides, a baseline (AllFea) that uses all features is also evaluated.

Datasets	Views	Samples	Features	Classes
WebKB	3	203	1703/230/230	4
ORL_mtv	3	400	4096/3304/6750	40
COIL20	3	1440	30/19/30	20
Mfeat	6	2000	216/76/64/6/240/47	10
Caltech101	6	2386	48/40/254/1984/512/928	20
Aloi	4	11025	77/13/64/64	100

Table 1: Dataset description

Comparison Schemes. As most comparison methods cannot be directly applied to incomplete multi-view data, we first impute the missing samples with the mean value of the corresponding view before applying these methods. To ensure a fair comparison, the parameters for all comparison methods are tuned using the grid search strategy to achieve the optimal results. Besides, the parameters of γ, η and λ in our method are tuned in the range of $\{1.5, 2, 3, 4, 5, 6\}$, $\{0.01, 0.1, 1, 10, 100\}$, and $\{0.001, 0.005, 0.01, 0.05\}$, respectively. Since determining the optimal number of selected features for a certain dataset remains a challenging problem (Li et al. 2017), we set the percentage of selected features from 10% to 90% in increments of 10%. In the evaluation phase, we follow commonly used metrics to assess the quality of the selected features in MUFS (Tang et al. 2022b; Fang et al. 2024): Clustering Accuracy (ACC) and Normalized Mutual Information (NMI). K-means clustering is then performed 30 times on the selected features, and the average clustering results are reported. All experiments were run in MATLAB R2022b on a desktop with an Intel Core i9-10900 CPU @ 2.80 GHz and 64 GB RAM. The source code is available at https://github.com/mingH-L/TIME-FS.

Experimental Results and Analysis

Performance Comparisons. To comprehensively evaluate the effectiveness of the proposed TIME-FS, we report the clustering performance of all methods with different feature selection ratios and missing ratios. Figs. 2 and 3 respectively show the performance of different methods in terms of ACC and NMI with varying feature selection ratios (FR) and a fixed missing ratio of 40%. As shown in Figs. 2 and 3, TIME-FS achieves competitive performance compared to the other methods when the feature selection ratio ranges from 10% to 90%. Specifically, on ORL_mtv, COIL20 and Mfeat datasets, TIME-FS obtains an average improvement of over 10% in both ACC and NMI in comparison with the best results of the SOTA methods. On Caltech101 and Aloi datasets, TIME-FS respectively achieves more than 9% and 6% improvement in average ACC and NMI compared with the second-best methods. TIME-FS still gains over 4% average improvement in ACC and NMI on WebKB dataset. Furthermore, Figs. 4 and 5 illustrate the performance of various methods in terms of ACC and NMI with varying missing ra-

[1]http://elenaher.dinauz.org/phd/data/networks/webkb/

[2]https://github.com/wangsiwei2010/awesome-multi-view-clustering

[3]https://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php

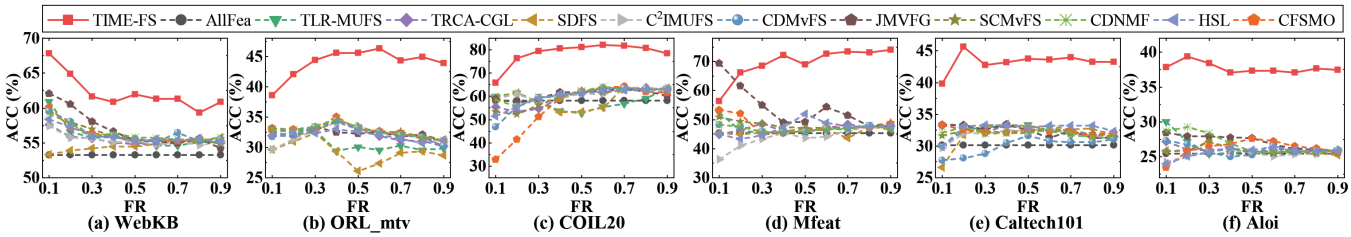


Figure 2: ACC of different methods on six datasets under different feature selection ratios.

