Information Fusion 96 (2023) 37–50

Contents lists available at ScienceDirect

Information Fusion

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

Full length article

Adaptive collaborative fusion for multi-view semi-supervised classification

Bingbing Jiang [a], Chenglong Zhang [a], Yan Zhong [b], Yi Liu [a], Yingwei Zhang [c], Xingyu Wu [d,*], Weiguo Sheng [a,*]

[a] School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China
[b] School of Mathematical Sciences, Peking University, Beijing 100871, China
[c] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100086, China
[d] School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China

ARTICLE INFO

Keywords:
Multi-view data fusion
Semi-supervised multi-view classification
Collaborative fusion
Adaptive graph fusion
Feature projection

ABSTRACT

Multi-view semi-supervised classification is inherently a challenging task in multi-view learning due to the lack of label information. Existing methods generally suffer from insufficient data fusion, expensive computation cost in the solution procedure and fail in tackling unseen samples directly, intensively limiting their applicability and efficiency in real scenarios. To address these issues, we propose an adaptive collaborative fusion method, seeking for an appropriate representation and fusion for multi-view data. The main advantage of the proposed method is that it simultaneously fuses both multiple feature projections and similarity graphs to learn a joint projection subspace as well as a unified similarity graph that fully preserve the correlation and distinction among views. Meanwhile, our method can coalesce different views in an adaptive-weighting manner, making the learned subspace more discriminative and facilitating label propagation on the fused graph. Furthermore, an acceleration strategy has been designed to reduce the computational complexity, thereby making the proposed method scalable to relatively large-scale data. Finally, an alternating optimization has been adopted to solve the formulated objective function. Extensive experiments on synthetic and real-world datasets are conducted to demonstrate the effectiveness and superiority of our proposed method.

1. Introduction

As data sources increase continuously, data with diverse feature representations become widely available in real-world applications. This kind of data is called multi-view data, in which each view corresponds to a feature representation that has an independent statistical property [1–4]. As a new learning paradigm, multi-view learning aims to make full use of the information from different feature representations to obtain a comprehensive representation and effective fusion for multi-view data, and has attracted considerable attention in recent years [5–8]. Depending on whether the labels of training data are involved or not, existing multi-view learning methods can be divided into three groups: supervised, unsupervised and semi-supervised [9–13]. In real-world applications, labeled data are usually scarce due to the expensive cost of manually labeling samples, whereas large amounts of unlabeled data are commonly available but have lower discriminative ability for class labels [14,15]. Therefore, numerous efforts have been made on multi-view semi-supervised learning, which jointly exploits unlabeled data and multiple views.

The key challenge of multi-view semi-supervised learning is how to effectively exploit the correlation and distinction between different views to enhance performance under the circumstance of abundant unlabeled data [16]. To tackle this problem, extensive research that explores the comprehensive information of multiple views has been carried out in recent years. A straightforward solution is to concatenate multiple views into one view and then handle multi-view data via single-view models. A popular approach in this direction is graph-based semi-supervised learning. Typical methods include label propagation [17] and flexible manifold embedding [18], which aim to propagate the label information from labeled data to unlabeled data according to the similarity structure of data. Considering that different views contain distinct features of data, such a feature concatenation scheme could discard the differences among multiple views, thus intensively degrading the effectiveness of models in multi-view scenarios. To make full use of multi-view data, the co-training regularization method [19], Laplacian regularization method [20] as well as their variants [21,22] were proposed to explore the consensus

* Corresponding authors.
E-mail addresses: jiangbb@hznu.edu.cn (B. Jiang), xingyuwu@mail.ustc.edu.cn (X. Wu), w.sheng@ieee.org (W. Sheng).

https://doi.org/10.1016/j.inffus.2023.03.002
Received 29 November 2022; Received in revised form 1 March 2023; Accepted 2 March 2023
Available online 8 March 2023

and complementarity among multiple views. These methods, however, are generally designed for binary classification, and are not directly applicable to multi-view multi-class problems.

Inheriting from the single-view label propagation, many forms of graph-based fusion models have been developed, which construct similarity graphs on each view separately and explicitly use view weights to incorporate them. For example, Karasuyama et al. propagated label information on multiple single-view graphs and eliminated irrelevant graphs by introducing a regularization term on the view weights [23]. Methods presented in [24,25] linearly integrated the processes of label propagation on different views. Apart from an extra kernel parameter involved in constructing graphs, these methods completely separate the graph construction from label propagation, impairing the reliability of graphs and finally limiting the effectiveness of label propagation across views. To alleviate this issue, Nie et al. proposed to fuse a unified similarity graph and perform the label propagation simultaneously [26–28]. Despite the good efforts, these graph fusion methods are incapable of directly predicting out-of-sample data directly, restricting their application scenarios [13,29]. Recently, Li et al. further developed a flexible multi-view semi-supervised model that can predict new-coming samples by learning a concatenated feature projection [8]. Additionally, benefiting from the linear regression model, the regression-based multi-view methods have been designed, which learn feature projections for different views and linearly fuse them linearly [30,31]. Zhuge et al. employed a unified regression target to learn multiple feature projections and discriminate the importance of various views via view weights [31]. By fusing multiple feature projections, predictions for out-of-sample data can be made. To avoid constructing similarity graphs, the regression-based methods do not consider the local similarity structure of data, which is essential for multi-view classification with scarce labeled samples.

Moreover, multi-view fusion methods proposed in [23–25,30,31] introduced a weight-related exponential or regularization parameter to control the distribution of view weights, so as to avoid the situation that only the best view has a significant weight. This weight-related parameter, however, is difficult to be tuned properly due to the lack of practical interpretations. In summary, these multi-view methods learn either similarity graphs or feature projections for each view separately, and merely consider the graph-level or projection-level information fusion, suffering from insufficient data fusion. Moreover, most graph-based methods involve in expensive computation cost during the process of training, degrading their applicability and efficiency for large-scale problems. To solve the aforementioned issues, we present an adaptive Collaborative Fusion for Multi-view Semi-supervised Classification (CFMSC). The main contributions of this paper are summarized as follows:

- Different from existing methods, in CFMSC, multiple feature projections and similarity graphs are simultaneously integrated via a collaborative fusion scheme, facilitating label propagation on the fused graph and thus enhancing the discrimination of learned projection subspace.
- Our method coalesces different views in an adaptive-weighting manner and learns the joint projection as well as the unified graph compatible across all views, thus avoiding the weight-related parameter while taking into consideration the correlation and distinction among multiple views.
- An acceleration strategy has been devised which can significantly reduce the computational complexity of CFMSC from $\mathcal{O}(n^3)$ to $\mathcal{O}(nm\log m + n(m+d)^2)$ for efficiently handling large-scale data, where n, m and d denote the numbers of samples, anchors and features, respectively.
- An effective alternate optimization with fast convergence has also been developed to solve the objective function of CFMSC, and extensive experiments have validated the effectiveness and efficiency of CFMSC.

Table 1
The notations used in this paper.

Notations	Descriptions
d_v	The feature dimension of the vth view
V	The number of views
$d = \sum_{v=1}^{V} d_v$	The total dimensionality of V views
c	The number of classes
l	The number of labeled data
$\boldsymbol{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_V]^T \in \mathbb{R}^{d\times n}$	The concatenated feature matrix of training data
$\boldsymbol{X}_v \in \mathbb{R}^{d_v\times n}$	The feature matrix of the vth view
$\boldsymbol{x}_i^v \in \mathbb{R}^{d_v\times 1}$	The ith sample in $\boldsymbol{X}_v$
$\boldsymbol{x}_i \in \mathbb{R}^{d\times 1}$	The ith sample
$\boldsymbol{W}_v \in \mathbb{R}^{d_v\times c}$	The feature projection matrix of the vth view
$\boldsymbol{Y}_l \in \mathbb{R}^{l\times c}$	Given label matrix of labeled data
$\boldsymbol{Y} = [\boldsymbol{Y}_l; \mathbf{0}]^T \in \mathbb{R}^{n\times c}$	Initial label matrix of training data
$\boldsymbol{F} \in \mathbb{R}^{n\times c}$	The prediction label matrix of training data
$\boldsymbol{F}_l \in \mathbb{R}^{l\times c}$	Prediction label matrix of labeled data
$\mathrm{f}_i \in \mathbb{R}^{c\times 1}$	The prediction label vector of $\boldsymbol{x}_i$

2. Notations and related works

In this section, we first introduce several basic notations throughout the paper, and then the previous methods closely related to our research are revisited.

2.1. Definitions and notation

Throughout the paper, vectors and matrices are written in boldface with lowercase and uppercase letters, respectively. $\|\boldsymbol{v}\|_2$ denotes the L_2-norm of a vector $\boldsymbol{v}$, $\|\boldsymbol{M}\|_F$ denotes the Frobenius norm of a matrix $\boldsymbol{M}$. $\mathbf{1}$ denotes a column vector of which elements are 1, and $\boldsymbol{I}_m$ denotes an $m\times m$ identify matrix. For simplicity, the notations frequently used in this paper are listed in Table 1.

2.2. Graph-based semi-supervised learning

As a popular semi-supervised learning paradigm, graph-based label propagation aims to propagate label information from labeled data to unlabeled data according to the similarity structure of data, which can be formulated as:

$$\min_{\boldsymbol{F}} \ \mathrm{Tr}(\boldsymbol{F}^T\boldsymbol{L}\boldsymbol{F}) + \mathrm{Tr}\Big((\boldsymbol{F}-\boldsymbol{Y})^T\boldsymbol{U}_n(\boldsymbol{F}-\boldsymbol{Y})\Big), \tag{1}$$

where $\boldsymbol{L} \in \mathbb{R}^{n\times n}$ denotes a graph Laplacian matrix, and $\boldsymbol{U}_n \in \mathbb{R}^{n\times n}$ is a predetermined diagonal matrix, whose ith diagonal element U_{ii} is a very large const (e.g., 10^6) if $\boldsymbol{x}_i$ is a labeled sample and 1 otherwise. To solve the out-of-sample data, Nie et al. [18] extended the label propagation and proposed a flexible manifold embedding method (FME), whose optimization objective is:

$$\min_{\boldsymbol{F},\boldsymbol{W},\boldsymbol{b}} \mathrm{Tr}(\boldsymbol{F}^T\boldsymbol{L}\boldsymbol{F}) + \mathrm{Tr}\Big((\boldsymbol{F}-\boldsymbol{Y})^T\boldsymbol{U}_n(\boldsymbol{F}-\boldsymbol{Y})\Big) + \gamma\Big(\|\boldsymbol{X}^T\boldsymbol{W} + \mathbf{1}\boldsymbol{b}^T - \boldsymbol{F}\|_F^2 + \mu\|\boldsymbol{W}\|_F^2\Big), \tag{2}$$

where γ and μ are regularization parameters to control different terms. FME explicitly uses the regression residue (i.e., $\boldsymbol{X}^T\boldsymbol{W} + \mathbf{1}\boldsymbol{b}^T - \boldsymbol{F}$) to encode the mismatch between the projection subspace and prediction labels.

2.3. Multi-view semi-supervised learning

In the past decade, multi-view semi-supervised learning especially graph-based models had attracted much attention. In this part, we will briefly introduce the state-of-the-art works which are closely related to our method.

(1) *Sparse Multiple Graph Integration (SMGI)* [23] propagates label information based on the linear combination of multiple single-view graphs. To control the distribution of the view weights $\{\alpha_v\}_{v=1}^V$, SMGI

explicitly introduces a parameter-involved regularization term, whose optimization objective is:

$$\min_{\boldsymbol{F},\alpha_v}\sum_{v=1}^{V}\Big(\frac{\alpha_v}{\|\boldsymbol{L}^v\|_F}\mathrm{Tr}(\boldsymbol{F}^T\boldsymbol{L}^v\boldsymbol{F})+\frac{\gamma}{2}\alpha_v^2\Big)+\lambda\|\boldsymbol{F}-\boldsymbol{Y}\|_F^2,\ \text{s.t.}\ \sum_{v=1}^{V}\alpha_v=1,\alpha_v\geq 0, \tag{3}$$

where $\boldsymbol{L}^v\in\mathbb{R}^{n\times n}$ denotes the Laplacian matrix for the vth view, and γ is the weight-related regularization parameter.