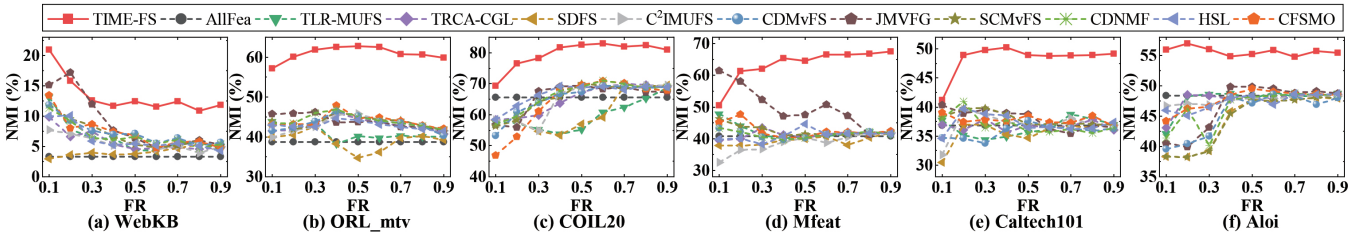


Figure 3: NMI of different methods on six datasets under different feature selection ratios.

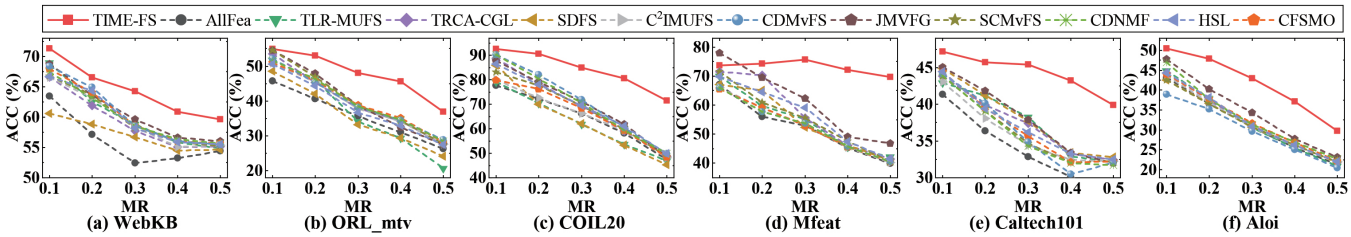


Figure 4: ACC of different methods on six datasets with different missing ratios.

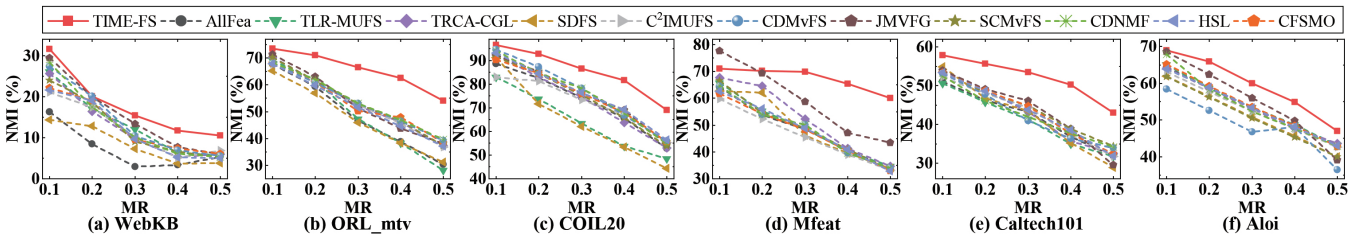


Figure 5: NMI of different methods on six datasets with different missing ratios.

tios (MR) and a fixed feature selection ratio of 40%, respectively. From these two figures, TIME-FS still achieves pretty good performance compared to other competing methods in most cases. The outstanding performance of TIME-FS results from integrating feature selection, adaptive missing-view imputation, and anchor graph learning into a unified framework, where each component is synergistically enhanced.

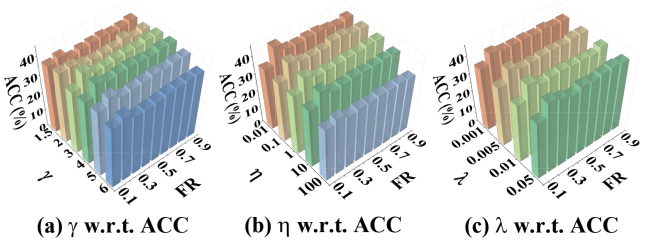
Running Time Comparison. To evaluate the computational efficiency of the proposed algorithm, Table 2 reports the average runtime of various algorithms on six datasets. Each algorithm was executed five times under identical conditions. We can see that TIME-FS has a much shorter runtime than other methods. Furthermore, we provide a theoretical comparison of the complexity of different algorithms. The last column of Table 2 shows the time complexity of each algorithm, clearly indicating that our method has the lowest time complexity. The efficiency advantage mainly arises from our algorithm's ability to capture the local structure information through learning an anchor graph and a view-preference weight matrix via tensor decomposition. In contrast, most competing methods construct similarity graphs and perform eigen-decomposition of the Laplacian matrix to capture local structure, which is more time-consuming.