(2) *Multi-view Learning with Adaptive Neighbors (MLAN)* [26] can learn a unified graph $\boldsymbol{S}\in\mathbb{R}^{n\times n}$ with adaptive neighbors, whose objective function is:

$$\begin{aligned}&\min_{\boldsymbol{F},\boldsymbol{S},\alpha_v}\sum_{v=1}^{V}\Big(\sum_{i,j=1}^{n}\|\boldsymbol{x}_i^v-\boldsymbol{x}_j^v\|_2^2 s_{ij}\Big)^{\frac{p}{2}}+\gamma\|\boldsymbol{S}\|_F^2+\lambda\mathrm{Tr}\big(\boldsymbol{F}^T\boldsymbol{L}_S\boldsymbol{F}\big)\\&\text{s.t.}\ \sum_{j=1}^{n}s_{ij}=1,s_{ij}\geq 0,\boldsymbol{F}_l=\boldsymbol{Y}_l.\end{aligned} \tag{4}$$

Unlike SMGI, MLAN can balance different views without explicitly introducing extra parameters, in which p can be tuned from $(0,1)$.

(3) *Accelerated Manifold Embedding for Multi-view semi-supervised Classification (AMEMC)* [25] involves a weight-related exponential parameter θ to tune the distribution of $\{\alpha_v\}_{v=1}^V$. The objective of AMEMC is:

$$\min_{\boldsymbol{F},\alpha_v}\sum_{v=1}^{V}\alpha_v^\theta\mathrm{Tr}\Big(\boldsymbol{F}^T\widehat{\boldsymbol{L}}^v\boldsymbol{F}\Big)+\lambda\mathrm{Tr}\Big((\boldsymbol{F}-\boldsymbol{Y})^T(\boldsymbol{F}-\boldsymbol{Y})\Big),\ \text{s.t.}\ \sum_{v=1}^{V}\alpha_v=1,\alpha_v\geq 0, \tag{5}$$

where $\widehat{\boldsymbol{L}}^v\in\mathbb{R}^{n\times n}$ denotes the normalized graph Laplacian for the vth view, and $\theta\geq 1$ needs to be tuned manually. The computational complexity of AMEMC can be reduced by using the convergence of the matrix spectral radius.

(4) *Flexible Multi-view SEmi-supervised Learning (FMSEL)* [8] incorporates the multiple graph fusion into the framework of FME, formulated as:

$$\begin{aligned}&\min_{\boldsymbol{F},\boldsymbol{W},\boldsymbol{b},\alpha_v,\boldsymbol{S}}\Big\|\boldsymbol{S}-\sum_{v=1}^{V}\alpha_v\boldsymbol{G}^v\Big\|_F^2+\mathrm{Tr}\Big((\boldsymbol{F}-\boldsymbol{Y})^T\boldsymbol{U}(\boldsymbol{F}-\boldsymbol{Y})\Big)\\&+\lambda\mathrm{Tr}\big(\boldsymbol{F}^T\boldsymbol{L}_S\boldsymbol{F}\big)+\gamma\Big(\|\boldsymbol{X}^T\boldsymbol{W}+\boldsymbol{1}\boldsymbol{b}^T-\boldsymbol{F}\|_F^2+\mu\|\boldsymbol{W}\|_F^2\Big)\\&\text{s.t.}\ \sum_{j=1}^{n}s_{ij}=1,s_{ij}\geq 0,\ \sum_{v=1}^{V}\alpha_v=1,\alpha_v\geq 0.\end{aligned} \tag{6}$$

FMSEL performs the multi-view fusion from the aspect of graphs, and it can handle out-of-sample data with the concatenated projection $\boldsymbol{W}$ and bias $\boldsymbol{b}$.

(5) *Multi-View semi-supervised classification via Adaptive Regression (MVAR)* [30] focuses on the feature projection fusion and can predict labels of new samples via learning feature projections for different views. MVAR distinguishes different views in the same style as [25], whose objective function is:

$$\begin{aligned}&\min_{\boldsymbol{F},\boldsymbol{W}_v,\boldsymbol{b}_v,\alpha_v}\sum_{v=1}^{V}\alpha_v^\theta\Big(\sum_{i=1}^{n}s_i\Big\|\boldsymbol{W}_v^T\boldsymbol{x}_i^v+\boldsymbol{b}_v-\mathbf{f}_i\Big\|_2+\lambda_v\|\boldsymbol{W}_v\|_F^2\Big)\\&\text{s.t.}\sum_{v=1}^{V}\alpha_v=1,\alpha_v\geq 0,\boldsymbol{F}_l=\boldsymbol{Y}_l,\end{aligned} \tag{7}$$

where λ^v and $\boldsymbol{b}_v$ are the regularization parameter and bias term of the vth view, respectively. In MVAR, s_i is manually tuned to distinguish different samples.

3. The proposed methodology

3.1. Problem formulation

To explicitly deal with out-of-sample data, the least square regression is widely adopted, which can be extended to the multi-view scenario as follows:

$$\min_{\boldsymbol{W}_v,\boldsymbol{F}}\sum_{v=1}^{V}\Big\|\boldsymbol{X}_v^T\boldsymbol{W}_v-\boldsymbol{F}\Big\|_F^2+\lambda\|\boldsymbol{W}_v\|_F^2. \tag{8}$$

For simplicity, we can integrate $\boldsymbol{b}_v$ into $\boldsymbol{W}_v$ by adding an $\mathbf{1}$ vector as an additional row of $\boldsymbol{X}_v$. It can be noted that Eq. (8) neglects the distinctions between different views and treats the regression loss on each view equally, affecting its effectiveness when there exist low-quality views. Considering that the features of multiple views have diverse characteristics and contribute variously to model, the fusion manner in [25,30] can be used to balance different views:

$$\min_{\boldsymbol{W}_v,\boldsymbol{F}}\sum_{v=1}^{V}\pi_v^\theta\Big\|\boldsymbol{X}_v^T\boldsymbol{W}_v-\boldsymbol{F}\Big\|_F^2+\lambda\|\boldsymbol{W}_v\|_F^2, \tag{9}$$

where $\boldsymbol{\pi}=[\pi_1,\ldots,\pi_V]^T$ denotes the weight vector, and θ controlling the distribution of π_v is a weight-related exponential parameter. It can be observed that Eq. (9) has a trivial solution with respect to (w.r.t.) $\boldsymbol{\pi}$ when $\theta=1$, that is $\pi_v=1$ for the best view and $\pi_v=0$ for others. To avert this problem, existing multi-view fusion methods have to manually tune θ from the range of $(1,\infty)$.

Through some simple algebraic transformations, an effective and elegant fusion manner is tactfully derived, actively releasing Eq. (9) from the exponential parameter θ. Specifically, with fixed $\boldsymbol{W}_v$ and $\boldsymbol{\pi}$, we explore the latent relation between the prediction label $\boldsymbol{F}$ and multiple projection subspaces $\{\boldsymbol{X}_v^T\boldsymbol{W}_v\}_{v=1}^V$ by setting the derivation of Eq. (9) w.r.t. $\boldsymbol{F}$ to zero:

$$\sum_{v=1}^{V}\pi_v^\theta\big(\boldsymbol{F}-\boldsymbol{X}_v^T\boldsymbol{W}_v\big)=\boldsymbol{0}\Longrightarrow\begin{cases}\boldsymbol{F}=\sum_{v=1}^{V}\alpha_v\boldsymbol{X}_v^T\boldsymbol{W}_v\\ \alpha_v=\pi_v^\theta/\sum_{v=1}^{V}\pi_v^\theta\end{cases}. \tag{10}$$

In Eq. (10), $\alpha_v\geq 0$ actually works as a view weight and $\sum_{v=1}^V\alpha_v=1$. Considering that the linear combination relation, i.e., $\boldsymbol{F}=\sum_{v=1}^V\alpha_v\boldsymbol{X}_v^T\boldsymbol{W}_v$ may be overstrict for data residing on non-linear distribution, we relax the equality constraint on $\boldsymbol{F}$ by introducing a flexible regression residue (i.e., $\boldsymbol{F}-\sum_{v=1}^V\alpha_v\boldsymbol{X}_v^T\boldsymbol{W}_v$). Thus, the multi-view fusion model in Eq. (9) is transformed into:

$$\min_{\boldsymbol{W}_v,\alpha_v\geq 0,\mathbf{1}^T\boldsymbol{\alpha}=1}\Big\|\boldsymbol{F}-\sum_{v=1}^{V}\alpha_v\boldsymbol{X}_v^T\boldsymbol{W}_v\Big\|_F^2+\lambda\|\boldsymbol{W}_v\|_F^2, \tag{11}$$

where $\boldsymbol{\alpha}=[\alpha_1,\ldots,\alpha_V]\in\mathbb{R}^V$ is a weight vector. Different from traditional methods in the literature, Eq. (11) weights projection subspaces straightforward and learns feature projections as well as view weights, tactfully avoiding extra parameters (e.g., θ). In this way, α_v depends on the mismatch between $\boldsymbol{F}$ and $\boldsymbol{X}_v^T\mathbf{W}_v$ and can be adaptively optimized. Therefore, Eq. (11) performs data fusion from the level of feature projections in an adaptive-weighting manner.

Since each sample usually plays different roles in the training process, the contribution of different samples on regression losses should be taken into consideration [32]. Existing methods [30,33] associated each sample with an additional weight and manually tuned the weights for different samples, lacking a reasonable learning mechanism. To alleviate this issue, we further design a new multi-view model that discriminates different samples in the self-weighted manner without explicitly introducing the sample weight parameter, formulated as:

$$\min_{\boldsymbol{W}_v,\alpha_v\geq 0,\mathbf{1}^T\boldsymbol{\alpha}=1}\frac{1}{n}\sum_{i=1}^{n}\Big\|\mathbf{f}_i-\sum_{v=1}^{V}\alpha_v\boldsymbol{W}_v^T\boldsymbol{x}_i^v\Big\|_2+\lambda\sum_{v=1}^{V}\Big\|\boldsymbol{W}_v\Big\|_F^2. \tag{12}$$

We can prove that Eq. (12) essentially accounts for the following problem (the proof is given in Appendix A):

$$\min_{\boldsymbol{W}_v,\alpha_v\geq 0,\mathbf{1}^T\boldsymbol{\alpha}=1,q_i\geq 0,\mathbf{1}^T\boldsymbol{q}=n}\sum_{i=1}^{n}\frac{1}{q_i}\Big\|\mathbf{f}_i-\sum_{v=1}^{V}\alpha_v\boldsymbol{W}_v^T\boldsymbol{x}_i^v\Big\|_2^2+\lambda\sum_{v=1}^{V}\Big\|\boldsymbol{W}_v\Big\|_F^2, \tag{13}$$

where $\frac{1}{q_i}$ denotes the divisor weight of $\boldsymbol{x}_i$, and the vector $\boldsymbol{q} = [q_1, \ldots, q_n]$. It is worth noting that Eq. (13) will degenerate into Eq. (11) if each sample is assigned with the same weight (i.e., $q_i = 1$).

In multi-view semi-supervised scenarios, how to exploit the similarity structures of different views is also critical since this structure can enrich the label information of data by label propagation [22,34]. To capture and fuse multiple similarity structure of data, the same fusion manner as Eq. (11) can be adopted to integrate the similarity graphs predefined on each view, fulfilled as:

$$\min_{\boldsymbol{S}} \left\| \boldsymbol{S} - \sum_{v=1}^{V} \alpha_v \boldsymbol{S}^v \right\|_F^2, \text{ s.t. } \boldsymbol{S}\mathbf{1} = \mathbf{1}, \boldsymbol{S} \geq 0, \tag{14}$$

where $\boldsymbol{S}$ denotes a unified graph that compatibly crosses multiple views, $\{\alpha_v\}_{v=1}^V$ discriminate multiple feature projections and similarity graphs simultaneously, and the single-view graphs $\{\boldsymbol{S}^v\}_{v=1}^V$ can be previously generated by [35].