Ablation Study. To demonstrate the effectiveness of the proposed module in TIME-FS, we conduct ablation experiments by comparing it with two of its variants.

(1)TIME-FS-I: The adaptive imputation module is removed from Eq. (4), and the missing values are filled with mean values.

Methods	Datasets						Time Complexity
	WebKB	ORL_mtv	COIL20	Mfeat	Caltech101	Aloi	
TIME-FS	0.33*	4.20*	0.45*	1.21*	6.54*	64.54*	$\mathcal{O}(dnc + c^2nV)$
TLR-MUFS	1.68	229.24	2.82	10.08	29.29	381.52	$\mathcal{O}((2n+1)nVlog(n) + \sum_{v=1}^{V} d_v^3)$
TRCA-CGL	1.62	105.45	15.81	53.23	127.30	4361.28	$\mathcal{O}(n^2V^2 + 2n^2Vlog(n) + d^3)$
SDFS	2.25	40.41	27.68	108.34	187.16	2984.11	$\mathcal{O}(\sum_{v=1}^{V}(n^3 + n^2d_v + nd_v^2 + d_v^2l))$
C^{2}IMUFS	2.01	27.04	110.84	375.28	621.90	4288.34	$\mathcal{O}(nc\sum_{v=1}^{V} max(d_v, n))$
CDMvFS	2.24	56.71	50.82	330.09	527.94	18331.96	$\mathcal{O}(Vn^3 + \sum_{v=1}^{V} d_v^3)$
JMVFG	2.80	12.37	3.22	29.91	68.75	825.82	$\mathcal{O}(\sum_{v=1}^{V} d_v^3 + n\sum_{v=1}^{V} d_v^2 + n^2d)$
SCMvFS	3.37	60.53	19.05	138.27	256.84	3482.33	$\mathcal{O}(Vn^3 + \sum_{v=1}^{V} d_v^3)$
CDNMF	1.72	39.05	1.07	6.14	24.42	108.73	$\mathcal{O}(n\sum_{v=1}^{V} d_v^2)$
HSL	15.32	499.09	2.47	7.52	94.47	73.78	$\mathcal{O}(d^3 + ndl)$
CFSMO	4.02	24.66	1.94	13.78	20.33	445.77	$\mathcal{O}(Vcdn + n^2V + \sum_{v=1}^{V} d_v^3)$

Table 2: Average running time (s) and time complexity of different methods across six datasets, where $d = \sum_{v=1}^{V} d_v$ and $*$ indicates a significant reduction in running time according to the Wilcoxon signed-rank test.



(a) γ w.r.t. ACC (b) η w.r.t. ACC (c) λ w.r.t. ACC

Figure 6: ACC of TIME-FS with varying parameters γ, η, λ and feature selection ratios on Caltech101 dataset.

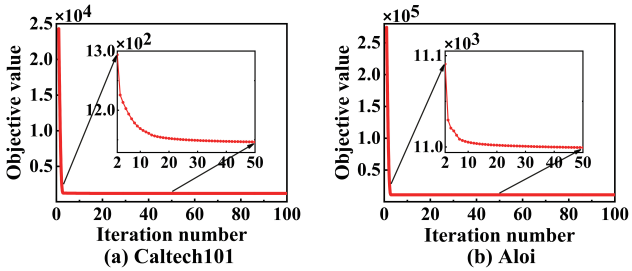


Figure 7: Convergence curves of TIME-FS on Caltech101 and Aloi datasets.

(2)TIME-FS-II: The tensor-based anchor graph learning module is excluded from Eq. (4).

Table 3 presents the ablation experiment results on six datasets with a missing ratio of 40% and a selection of 40% of all features. We can see that the performance of TIME-FS-I significantly decreases when compared to TIME-FS. This highlights the effectiveness of jointly learning adaptive imputation and feature selection to enhance performance. Moreover, TIME-FS performs significantly better than TIME-FS-II, indicating that using the consistent anchor graph and view-specific weight information can more accurately capture the local data structure.