Consequently, a novel multi-view semi-supervised classification model via the adaptive collaborative fusion of feature projections and similarity graphs is achieved by incorporating the above two parts into label propagation, mathematically formulated as:

$$\begin{aligned} &\min_{\boldsymbol{F},\boldsymbol{S},\boldsymbol{W}_v,\boldsymbol{\alpha},\boldsymbol{q}} \sum_{i=1}^{n} \frac{1}{q_i} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2^2 + \lambda \sum_{v=1}^{V} \|\boldsymbol{W}_v\|_F^2 + \beta \mathrm{Tr}(\boldsymbol{F}^T \boldsymbol{L}_S \boldsymbol{F}) \\ &+ \gamma \left\| \boldsymbol{S} - \sum_{v=1}^{V} \alpha_v \boldsymbol{S}^v \right\|_F^2 + \mathrm{Tr}\big((\boldsymbol{F} - \boldsymbol{Y})^T \boldsymbol{U}_n (\boldsymbol{F} - \boldsymbol{Y})\big) \\ &\text{s.t. } \boldsymbol{S}\mathbf{1} = \mathbf{1}, \boldsymbol{S} \geq 0, \ \boldsymbol{\alpha} \geq 0, \boldsymbol{\alpha}^T \mathbf{1} = 1, \ \boldsymbol{q} \geq 0, \boldsymbol{q}^T \mathbf{1} = n, \end{aligned} \tag{15}$$

where $\boldsymbol{L}_S \in \mathbb{R}^{n \times n}$ denotes the Laplacian matrix of $\boldsymbol{S}$, and $\boldsymbol{U}_n \in \mathbb{R}^{n \times n}$ is a predetermined diagonal matrix. $\boldsymbol{L}_S = \boldsymbol{D}_S - \boldsymbol{S}$, where $\boldsymbol{D}_S$ is the diagonal degree matrix with its the ith diagonal element being $\sum_{j=1}^{n} s_{ij}$. λ, γ and β are the parameters to balance different terms. Different from existing methods that merely consider the fusion of feature projections or similarity graphs, CFMSC can achieve a comprehensive representation of multi-view data relying on the collaborative fusion on both projections and graphs. Moreover, this fusion manner is effective and parameter-free, balancing different views in an adaptive-weighting manner. Besides, by integrating the projection learning in Eq. (13) as well as the graph fusion and learning in Eq. (14) into a unified framework, the label information can be accurately propagated on the fused graph $\boldsymbol{S}$ so that the learned feature projections will be more discriminative for tackling new samples.

3.2. Alternate optimization

Noting that the objective function in (15) is difficult to be directly solved since it is not jointly convex concerning all variables. To achieve the optimal solution, the strategy that alternately optimizes each variable by fixing other variables is adopted to solve (15). Specifically, four relatively simple subproblems are separately solved to find the optimal $\boldsymbol{W}_v$, $\boldsymbol{F}$, $\boldsymbol{S}$, $\boldsymbol{\alpha}$ and $\boldsymbol{q}$.

Update $\boldsymbol{W}_v$ and $\boldsymbol{F}$: When other variables are fixed except $\boldsymbol{W}_v$ and $\boldsymbol{F}$, the view weight α_v can be merged into $\boldsymbol{W}_v$ as $\alpha_v \boldsymbol{W}_v = \widetilde{\boldsymbol{W}}_v$, in which $\widetilde{\boldsymbol{W}}_v$ denotes the weighted feature projection of the vth view. With fixed $\{\alpha_v\}_{v=1}^V$, solving $\{\boldsymbol{W}_v\}_{v=1}^V$ is equivalent to solving the joint weighted feature projection $\widetilde{\boldsymbol{W}} = [\widetilde{\boldsymbol{W}}_1, \ldots, \widetilde{\boldsymbol{W}}_V]^T \in \mathbb{R}^{d \times c}$. Accordingly, we should solve the following subproblem:

$$\begin{aligned} &\min_{\widetilde{\boldsymbol{W}},\boldsymbol{F}} \mathrm{Tr}\big((\boldsymbol{X}^T \widetilde{\boldsymbol{W}} - \boldsymbol{F})^T \boldsymbol{Q} (\boldsymbol{X}^T \widetilde{\boldsymbol{W}} - \boldsymbol{F})\big) + \lambda \mathrm{Tr}\big(\widetilde{\boldsymbol{W}}^T \boldsymbol{A}^{-1} \widetilde{\boldsymbol{W}}\big) \\ &+ \beta \mathrm{Tr}(\boldsymbol{F}^T \boldsymbol{L}_S \boldsymbol{F}) + \mathrm{Tr}\big((\boldsymbol{F} - \boldsymbol{Y})^T \boldsymbol{U}_n (\boldsymbol{F} - \boldsymbol{Y})\big), \end{aligned} \tag{16}$$

where $\boldsymbol{Q}$ is a diagonal matrix with the ith diagonal element being $\frac{1}{q_i}$, and $\boldsymbol{A} = \mathrm{diag}(\alpha_1^2, \ldots, \alpha_1^2, \alpha_2^2, \ldots, \alpha_2^2, \ldots, \alpha_V^2, \ldots, \alpha_V^2)$ with each α_v^2 ($v = 1, 2, \ldots, V$) repeating d_v times. According to the proof in Appendix B, the objective function in Eq. (16) is jointly convex w.r.t. $\widetilde{\boldsymbol{W}}$ and $\boldsymbol{F}$. To obtain the optimal solution, we first take the derivative of Eq. (16) w.r.t. $\widetilde{\boldsymbol{W}}$ and set it to zero:

$$\boldsymbol{X}\boldsymbol{Q}\boldsymbol{X}^T \widetilde{\boldsymbol{W}} - \boldsymbol{X}\boldsymbol{Q}\boldsymbol{F} + \lambda \boldsymbol{A}^{-1} \widetilde{\boldsymbol{W}} = 0 \Longrightarrow \widetilde{\boldsymbol{W}} = \boldsymbol{B}\boldsymbol{F}, \tag{17}$$

where $\boldsymbol{B} = (\boldsymbol{X}\boldsymbol{Q}\boldsymbol{X}^T + \lambda \boldsymbol{A}^{-1})^{-1} \boldsymbol{X}\boldsymbol{Q}$ if $d < n$ and $\boldsymbol{B} = \boldsymbol{A}\boldsymbol{X}(\boldsymbol{X}^T \boldsymbol{A}\boldsymbol{X} + \lambda \boldsymbol{Q}^{-1})^{-1}$ otherwise according to the matrix identity.[1] Based on the joint weighted feature projection $\widetilde{\boldsymbol{W}}$, the vth view feature projection $\boldsymbol{W}_v$ can be decoupled as $\frac{1}{\alpha_v} \widetilde{\boldsymbol{W}}_v$. Substituting $\widetilde{\boldsymbol{W}}$ of Eq. (17) into Eq. (16), we have the following subproblem:

$$\begin{aligned} &\min_{\boldsymbol{F}} \mathrm{Tr}\big((\boldsymbol{X}^T \boldsymbol{B}\boldsymbol{F} - \boldsymbol{F})^T \boldsymbol{Q} (\boldsymbol{X}^T \boldsymbol{B}\boldsymbol{F} - \boldsymbol{F})\big) + \lambda \mathrm{Tr}\big(\boldsymbol{F}^T \boldsymbol{B}^T \boldsymbol{A}^{-1} \boldsymbol{B}\boldsymbol{F}\big) \\ &+ \beta \mathrm{Tr}(\boldsymbol{F}^T \boldsymbol{L}_S \boldsymbol{F}) + \mathrm{Tr}\big((\boldsymbol{F} - \boldsymbol{Y})^T \boldsymbol{U}_n (\boldsymbol{F} - \boldsymbol{Y})\big). \end{aligned} \tag{18}$$

By setting the derivative of Eq. (18) w.r.t. $\boldsymbol{F}$ to zero, $\boldsymbol{F}$ is solved as follows:

$$\boldsymbol{F} = (\boldsymbol{Q} - \boldsymbol{B}^T \boldsymbol{X}\boldsymbol{Q} + \beta \boldsymbol{L}_S + \boldsymbol{U}_n)^{-1} \boldsymbol{U}_n \boldsymbol{Y}. \tag{19}$$

With the optimal solution of $\boldsymbol{F}$, the class of unlabeled samples can be determined by $\arg\max_{1 \leq j \leq c} F_{ij}$ ($\forall i = l+1, \ldots, n$).

Update $\boldsymbol{S}$: By fixing other variables, the optimization problem for $\boldsymbol{S}$ is:

$$\min_{\boldsymbol{S}\mathbf{1}=\mathbf{1}, \boldsymbol{S} \geq 0} \sum_{i=1}^{n} \left\| \boldsymbol{s}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{s}_i^v \right\|_2^2 + \frac{\beta}{2\gamma} \sum_{i,j=1}^{n} \left\| \mathbf{f}_i - \mathbf{f}_j \right\|_2^2 s_{ij}, \tag{20}$$

where $\boldsymbol{s}_i$ and $\boldsymbol{s}_i^v$ are the ith rows of $\boldsymbol{S}$ and $\boldsymbol{S}^v$, respectively. Noting that the optimization in Eq. (20) is independent for each row, thus we solve $\boldsymbol{S}$ by rows:

$$\min_{\boldsymbol{s}_i \mathbf{1}=1, s_{ij} \geq 0} \left\| \boldsymbol{s}_i + \frac{1}{2\gamma} \boldsymbol{d}_i \right\|_2^2, \tag{21}$$

where $\boldsymbol{d}_i$ is a row vector with $d_{ij} = \frac{\beta}{2} \|\mathbf{f}_i - \mathbf{f}_j\|_2^2 - \gamma \sum_{v=1}^{V} \alpha_v s_{ij}^v$. Eq. (21) can be efficiently solved with a closed-form solution [35].

Update $\boldsymbol{\alpha}$: By fixing other variables, we have the following problem:

$$\min_{\boldsymbol{\alpha} \geq 0, \boldsymbol{\alpha}^T \mathbf{1}=1} \left\| \hat{\boldsymbol{F}} - \sum_{v=1}^{V} \alpha_v \hat{\boldsymbol{P}}_v \right\|_F^2 + \gamma \left\| \boldsymbol{S} - \sum_{v=1}^{V} \alpha_v \boldsymbol{S}^v \right\|_F^2, \tag{22}$$

where $\hat{\boldsymbol{F}} = \boldsymbol{Q}^{\frac{1}{2}} \boldsymbol{F}$ and $\hat{\boldsymbol{P}}_v = \boldsymbol{Q}^{\frac{1}{2}} \boldsymbol{X}_v^T \boldsymbol{W}_v$. Converting $\hat{\boldsymbol{F}}$, $\boldsymbol{S}$, $\hat{\boldsymbol{P}}_v$, and $\boldsymbol{S}^v$ into the vector forms, i.e., $\mathbf{v}_1 = \mathrm{vec}(\hat{\boldsymbol{F}}) \in \mathbb{R}^{nc \times 1}$, $\mathbf{v}_2 = \mathrm{vec}(\boldsymbol{S}) \in \mathbb{R}^{n^2 \times 1}$, $\mathrm{vec}(\hat{\boldsymbol{P}}_v) \in \mathbb{R}^{nc \times 1}$, and $\mathrm{vec}(\boldsymbol{S}^v) \in \mathbb{R}^{n^2 \times 1}$, then the optimization problem for $\boldsymbol{\alpha}$ can be transformed into:

$$\min_{\boldsymbol{\alpha} \geq 0, \boldsymbol{\alpha}^T \mathbf{1}=1} \boldsymbol{\alpha}^T \boldsymbol{M} \boldsymbol{\alpha} - \boldsymbol{\alpha}^T \boldsymbol{h}, \tag{23}$$

where $\boldsymbol{M} = \boldsymbol{P}^T \boldsymbol{P} + \gamma \boldsymbol{S}_b^T \boldsymbol{S}_b$, $\boldsymbol{h} = 2(\boldsymbol{P}^T \mathbf{v}_1 + \gamma \boldsymbol{S}_b^T \mathbf{v}_2)$, $\boldsymbol{P} = [\mathrm{vec}(\hat{\boldsymbol{P}}_1), \cdots, \mathrm{vec}(\hat{\boldsymbol{P}}_V)]$ and $\boldsymbol{S}_b = [\mathrm{vec}(\boldsymbol{S}^1), \cdots, \mathrm{vec}(\boldsymbol{S}^V)]$. Due to the semi-definite $\boldsymbol{M}$, Eq. (23) is a quadratic convex programming problem and can be directly solved. The detailed derivation is given in Appendix C.

Update $\boldsymbol{q}$: By fixing the other variables, the optimization for $\boldsymbol{q}$ is:

$$\min_{q_i \geq 0, \boldsymbol{q}^T \mathbf{1}=n} \sum_{i=1}^{n} \frac{e_i^2}{q_i}, \tag{24}$$

where $e_i = \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2$. With simple algebraic manipulations, the optimal solution of q_i can be derived as follows:

$$q_i = \frac{n e_i}{\sum_{i=1}^{n} e_i}. \tag{25}$$

Thus, the value of q_i can be automatically determined according to the ratio of the regression loss on sample $\boldsymbol{x}_i$ to the sum of regression losses on all samples. Obviously, a smaller regression loss e_i means a smaller q_i, corresponding to a larger divisor weight $\frac{1}{q_i}$. In other words, the samples with larger regression losses will contribute less to the objective function and have smaller divisor weights, thereby enabling the proposed model to reduce the effect of potential noises.

[1] $(\boldsymbol{P}^{-1} + \boldsymbol{B}^T \boldsymbol{R}^{-1} \boldsymbol{B})^{-1} \boldsymbol{B}^T \boldsymbol{R}^{-1} = \boldsymbol{P}\boldsymbol{B}^T (\boldsymbol{B}\boldsymbol{P}\boldsymbol{B}^T + \boldsymbol{R})^{-1}$.

Algorithm 1 Optimization Algorithm for CFMSC

Input: Data $\boldsymbol{X}=[\boldsymbol{X}_1,\cdots,\boldsymbol{X}_V]^T$, labels $\boldsymbol{Y}_L$ of labeled data, graphs $\{\boldsymbol{S}^v\}_{v=1}^V$, and parameters λ, β and γ.

1: Initialize $\alpha_v=\frac{1}{V}$, and $\boldsymbol{S}=\sum_{v=1}^V \boldsymbol{S}^v/V$;
2: **repeat**
3: Update $\widetilde{\boldsymbol{W}}$ by Eq. (17), and calculate each $\boldsymbol{W}_v$ as $\frac{1}{\alpha_v}\widetilde{\boldsymbol{W}}_v$;
4: Update $\boldsymbol{F}$ by Eq. (19) or Eq. (34) for large-scale data;
5: Update each row of $\boldsymbol{S}$ by solving Eq. (21);
6: Update α by solving Eq. (23) ;
7: Update q_i by Eq. (25) ;
8: **until** Convergence

Output: The prediction label $\boldsymbol{F}$, the feature projections $\{\boldsymbol{W}_v\}_{v=1}^V$, and the view weight factor α.

With solving the subproblems of $\{\boldsymbol{W}_v\}_{v=1}^V$, $\boldsymbol{F}$, $\boldsymbol{S}$, $\boldsymbol{\alpha}$ and $\boldsymbol{q}$ in Eq. (15) separately, we further summarize the whole optimization steps in Algorithm 1. Here, we further analyze the computational complexity of CFMSC. Firstly, calculating $\widetilde{\boldsymbol{W}}$ and $\boldsymbol{F}$ involve the inverse operation of matrices, taking $\mathcal{O}\big(nd*\min(n,d)\big)$ and $\mathcal{O}(n^3)$ respectively in each iteration. For the optimization of $\boldsymbol{\alpha}$, it usually takes $\mathrm{poly}(V)$ to solve Eq. (23), which is neglectable since V is usually small in practice. For $\boldsymbol{S}$, it first needs to compute $\boldsymbol{d}_i$, then updates the similarities by Eq. (21), taking $\mathcal{O}(n^2d+n^2\log n)$. Besides, updating $\boldsymbol{q}$ also requires $\mathcal{O}(ndc)$. Due to $c\ll n$ and $V\ll n$, CFMSC approximately costs $\mathcal{O}(n^3+n^2\log n+n^2d+nd*\min(n,d))$ in each iteration. Generally, CFMSC is comparable to the state-of-the-art methods [8,26,27,30,31] in terms of computational complexity. From Algorithm 1, we can see that $\widetilde{\boldsymbol{W}}$ relies on $\boldsymbol{F}$ due to $\widetilde{\boldsymbol{W}}=\boldsymbol{B}\boldsymbol{F}$, which means that the similarity structures of data can be delivered into the feature projection $\widetilde{\boldsymbol{W}}$. Therefore, CFMSC can directly make accurate predictions for out-of-sample data without rebuilding similarity graphs and retraining models. Specifically, for a newly coming sample $\hat{\boldsymbol{x}}\in\mathbb{R}^{d\times 1}$, let $\boldsymbol{e}=[e_1,\ldots,e_c]^T=\widetilde{\boldsymbol{W}}^T\hat{\boldsymbol{x}}$, then its prediction label $y_{\hat{\boldsymbol{x}}}$ will be determined by $y_{\hat{\boldsymbol{x}}}=\arg\max_{1\le i\le c} e_i$.

3.3. Accelerating CFMSC with the anchor-based bipartite graph

In CFMSC, computing the inverse of an $n\times n$ dense matrix and learning the similarity graph requires high computation and storage costs, making it unbearable for large-scale data. Inspired by the anchor-based bipartite graph learning [36,37], an accelerated strategy is further designed to enhance the computation efficiency of CFMSC. Concretely, we first use the k-means to generate m ($m\ll n$) clustering centers of data as the anchor points, then construct the $n\times m$ bipartite graphs to depict the similarity relations between training samples and anchors, taking the computational complexity of $\mathcal{O}(nmd)$ and $\mathcal{O}(nm\log m+nmd)$, respectively. Let $\boldsymbol{V}\in\mathbb{R}^{d\times m}$ denote the generated anchors that can be regarded as the unlabeled data, and $\boldsymbol{G}\in\mathbb{R}^{m\times c}$ denotes the corresponding prediction label of $\boldsymbol{V}$, which is an auxiliary variable. With the generated m anchors, the $n\times n$ full graphs (i.e., $\{\boldsymbol{S}^v\}_{v=1}^V$ and $\boldsymbol{S}$) turn into the $n\times m$ bipartite graphs, focusing on the similarities between $\boldsymbol{X}$ and $\boldsymbol{V}$. Accordingly, we can define the augmented graph of bipartite graph $\boldsymbol{S}$ as follows:

$$\hat{\boldsymbol{S}}=\begin{bmatrix}\boldsymbol{0} & \boldsymbol{S}\\ \boldsymbol{S}^T & \boldsymbol{0}\end{bmatrix}\in\mathbb{R}^{(n+m)\times(n+m)}. \tag{26}$$

With augmented graph $\hat{\boldsymbol{S}}$, the objective function of CFMSC is reformulated as:

$$\begin{aligned}\min_{\widetilde{\boldsymbol{W}},\boldsymbol{F},\boldsymbol{S},\boldsymbol{\alpha},\boldsymbol{q}}\ & \mathrm{Tr}\big((\boldsymbol{X}^T\widetilde{\boldsymbol{W}}-\boldsymbol{F})^T\boldsymbol{Q}(\boldsymbol{X}^T\widetilde{\boldsymbol{W}}-\boldsymbol{F})\big)+\lambda\mathrm{Tr}\big(\widetilde{\boldsymbol{W}}^T\boldsymbol{A}^{-1}\widetilde{\boldsymbol{W}}\big)\\ &+\beta\mathrm{Tr}(\boldsymbol{H}^T\boldsymbol{L}_{\hat{\boldsymbol{S}}}\boldsymbol{H})+\gamma\Big\|\boldsymbol{S}-\sum_{v=1}^V\alpha_v\boldsymbol{S}^v\Big\|_F^2+\mathrm{Tr}\big((\boldsymbol{H}-\hat{\boldsymbol{Y}})^T\boldsymbol{U}(\boldsymbol{H}-\hat{\boldsymbol{Y}})\big)\\ \text{s.t.}\ & \boldsymbol{S}\boldsymbol{1}=\boldsymbol{1},\boldsymbol{S}\ge 0,\ \boldsymbol{\alpha}\ge 0,\boldsymbol{\alpha}^T\boldsymbol{1}=1,\ \boldsymbol{q}\ge 0,\boldsymbol{q}^T\boldsymbol{1}=1,\end{aligned} \tag{27}$$

where $\boldsymbol{H}=\begin{bmatrix}\boldsymbol{F}\\ \boldsymbol{G}\end{bmatrix}\in\mathbb{R}^{(n+m)\times c}$, $\hat{\boldsymbol{Y}}=\begin{bmatrix}\boldsymbol{Y}\\ \boldsymbol{0}\end{bmatrix}\in\mathbb{R}^{(n+m)\times c}$, $\boldsymbol{U}=\begin{bmatrix}\boldsymbol{U}_n & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}\in\mathbb{R}^{(n+m)\times(n+m)}$ and $\boldsymbol{L}_{\hat{\boldsymbol{S}}}\in\mathbb{R}^{(n+m)\times(n+m)}$. According to the definition of $\hat{\boldsymbol{S}}$ in Eq. (26), we have:

$$\boldsymbol{L}_{\hat{\boldsymbol{S}}}=\begin{bmatrix}\boldsymbol{D}_{\boldsymbol{S}} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{\Lambda}\end{bmatrix}-\begin{bmatrix}\boldsymbol{0} & \boldsymbol{S}\\ \boldsymbol{S}^T & \boldsymbol{0}\end{bmatrix}=\begin{bmatrix}\boldsymbol{I}_n & -\boldsymbol{S}\\ -\boldsymbol{S}^T & \boldsymbol{\Lambda}\end{bmatrix}, \tag{28}$$

where $\boldsymbol{\Lambda}\in\mathbb{R}^{m\times m}$ is a diagonal matrix whose diagonal elements are column sums of the bipartite graph $\boldsymbol{S}$. With Eq. (27), the optimization for $\boldsymbol{S}$ becomes:

$$\min_{\boldsymbol{S}\boldsymbol{1}=\boldsymbol{1},\boldsymbol{S}\ge 0}\sum_{i=1}^n\Big\|\boldsymbol{S}-\sum_{v=1}^V\alpha_v\boldsymbol{S}^v\Big\|_2^2+\frac{\beta}{\gamma}\sum_{i=1}^n\sum_{j=1}^m\|\mathbf{f}_i-\mathbf{g}_j\|_2^2 s_{ij}, \tag{29}$$

where $\mathbf{g}_j$ denotes the jth row of $\boldsymbol{G}$. In Eq. (29), constructing each bipartite graph $\boldsymbol{S}^v$ takes $\mathcal{O}(nm\log m+nmd_v)$, and solving $\boldsymbol{S}$ by rows takes $\mathcal{O}(nm)$, making the graph learning scalable well with the data size. In Eq. (27), the optimization problems of $\widetilde{\boldsymbol{W}}$, $\boldsymbol{\alpha}$ and $\boldsymbol{q}$ are unchanged, which means that they can be directly solved via the corresponding steps in Algorithm 1. As for $\boldsymbol{F}$, the optimization problem is transformed into:

$$\begin{aligned}&\min\mathrm{Tr}\big((\boldsymbol{X}^T\boldsymbol{B}\boldsymbol{F}-\boldsymbol{F})^T\boldsymbol{Q}(\boldsymbol{X}^T\boldsymbol{B}\boldsymbol{F}-\boldsymbol{F})\big)+\lambda\mathrm{Tr}\big(\boldsymbol{F}^T\boldsymbol{B}^T\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{F}\big)\\ &+\beta\mathrm{Tr}\left(\begin{bmatrix}\boldsymbol{F}\\ \boldsymbol{G}\end{bmatrix}^T\begin{bmatrix}\boldsymbol{I}_n & -\boldsymbol{S}\\ -\boldsymbol{S}^T & \boldsymbol{\Lambda}\end{bmatrix}\begin{bmatrix}\boldsymbol{F}\\ \boldsymbol{G}\end{bmatrix}\right)+\mathrm{Tr}\left(\begin{bmatrix}\boldsymbol{F}-\boldsymbol{Y}\\ \boldsymbol{G}\end{bmatrix}^T\begin{bmatrix}\boldsymbol{U}_n & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{F}-\boldsymbol{Y}\\ \boldsymbol{G}\end{bmatrix}\right).\end{aligned} \tag{30}$$