Parameter Sensitivity and Convergence Analysis. There are three tuning parameters γ, η and λ in the proposed TIME-FS. We illustrate how the performance of TIME-FS varies with changes in the parameters and the feature selection ratio, as shown in Fig. (6). Due to limited space, we only report the ACC results on Caltech101 dataset. From Fig. (6), we can observe that TIME-FS does not vary much when γ, η, and λ are within a certain range. However, it is relatively sensitive to the feature selection ratio, which remains a challenging problem in MUFS. Furthermore, Fig. (7) shows the convergence curves of TIME-FS on Caltech101 and Aloi datasets. As can be seen, the proposed optimization algorithm converges quickly within 40 iterations.

Datasets	TIME-FS		TIME-FS-I		TIME-FS-II	
	ACC	NMI	ACC	NMI	ACC	NMI
WebKB	**60.85***	**11.70***	55.40	4.45	55.21	4.82
ORL_mtv	**45.55***	**62.54***	34.81	46.97	33.98	46.14
COIL20	**80.65***	**81.64***	55.39	59.14	59.80	65.07
Mfeat	**72.18***	**65.42***	45.95	39.23	45.62	38.77
Caltech101	**62.73***	**49.27***	34.81	46.97	32.08	37.64
Aloi	**37.10***	**54.87***	26.44	47.59	25.82	47.15

Table 3: Average performance comparison of TIME-FS and its two variants in terms of ACC and NMI, where $*$ indicates significant improvement by the Wilcoxon signed-rank test.

Conclusion

In this paper, we proposed a novel MUFS method, TIME-FS, for incomplete multi-view data by integrating feature selection, adaptive missing-view imputation, and anchor graph learning into a unified framework, where these components mutually enhance each other. Moreover, TIME-FS uses tensor CP decomposition to learn a consistent anchor graph across views and the view-preference weight matrix, effectively capturing the local data structure while reducing computational overhead. Extensive experiments conducted on six real-world datasets demonstrated that TIME-FS outperforms SOTA methods in both effectiveness and efficiency.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 72495122), the Natural Science Foundation Project of Sichuan Province (No. 2024NSFSC0504), and the Youth Fund Project of Humanities and Social Science Research of Ministry of Education (No. 21YJCZH045).

References

Brand, M. 2006. Fast Low-Rank Modifications of the Thin Singular Value Decomposition. *Linear Algebra and its Applications*, 415(1): 20–30.

Cao, Z.; and Xie, X. 2024a. Multi-View Unsupervised Complementary Feature Selection with Multi-Order Similarity Learning. *Knowledge-Based Systems*, 283: 111172.

Cao, Z.; and Xie, X. 2024b. Structure Learning with Consensus Label Information for Multi-View Unsupervised Feature Selection. *Expert Systems with Applications*, 238: 121893.

Cao, Z.; Xie, X.; and Li, Y. 2024. Multi-View Unsupervised Feature Selection with Consensus Partition and Diverse Graph. *Information Sciences*, 661: 120178.

Carroll, J. D.; and Chang, J.-J. 1970. Analysis of Individual Differences in Multidimensional Scaling via an N-Way Generalization of "Eckart-Young" Decomposition. *Psychometrika*, 35(3): 283–319.

Duan, M.; Song, P.; Zhou, S.; Mu, J.; and Liu, Z. 2024. Consensus and Discriminative Non-Negative Matrix Factorization for Multi-View Unsupervised Feature Selection. *Digital Signal Processing*, 154: 104668.

Elkahky, A. M.; Song, Y.; and He, X. 2015. A Multi-View Deep Learning Approach for Cross Domain User Modeling in Recommendation Systems. In *Proceedings of the 24th International Conference on World Wide Web*, 278–288.

Fang, S.-G.; Huang, D.; Wang, C.-D.; and Tang, Y. 2024. Joint Multi-View Unsupervised Feature Selection and Graph Learning. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(1): 16–31.

Hou, C.; Nie, F.; Li, X.; Yi, D.; and Wu, Y. 2014. Joint Embedding Learning and Sparse Regression: A Framework for Unsupervised Feature Selection. *IEEE Transactions on Cybernetics*, 44(6): 793–804.