Firstly, taking the derivative of Eq. (30) w.r.t. $\boldsymbol{G}$ to zero, we obtain:

$$\boldsymbol{\Lambda}\boldsymbol{G}-\boldsymbol{S}^T\boldsymbol{F}=0\Longrightarrow\boldsymbol{G}=\boldsymbol{\Lambda}^{-1}\boldsymbol{S}^T\boldsymbol{F}, \tag{31}$$

Then, substituting $\boldsymbol{G}=\boldsymbol{\Lambda}^{-1}\boldsymbol{S}^T\boldsymbol{F}$ into Eq. (30) and setting its derivative w.r.t. $\boldsymbol{F}$ to zero, we can obtain the solution of $\boldsymbol{F}$:

$$\boldsymbol{F}=\Big(\boldsymbol{Q}+\beta\boldsymbol{I}_n+\boldsymbol{U}_n-\boldsymbol{Q}\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{Q}\boldsymbol{X}^T+\lambda\boldsymbol{A}^{-1})^{-1}\boldsymbol{X}\boldsymbol{Q}-\beta\boldsymbol{S}\boldsymbol{\Lambda}^{-1}\boldsymbol{S}^T\Big)^{-1}\boldsymbol{U}_n\boldsymbol{Y}. \tag{32}$$

Eq. (32) also involves the inverse operation of an $n\times n$ matrix, limiting its application for large-scale data. To make Eq. (32) adapt to large-scale problems, the quadratic form in Eq. (32) can be reformulated as follows:

$$\boldsymbol{Q}\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{Q}\boldsymbol{X}^T+\lambda\boldsymbol{A}^{-1})^{-1}\boldsymbol{X}\boldsymbol{Q}+\beta\boldsymbol{S}\boldsymbol{\Lambda}^{-1}\boldsymbol{S}^T=\boldsymbol{C}\boldsymbol{D}\boldsymbol{C}^T, \tag{33}$$

where $\boldsymbol{C}=\begin{bmatrix}\boldsymbol{Q}\boldsymbol{X}^T & \boldsymbol{S}\end{bmatrix}\in\mathbb{R}^{n\times(m+d)}$ and $\boldsymbol{D}=\begin{bmatrix}(\boldsymbol{X}\boldsymbol{Q}\boldsymbol{X}^T+\lambda\boldsymbol{A}^{-1})^{-1} & \\ & \beta\boldsymbol{\Lambda}^{-1}\end{bmatrix}\in\mathbb{R}^{(m+d)\times(m+d)}$. According to the Woodbury matrix identity,[2] Eq. (32) can be simplified as:

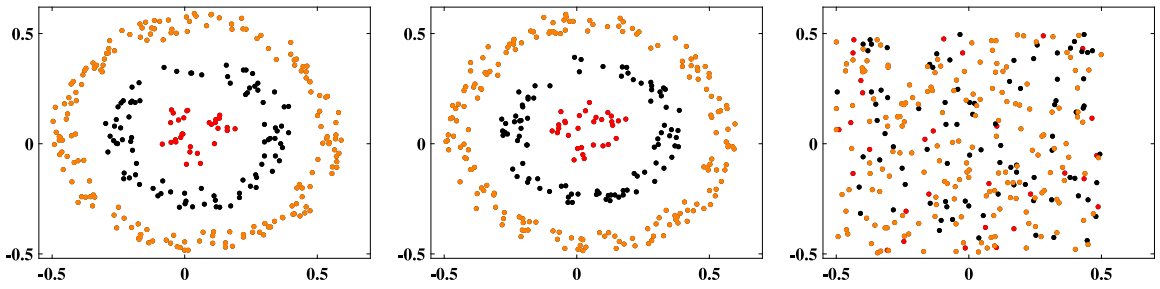
$$\boldsymbol{F}=\boldsymbol{V}^{-1}\boldsymbol{U}_n\boldsymbol{Y}+\boldsymbol{V}^{-1}\boldsymbol{C}\Big(\boldsymbol{D}^{-1}-\boldsymbol{C}^T\boldsymbol{V}^{-1}\boldsymbol{C}\Big)^{-1}\boldsymbol{C}^T\boldsymbol{V}^{-1}\boldsymbol{U}_n\boldsymbol{Y}, \tag{34}$$

where $\boldsymbol{V}=\boldsymbol{Q}+\beta\boldsymbol{I}_n+\boldsymbol{U}_n$. Eq. (34) substitutes the inverse operation of an $n\times n$ dense matrix in Eq. (32) with the inverse of a diagonal matrix, the inverse of an $(m+d)\times(m+d)$ matrix, as well as several matrix multiplications. Specifically, the first term of Eq. (34) takes $\mathcal{O}(nc)$ since $\boldsymbol{V}$ and $\boldsymbol{U}_n$ are both diagonal matrices, and the second term is computed one by one from right to left, taking $\mathcal{O}(nc(m+d)+n(m+d)^2+c(m+d)^2+(m+d)^3)$. Due to $m+d\ll n$ for large-scale data, our accelerated strategy can reduce the main computational complexity from $\mathcal{O}(n^3+n^2\log n+n^2d+nd^2)$ to $\mathcal{O}(nm\log m+n(m+d)^2+(m+d)^3)$, which is approximately linear to the number of training samples n.

4. Experiments

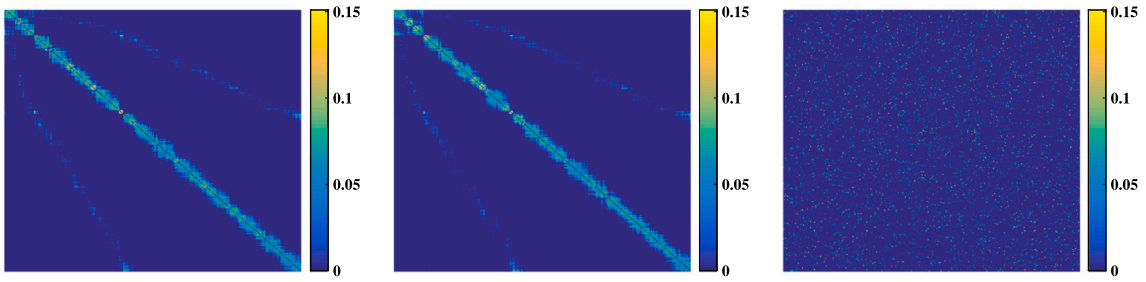
In this section, extensive experiments are conducted to verify the effectiveness and superiority of the proposed CFMSC. Specifically, we first use synthetic data to visually demonstrate what capacities our multi-view fusion model exactly possesses. Secondly, we compare CFMSC with other state-of-the-art methods to further evaluate its

[2] $(\boldsymbol{J}+\boldsymbol{K}\boldsymbol{R}\boldsymbol{K}^T)^{-1}=\boldsymbol{J}^{-1}-\boldsymbol{J}^{-1}\boldsymbol{K}(\boldsymbol{R}^{-1}+\boldsymbol{K}^T\boldsymbol{J}^{-1}\boldsymbol{K})^{-1}\boldsymbol{K}^T\boldsymbol{J}^{-1}$



(a) The first view (view 1) (b) The second view (view 2) (c) The third view (view 3)

Fig. 1. The data distributions of Three-Ring.



(a) The graph matrix on view 1 (b) The graph matrix on view 2 (c) The graph matrix on view 3

Fig. 2. The visualization results of the initialized graph matrices on Three-Ring data.

classification performance and efficiency on real multi-view datasets with different data sizes. Finally, we analyze the parameter sensitivity and convergence of CFMSC. All experiments are implemented in Matlab R2016a and run on a Windows 10 computer with a 3.00 GHz Intel Core i7-9700 CPU and 16 GB RAM.

4.1. Experiments on synthetic data

Since the features of different views have diverse data distributions, the single-view graphs constructed from different views reflect multiple structures of data. To verify the capability of CFMSC in learning the unified graph across multiple views as well as the discriminability against noisy views, we followed [38] to randomly generate synthetic data (i.e., Three-Ring), consisting of three views from three different classes. The first two views have a relatively clear shape, shown in Figs. 1(a) and 1(b), and the features of the third view are all noises, shown in Fig. 1(c). Fig. 2 shows the visualization results of the initialized graph matrices on three views.

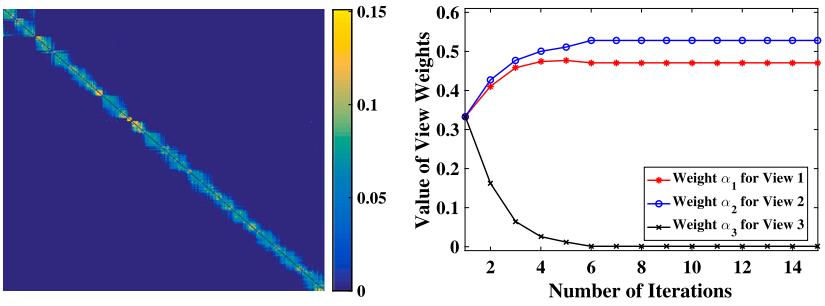
From the results in Fig. 1, it can be seen that the data distributions of the first two views and the third view vary obviously. Although the graph matrices of views 1 and 2 shown in Fig. 2 can roughly reflect the structure of the original data, there still exist several connections between different classes. Furthermore, the graph matrix of view 3 seems to be randomly filled without any similarity structures, which indicates that the noisy view makes the samples belonging to different classes difficult to be separated. By using the proposed model, the learned unified graph is shown in Fig. 3(a). From this figure, we observe that there are much fewer inter-class connections, demonstrating that the unified graph $\boldsymbol{S}$ is more effective for label propagation and classification than the single-view graphs of Fig. 1. Accordingly, Fig. 3(b) shows the weights assigned to three views after each iteration. We finally conclude that CFMSC not only utilizes the similarity structures across multiple views to learn a unified graph but also adaptively assigns appropriate weights to different views such that the noisy views associated with small weights cannot degrade the performance.

4.2. Experiments on multi-view datasets

4.2.1. Datasets and experimental settings

Towards the evaluation of the proposed CFMSC, several real multi-view datasets are used in this part, including MSRC-v1, Handwritten (HW), Caltech101-7 (Cal-7), ORL, COIL20, Leaves, Hdigit, and MNIST. Specifically, MSRC-v1 is an object recognition dataset and includes 7 different categories, with each category having 30 images. HW is generated from the UCI machine learning repository and contains 2000 samples, in which each sample is represented by six different view features. Cal-7 is the frequently used subset of the object recognition dataset Caltech101, containing 1474 samples from 7 classes. ORL is comprised of the face images of 40 different subjects with 10 images for each subject. COIL20 consists of 1440 images with 72 samples per class, in which each sample has four different views. The Leaves dataset is comprised of the leaf images from 100 different plants with 16 images for each plant. Hdigit and MNIST are two relatively large-scale datasets, in which Hdigit is collected from two sources and consists of 10000 samples, and MNIST has 70 000 samples. The detailed information of each dataset is summarized in Table 2.

We first compare CFMSC with a representative single-view method (i.e., FME) [18] that treats different views equally to examine whether the proposed multi-view collaborative fusion scheme can enhance learning performance. Then, CFMSC is compared with seven state-of-the-art multi-view methods, including multi-view adaptive regression (MVAR) [30], joint consensus and diversity (JCD) [31], multi-view learning with adaptive neighbors (MLAN) [26], flexible multi-view semi-supervised learning (FMSEL) [8], sparse multiple graph integration (SMGI) [23], accelerated manifold embedding for multi-view



(a) The adaptively fused graph matrix (b) The allocated weights for views

Fig. 3. The graph matrix and the view weights learned by our multi-view fusion model.

Table 2
The detailed description of multi-view datasets.

View	MSRC-v1	HW	Cal-7	ORL	COIL20	Leaves	Hdigit	MNIST
#1	1302	240	48	512	512	64	784	256
#2	48	76	40	59	420	64	256	144
#3	512	216	254	864	1239	64	–	59
#4	100	47	1984	254	630	–	–	–
#5	256	64	512	–	–	–	–	–
#6	200	6	928	–	–	–	–	–
Feature size	2418	649	3766	1689	2801	192	1040	459
Classes	7	10	7	40	20	100	10	10
Data size	210	2000	1474	400	1440	1600	10000	70000

semi-supervised classification (AMEMC) [25], and robust adaptive-weighting multi-view classification (RAMC) [39]. RAMC is a supervised method, which is employed to validate if using unlabeled data can improve the effectiveness of multi-view semi-supervised classification.

For each dataset except MNIST, 70% of samples are randomly selected for training, and the rest of the 30% of samples are used for testing. On the MNIST dataset, we randomly select 20 000 samples to make up the training set, and the remaining samples are testing data. To mimic the real situation, each training set is also randomly divided into the labeled subset and unlabeled subset by varying the labeled ratio from 10% to 30% except Hdigit and MNIST. On the Hdigit and MNIST datasets, only 1% to 3% of training samples are randomly selected to assign class labels. To ensure a fair comparison, we tune the parameters of all compared methods in the same way as described in their respective literature, in which the regularization parameter is tuned from $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$, and the weight-related parameter is tuned from {1.5, 2.0, 2.5, 3.0}. In CFMSC, the λ, β and γ are also tuned from $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. To reduce the computational complexity of CFMSC on the Hdigit and MNIST datasets, the acceleration strategy proposed in Section 3.3 is applied to construct the anchor-based bipartite graph to replace the $n \times n$ entire graph. The number of anchors m is set according to the number of samples n in different datasets. Specifically, k-means is used to generate 200 anchor points on the Hdigit dataset and 500 anchor points on the MNIST dataset. To reduce the statistical variability, all methods are independently run 20 times on different training and testing sets, and the means and variances of classification results on labeled and unlabeled data are recorded.

4.2.2. Experimental results on multi-view datasets

Comparison to Single-view Method. To demonstrate the superiority of CFMSC over the representative single-view method, we first compare the performance of CFMSC and FME with different percentages of labeled data. The classification results are shown in Figs. 4 and 5, where S-FME and C-FME denote the best results of FME on each view and the results by using the concatenated features, respectively.

From these figures, we observe that CFMSC consistently outperforms the single-view methods (i.e., S-FME and C-FME) on all datasets, which indicates that feature representations of different views are distinct and CFMSC can obtain more informative knowledge from multiple views. Meanwhile, CFMSC is superior to C-FME that directly concatenates multiple views, and C-FME achieves better performance than S-FME on all datasets except for Cal-7 and ORL, showing that the effective fusion of feature information from different views can greatly enhance the classification performance, whereas the improper feature concatenation might degrade the performance. Consequently, compared to S-FME which uses the single-view features, and C-FME which indiscriminately uses multiple feature representations, CFMSC can effectively utilize the correlations and distinctions among views as well as coordinate different views via adaptively assigning appropriate weights to them, such that the excellent views are emphasized while the poor views are weakened, achieving better representation and fusion of multi-view data.

Comparison to Multi-view Methods. To further validate the effectiveness of CFMSC, we compare it to the state-of-the-art multi-view methods. The classification accuracies of CFMSC and other competitors on unlabeled samples and testing samples are respectively recorded in Tables 3 and 4, in which "OM" denotes an out-of-memory error while running the experiment. The running time of multi-view semi-supervised methods on eight datasets is provided in Table 5. The proposed CFMSC shows better or highly competitive performance in comparison with the state-of-the-art competitors on all datasets and also achieves different levels of improvement on unlabeled and testing samples. Specifically, CFMSC is considerably superior to the multi-view supervised classification method (i.e., RAMC). Taking the Leaves dataset as an example, CFMSC respectively achieves 11.75%, 10.65% and 7.49% average improvements for the testing samples compared with RAMC with varying the ratio of labeled samples from 10% to 30%, fully validating that mining the similarity structure of unlabeled samples can enhance the performance. Compared with the multi-view semi-supervised methods that focus on the similarity graph fusion (i.e., MLAN, FMSEL, SMGI and AMEMC) or the feature projection fusion (i.e., MVAR and JCD), CFMSC gains the competitive performance on most datasets, showing its powerful effectiveness in multi-view classification. On the MSRC-v1 dataset with 10% to 30% labeled samples, CFMSC respectively achieves 2.31%, 2.74% and 1.26% average

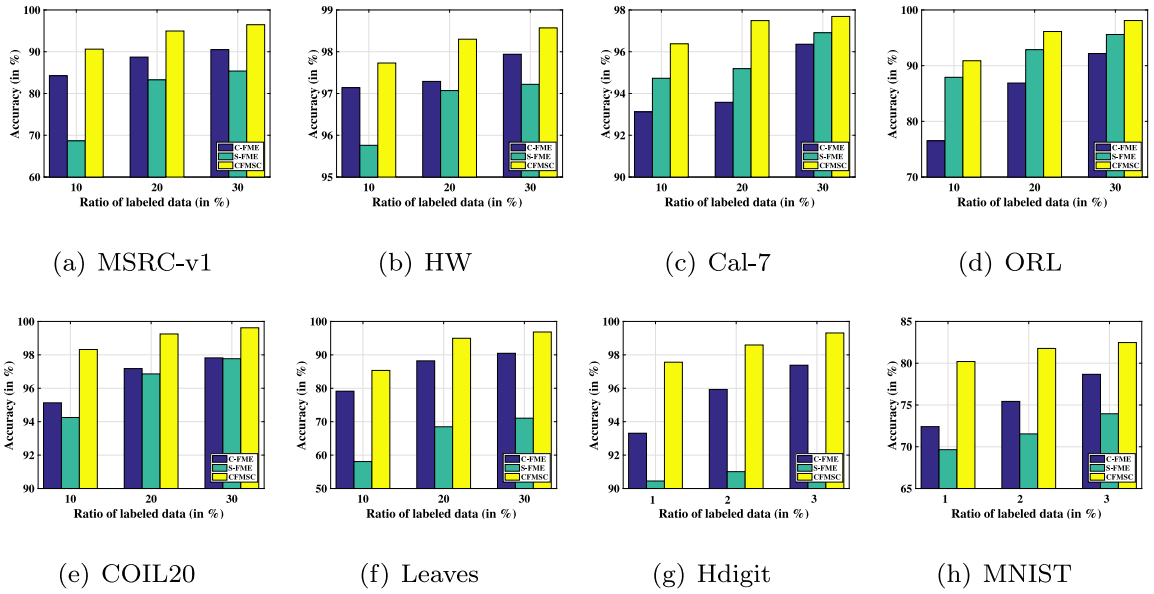


Fig. 4. Classification accuracy comparison of FME and CFMSC on unlabeled data.

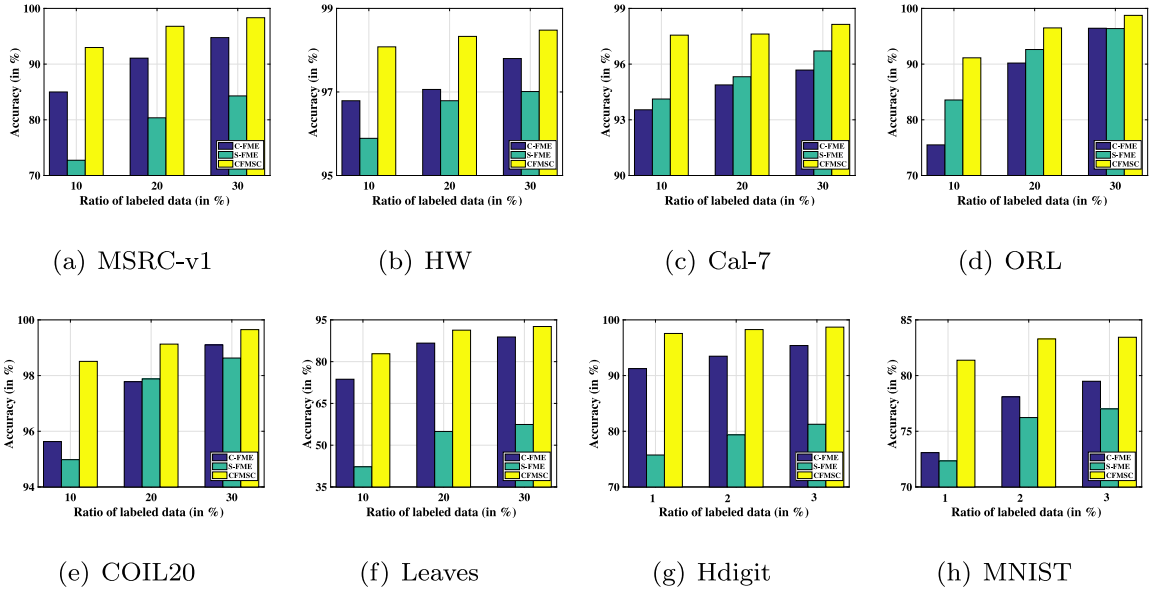


Fig. 5. Classification accuracy comparison of FME and CFMSC on testing data.

improvements for the unlabeled samples compared with the best results of graph-based fusion methods, and it respectively achieves 2.09%, 2.50%, 1.48% and 1.47% average improvements for the testing samples compared with the best results of projection-based fusion methods. Due to the limited memory space of the computer, some graph-based fusion methods encounter the out-of-memory error on the MNIST dataset, such as MLAN, FMSEL and SMGI. Although another graph fusion method (i.e., AMEMC) and two projection fusion methods (i.e., MVAR and JCD) can be run on MNIST, these methods have poor classification performance in dealing with this large-scale dataset. Moreover, on the Hdigit and MNIST datasets, CFMSC not only achieves comparable classification performance but also consumes less running time than the graph-based fusion method and the projection-based fusion methods except JCD, demonstrating that the anchor-based acceleration strategy greatly reduces the computational complexity of CFMSC and thereby make it scalable to relatively large-scale data. In summary, the superior classification accuracy and less time cost of CFMSC validate the effectiveness and superiority of the proposed collaborative fusion of both feature projections and similarity graphs.

Different from existing methods, CFMSC can simultaneously coalesce multiple graphs and projection subspaces in an adaptive-weighting manner. Benefiting from this fusion scheme, the view weights can be adaptively optimized. The adaptive weights learned on all views are reported in Table 6, from which we observe that different views contribute variously to CFMSC. For example, on the Cal-7 dataset, the sum of weights for view #1, view #2 and view #4 is greater than 0.95, which means that the rest three views have much less contribution for data fusion. The results in Table 6 demonstrate that CFMSC can effectively discriminate different views, even though there exist the low-quality views, such that the excellent views are emphasized while the low-quality views are weakened. It is this mechanism that enables CFMSC to make full use of the comprehensive information of multiple views to achieve a complete fusion of multi-view data, facilitating performance improvement.

Table 3
Classification results (ACC% ± STD×10^2) of different methods on unlabeled data with various ratios of labeled data. The best results are in bold, and those not significantly worse than the best are marked with * using the paired t-test at the 95% confidence level.