Huang, Y.; Shen, Z.; Cai, Y.; Yi, X.; Wang, D.; Lv, F.; and Li, T. 2023. C^2IMUFS: Complementary and Consensus Learning-Based Incomplete Multi-View Unsupervised Feature Selection. *IEEE Transactions on Knowledge and Data Engineering*, 35(10): 10681–10694.

Kolda, T. G.; and Bader, B. W. 2009. Tensor Decompositions and Applications. *SIAM Review*, 51(3): 455–500.

Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Trevino, R. P.; Tang, J.; and Liu, H. 2017. Feature Selection: A Data Perspective. *ACM Computing Surveys*, 50(6): 1–45.

Liang, C.; Wang, L.; Liu, L.; Zhang, H.; and Guo, F. 2023. Multi-View Unsupervised Feature Selection with Tensor Robust Principal Component Analysis and Consensus Graph Learning. *Pattern Recognition*, 141: 109632.

Lim, H.; and Kim, D.-W. 2021. Pairwise Dependence-Based Unsupervised Feature Selection. *Pattern Recognition*, 111: 107663.

Lin, Y.; Gou, Y.; Liu, X.; Bai, J.; Lv, J.; and Peng, X. 2022. Dual Contrastive Prediction for Incomplete Multi-View Representation Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 4447–4461.

Mairal, J.; Bach, F.; Ponce, J.; and Sapiro, G. 2010. Online Learning for Matrix Factorization and Sparse Coding. *Journal of Machine Learning Research*, 11(1): 19–60.

Mi, Y.; Chen, H.; Luo, C.; Horng, S.-J.; and Li, T. 2024. Unsupervised Feature Selection with High-Order Similarity Learning. *Knowledge-Based Systems*, 285: 111317.

Nie, F.; Huang, H.; Cai, X.; and Ding, C. 2010. Efficient and Robust Feature Selection via Joint ℓ2,1-Norms Minimization. In *Proceedings of the 23th International Conference on Neural Information Processing Systems*, 1813–1821.

Sun, S. 2013. A Survey of Multi-View Machine Learning. *Neural Computing and Applications*, 23(7-8): 2031–2038.

Tang, C.; Zheng, X.; Liu, X.; Zhang, W.; Zhang, J.; Xiong, J.; and Wang, L. 2022a. Cross-View Locality Preserved Diversity and Consensus Learning for Multi-View Unsupervised Feature Selection. *IEEE Transactions on Knowledge and Data Engineering*, 34(10): 4705–4716.

Tang, C.; Zheng, X.; Liu, X.; Zhang, W.; Zhang, J.; Xiong, J.; and Wang, L. 2022b. Cross-View Locality Preserved Diversity and Consensus Learning for Multi-View Unsupervised Feature Selection. *IEEE Transactions on Knowledge and Data Engineering*, 34(10): 4705–4716.

Viklands, T. 2006. *Algorithms for the Weighted Orthogonal Procrustes Problem and Other Least Squares Problems*. Ph.D. diss., Datavetenskap.

Wang, H.; Yang, Y.; and Liu, B. 2019. GMC: Graph-Based Multi-View Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32(6): 1116–1129.

Xie, L.; Tao, D.; and Wei, H. 2016. Multi-View Exclusive Unsupervised Dimension Reduction for Video-Based Facial Expression Recognition. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 2217–2223.

Xu, Y.; Yin, Y.; Wang, J.; Wei, J.; Liu, J.; Yao, L.; and Zhang, W. 2021. Unsupervised Cross-View Feature Selection on Incomplete Data. *Knowledge-Based Systems*, 234: 107595.

Yuan, H.; Li, J.; Liang, Y.; and Tang, Y. Y. 2022. Multi-View Unsupervised Feature Selection with Tensor Low-Rank Minimization. *Neurocomputing*, 487: 75–85.

Zhang, R.; Nie, F.; Li, X.; and Wei, X. 2019. Feature Selection with Multi-View Data: A Survey. *Information Fusion*, 50: 158–167.

Zhao, J.; Xie, X.; Xu, X.; and Sun, S. 2017. Multi-View Learning Overview: Recent Progress and New Challenges. *Information Fusion*, 38: 43–54.

Zhou, S.; Song, P.; Yu, Y.; and Zheng, W. 2023. Structural Regularization Based Discriminative Multi-View Unsupervised Feature Selection. *Knowledge-Based Systems*, 272: 110601.