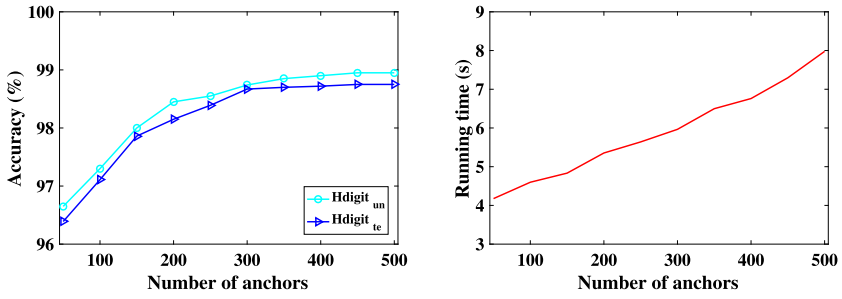
Datasets	Ratio	RAMC	MVAR	JCD	MLAN	FMSEL	SMGI	AMEMC	CFMSC
MSRC-v1	10%	68.67 ± 5.32	89.90 ± 2.85*	89.45 ± 3.22*	87.95 ± 2.97	85.55 ± 2.80	85.65 ± 3.10	88.31 ± 2.56	**90.62 ± 2.83**
	20%	79.85 ± 3.12	92.95 ± 2.16	92.59 ± 2.20	92.22 ± 2.34	90.19 ± 2.72	90.92 ± 2.69	90.15 ± 2.14	**94.96 ± 1.58**
	30%	82.31 ± 2.27	95.04 ± 2.47	94.58 ± 2.30	94.16 ± 1.80	95.21 ± 2.39	93.18 ± 2.39	91.51 ± 1.51	**96.47 ± 1.82**
HW	10%	94.73 ± 0.59	96.16 ± 0.35	96.90 ± 0.43	96.60 ± 0.91	96.99 ± 0.62	97.33 ± 0.64*	95.09 ± 1.08	**97.73 ± 0.38**
	20%	96.16 ± 0.32	96.93 ± 0.46	97.28 ± 0.52	97.81 ± 0.41	97.93 ± 0.48*	97.70 ± 0.56	97.27 ± 0.25	**98.30 ± 0.34**
	30%	96.96 ± 0.24	97.26 ± 0.34	97.48 ± 0.42	98.38 ± 0.29*	98.52 ± 0.25*	97.99 ± 0.33	97.44 ± 0.31	**98.57 ± 0.32**
Cal-7	10%	79.58 ± 3.25	95.73 ± 2.15*	95.52 ± 0.56	93.51 ± 0.98	**96.43 ± 0.74**	93.75 ± 0.88	94.01 ± 1.50	96.38 ± 1.02*
	20%	92.64 ± 2.05	97.23 ± 0.42*	95.58 ± 0.59	94.54 ± 0.66	97.13 ± 0.54*	95.66 ± 0.78	96.14 ± 0.53	**97.49 ± 0.52**
	30%	92.93 ± 1.57	97.43 ± 0.45*	97.34 ± 0.94	96.83 ± 0.68	97.32 ± 0.39*	96.66 ± 0.68	96.88 ± 0.37	**97.69 ± 0.49**
ORL	10%	69.39 ± 3.91	85.18 ± 2.10	81.11 ± 2.09	85.73 ± 2.17	88.68 ± 1.33	87.21 ± 1.88	88.82 ± 2.99*	**90.88 ± 1.68**
	20%	86.29 ± 3.39	93.13 ± 2.48	91.40 ± 1.97	95.29 ± 2.38	95.10 ± 2.24	95.67 ± 2.28*	94.94 ± 1.64	**96.13 ± 1.97**
	30%	92.20 ± 3.13	97.10 ± 1.71	95.63 ± 3.56	97.57 ± 1.90	98.00 ± 1.82*	97.30 ± 1.64	96.65 ± 1.60	**98.10 ± 1.46**
COIL20	10%	95.15 ± 1.18	97.63 ± 1.49	96.83 ± 1.13	98.05 ± 1.01*	97.33 ± 0.72	97.17 ± 1.28	94.16 ± 1.27	**98.32 ± 0.98**
	20%	96.63 ± 0.90	98.75 ± 0.71	98.53 ± 0.93	99.16 ± 0.94*	**99.35 ± 0.51**	98.95 ± 0.76*	97.01 ± 0.96	99.25 ± 0.55*
	30%	97.65 ± 0.68	99.30 ± 0.37	99.22 ± 0.38	99.51 ± 0.52*	99.61 ± 0.43*	99.04 ± 0.37	98.92 ± 0.55	**99.62 ± 0.39**
Leaves	10%	72.58 ± 2.11	77.67 ± 1.65	79.70 ± 1.79	83.25 ± 2.70	84.82 ± 1.52*	84.41 ± 2.19	80.38 ± 1.43	**85.34 ± 1.34**
	20%	81.55 ± 1.67	89.74 ± 1.55	90.22 ± 1.18	90.30 ± 1.87	93.82 ± 1.01	94.00 ± 0.89	91.28 ± 0.99	**94.96 ± 0.69**
	30%	85.34 ± 1.23	90.81 ± 1.33	92.27 ± 1.07	92.43 ± 1.29	93.86 ± 1.24	95.66 ± 0.78	92.99 ± 0.87	**96.84 ± 0.87**
Hdigit	1%	82.65 ± 0.76	92.77 ± 0.47	90.76 ± 0.99	**97.78 ± 0.73**	97.53 ± 0.71*	97.11 ± 0.92*	93.16 ± 1.24	97.56 ± 0.64*
	2%	89.28 ± 0.55	93.02 ± 0.42	92.46 ± 0.85	**98.90 ± 0.55**	98.81 ± 0.44*	98.68 ± 0.27*	95.33 ± 0.49	98.59 ± 0.43*
	3%	91.08 ± 0.39	93.61 ± 0.27	93.83 ± 0.56	**99.55 ± 0.24**	99.34 ± 0.18*	99.34 ± 0.12	97.34 ± 0.40	99.51 ± 0.28*
MNIST	1%	67.31 ± 1.89	78.43 ± 1.30	73.43 ± 1.24	OM	OM	OM	77.19 ± 1.72	**80.20 ± 1.12**
	2%	70.47 ± 1.35	80.09 ± 0.50	79.06 ± 1.11	OM	OM	OM	79.01 ± 1.20	**81.77 ± 0.72**
	3%	70.93 ± 1.28	80.54 ± 0.40	80.71 ± 0.70	OM	OM	OM	80.45 ± 0.91	**82.47 ± 0.58**

Table 4
Classification results (ACC% ± STD×10^2) of different methods on testing data with various ratios of labeled data. The best results are in bold, and those not significantly worse than the best are marked with * using the paired t-test at the 95% confidence level.

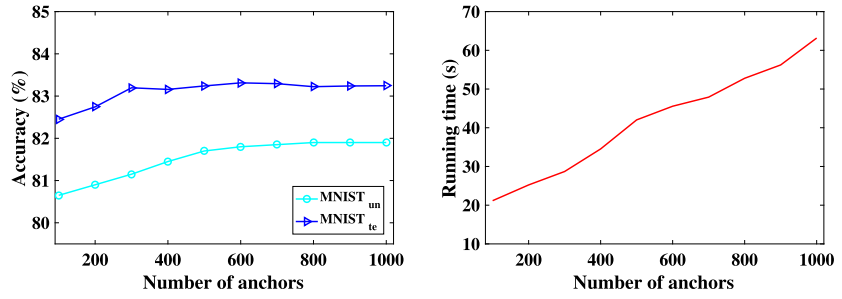
Datasets	Ratio	RAMC	MVAR	JCD	FMSEL	CFMSC
MSRC-v1	10%	68.81 ± 6.37	90.48 ± 2.79	88.69 ± 4.43	89.36 ± 4.63	**92.98 ± 4.47**
	20%	80.12 ± 6.00	94.17 ± 3.04	95.31 ± 2.38*	95.07 ± 3.56*	**96.79 ± 2.59**
	30%	81.79 ± 6.05	95.29 ± 3.03	96.86 ± 1.53	97.71 ± 1.75*	**98.33 ± 1.56**
HW	10%	94.31 ± 1.53	96.13 ± 1.19	96.49 ± 1.27	97.07 ± 0.78	**98.08 ± 0.69**
	20%	95.20 ± 0.88	96.61 ± 0.90	96.86 ± 1.02	97.52 ± 0.66	**98.33 ± 0.61**
	30%	96.75 ± 0.89	96.80 ± 0.72	97.23 ± 0.98	98.36 ± 0.46*	**98.48 ± 0.60**
Cal-7	10%	79.92 ± 4.16	96.81 ± 1.75	94.76 ± 1.72	96.63 ± 1.67	**97.56 ± 1.06**
	20%	92.17 ± 2.49	97.08 ± 1.17*	95.22 ± 1.51	96.14 ± 1.11	**97.62 ± 1.12**
	30%	92.73 ± 1.92	97.22 ± 1.01	97.27 ± 1.21	97.71 ± 1.15*	**98.14 ± 1.05**
ORL	10%	67.00 ± 5.60	85.94 ± 4.80	85.81 ± 5.40	87.13 ± 3.91	**91.13 ± 4.07**
	20%	86.12 ± 5.56	94.75 ± 3.21	92.00 ± 3.13	95.06 ± 3.18	**96.50 ± 2.21**
	30%	93.94 ± 3.98	98.00 ± 1.59*	97.44 ± 1.79	98.38 ± 1.47*	**98.75 ± 1.57**
COIL20	10%	94.86 ± 1.65	95.76 ± 1.68	96.51 ± 0.95	97.36 ± 1.01	**98.51 ± 1.09**
	20%	96.63 ± 1.30	98.99 ± 0.49	98.66 ± 0.49	99.27 ± 0.48*	**99.13 ± 0.52**
	30%	97.65 ± 0.48	99.14 ± 0.35*	99.27 ± 0.41*	99.55 ± 0.39*	**99.65 ± 0.46**
Leaves	10%	71.09 ± 2.11	77.59 ± 2.66	79.31 ± 1.89	79.66 ± 2.56	**82.84 ± 2.69**
	20%	80.66 ± 1.54	88.94 ± 2.03	90.03 ± 1.80	87.72 ± 1.60	**91.31 ± 1.28**
	30%	85.13 ± 1.41	91.19 ± 1.37	92.31 ± 1.38*	90.51 ± 1.12	**92.62 ± 1.09**
Hdigit	1%	83.69 ± 0.76	91.57 ± 0.89	90.00 ± 1.95	95.07 ± 1.31	**97.65 ± 0.36**
	2%	90.39 ± 0.61	91.97 ± 0.75	92.67 ± 0.71	95.74 ± 0.55	**98.26 ± 0.19**
	3%	91.18 ± 0.55	92.31 ± 0.64	94.16 ± 0.62	96.90 ± 0.53	**98.70 ± 0.18**
MNIST	1%	67.68 ± 1.86	80.13 ± 0.41	73.82 ± 1.18	OM	**81.38 ± 0.72**
	2%	70.79 ± 2.20	81.00 ± 0.25	80.18 ± 0.75	OM	**83.29 ± 0.21**
	3%	71.13 ± 1.22	81.47 ± 0.23	81.44 ± 0.56	OM	**83.44 ± 0.16**

Table 5
The average running time (seconds) of the multi-view semi-supervised methods on all datasets with 20% or 2% labeled data. OM denotes the out-of-memory error.

Dataset	MVAR	JCD	MLAN	FMSEL	SMGI	AMEMC	CFMSC
MSRC-v1	0.24	0.17	0.23	0.40	0.15	0.12	0.20
HW	4.10	0.48	10.85	20.51	2.36	1.65	4.77
Cal-7	6.02	2.15	12.47	16.59	5.29	1.16	6.47
ORL	0.40	0.61	0.62	1.02	0.13	0.14	0.28
COIL20	2.22	2.16	3.92	5.39	2.59	0.72	1.70
Leaves	1.51	0.51	4.54	5.31	2.97	0.69	1.04
Hdigit	29.26	15.02	393.63	85.66	64.50	36.17	5.68
MNIST	86.65	38.57	OM	OM	OM	205.23	41.93



(a) Accuracy versus the number of anchors (b) Time versus the number of anchors



(c) Accuracy versus the number of anchors (d) Time versus the number of anchors

Fig. 6. The classification accuracy and running time versus the different number of anchors, in which (a) and (b) show the results on Hdigit, and (c) and (d) show the results on MNIST.

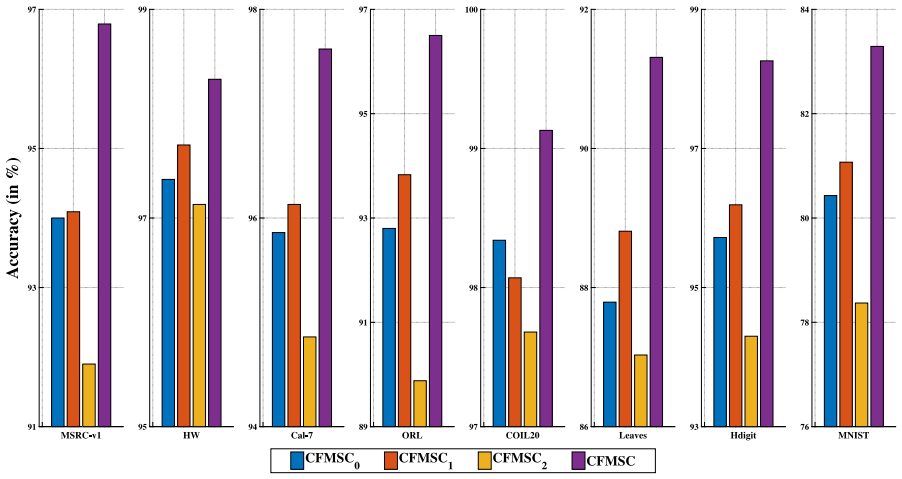
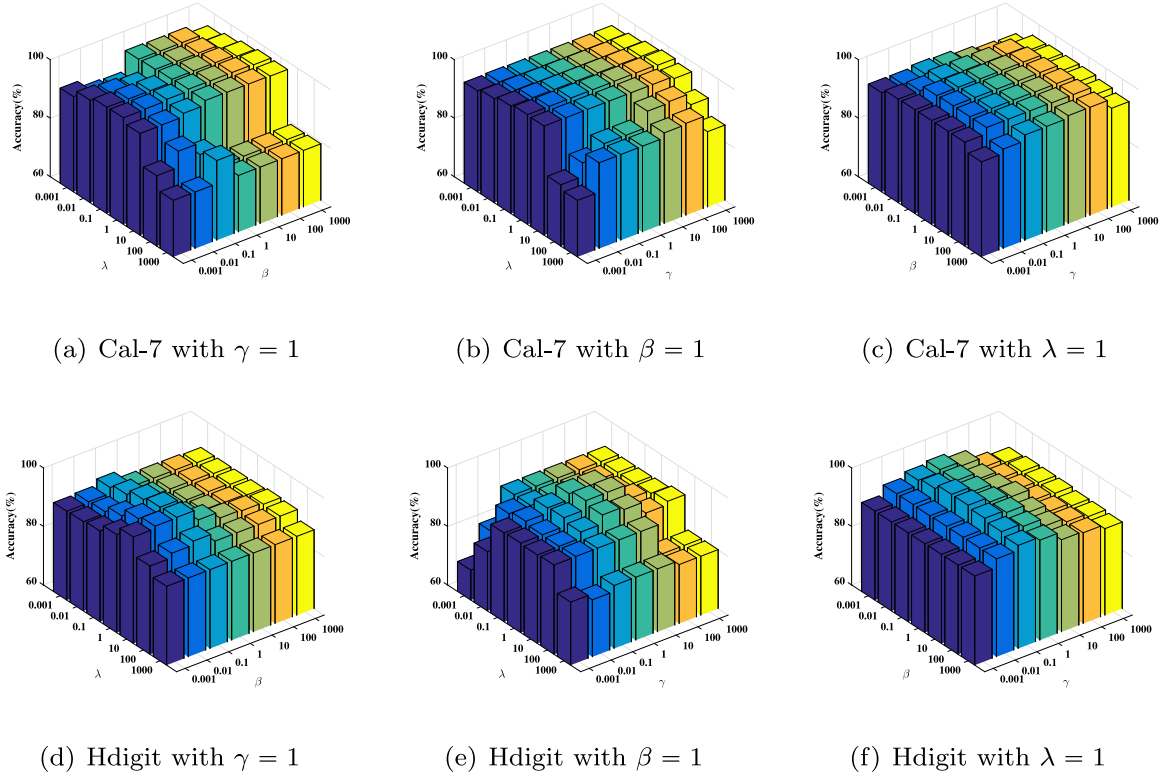


Fig. 7. Classification results of CFMSC, $CFMSC_0$, $CFMSC_1$ and $CFMSC_2$ on testing data.

Table 6
The learned average weight for each view using the proposed method.

View	MSRC-v1	HW	Cal-7	ORL	COIL20	Leaves	Hdigit	MNIST
#1	0.187	0.355	0.378	0.238	0.415	0.390	0.525	0.394
#2	0.067	0.051	0.338	0.449	0.005	0.316	0.475	0.323
#3	0.201	0.315	0.035	0.238	0.454	0.294	–	0.283
#4	0.067	0.026	0.235	0.075	0.126	–	–	–
#5	0.194	0.248	0.009	–	–	–	–	–
#6	0.284	0.005	0.005	–	–	–	–	–

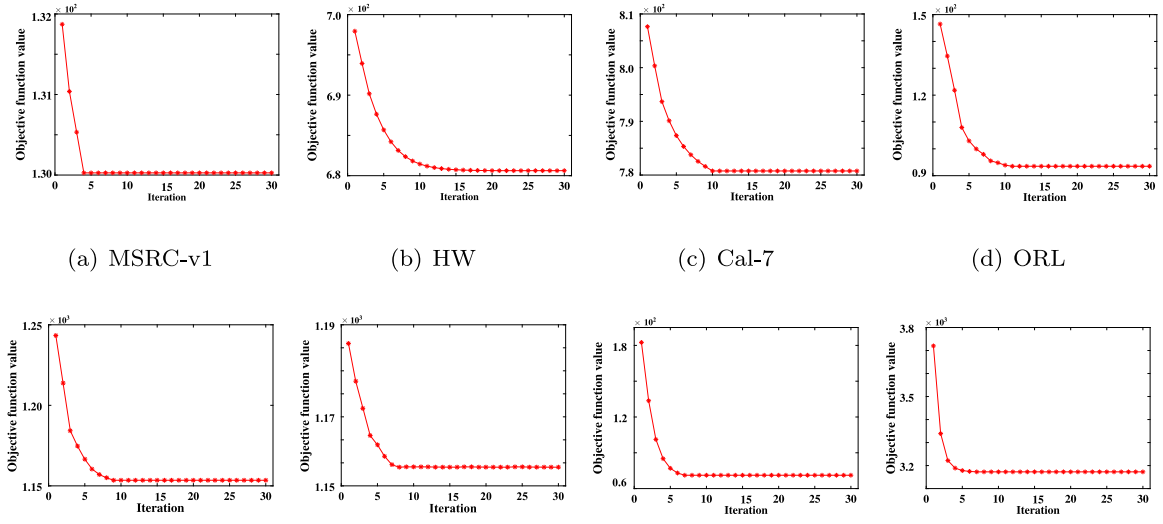
Effect of Anchor Points' Number. From Section 3.3, we note that the number of anchor points (i.e., m) has a direct effect on the computation complexity of the acceleration strategy designed for CFMSC. To quantitatively analyze the effect of anchors on the performance of CFMSC, we conduct experiments with different numbers of anchors. On the Hdigit and MNIST datasets, the numbers of anchors vary from the ranges of {50,100, ⋯,500} and {100,200, ⋯,1000}, respectively. For each number of anchors, CFMSC is independently run 20 times with 2% labeled samples, and the parameters are set as same as that in Table 3. The classification accuracy on unlabeled data and testing data as well as the running time are shown in Fig. 6. From the results, we observe that there exist different variation trends in the accuracy and



(a) Cal-7 with $\gamma = 1$ (b) Cal-7 with $\beta = 1$ (c) Cal-7 with $\lambda = 1$

(d) Hdigit with $\gamma = 1$ (e) Hdigit with $\beta = 1$ (f) Hdigit with $\lambda = 1$

Fig. 8. The accuracy variations of CFMSC with parameters λ, β and γ on Cal-7 and Hdigit.



(a) MSRC-v1 (b) HW (c) Cal-7 (d) ORL

(e) COIL20 (f) Leaves (g) Hdigit (h) MNIST

Fig. 9. Variation curves of objective function values.

time as the number of anchors increases. Specifically, the classification accuracy of CFMSC grows smoothly and then reaches a stable value as depicted in Figs. 6(a) and (c). However, the running time significantly increases along with the number of anchors as depicted in Figs. 6 (b) and (d), which demonstrates that it is not suitable for CFMSC to use too many anchor points. Consequently, to maintain comparable accuracy and acceptable running time, we need to generate a proper number of anchors in which the number of anchors should be much smaller than the data size (i.e., $m \ll n$), and set a threshold for the number of anchors on different datasets. Taking both the difference in data sizes and the balance of classification ability and time cost into consideration, how to determine the optimal number of anchor points on each dataset is worthy of further research.

Ablation study. To further analyze the influence of the adaptive collaborative fusion of both feature projections and similarity graphs, we conduct an ablation study in this subsection. Specifically, we first remove the feature projection fusion part from CFMSC, and thus get a simplified multi-view model (named CFMSC_0) that treats each projection subspace equally. CFMSC_0 performs the multi-view fusion from the aspect of similarity graphs. Accordingly, we can remove the adaptive graph fusion part from CFMSC, thereby getting another multi-view variant of CFMSC (named CFMSC_1). CFMSC_1 uses a fixed graph

(i.e., $\mathbf{S} = \sum_{v=1}^{V} \mathbf{S}^v/V$) and merely considers the feature projection-level fusion. Finally, we entirely remove the collaborative fusion part and thus get a single-view variant of CFMSC (named CFMSC_2). The classification results on testing data with 20% or 2% of labeled samples are depicted in Fig. 7. From the results, we observe that CFMSC is consistently superior to its two multi-view variants (i.e., CFMSC_0 and CFMSC_1) which merely consider the multi-view data fusion in the level of graph or projection. This indicates that the adaptive collaborative fusion of feature projections and similarity graphs is indeed helpful to boost the performance of multi-view classification. Benefiting from this simultaneous fusion manner, CFMSC can obtain more informative knowledge from multi-view data, not only improving the reliability of label propagation but also making CFMSC learn a more discriminative projection subspace. Meanwhile, the multi-view CFMSC_0 and CFMSC_1 also outperforms the single-view variant of CFMSC (i.e., CFMSC_2) which sets the view weights $\alpha_v (v = 1, \ldots, V)$ to be $1/V$, demonstrating that it is inappropriate to treat each view equally and simply concatenate multiple views (i.e., projections and graphs) for classification. This further verifies that our proposed fusion manner can discriminate different views and simultaneously assign appropriate weights to them such that the effects of low-quality views can be largely weakened. As a result, it is effective and necessary to adaptively integrate both feature projections and similarity graphs within a unified framework.

4.3. Parameter sensitivity and convergence analysis

In the proposed CFMSC, there are three parameters λ, β, and γ that need to be tuned, in which λ adjusts the over-fitting problem of the feature projection fusion, β controls the smoothness of label propagation, and γ balances the importance of the graph fusion and learning. To investigate the influence of these parameters on performance, we conduct experiments on the Cal-7 and Hdigit datasets. Specifically, we change two parameters in the range of $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ and fix the other parameter as 1. The classification results on testing data with 20% or 2% of labeled samples are depicted in Fig. 8. It can be seen that CFMSC is not very sensitive to β, and is somewhat sensitive to λ and γ. CFMSC achieves relatively good results with varying λ, β, and γ in the range of $\{10^{-1}, 10^{0}, 10^{1}\}$, whereas CFMSC performs worse when λ and γ are too small or large (e.g., 10^{-3}), demonstrating that the terms corresponding to λ and γ are particularly important for CFMSC. Considering that the graph fusion and learning guarantees that the fused graph can accurately propagate label information, and the feature projection fusion facilitates learning a discriminative joint feature projection, λ and γ should be appropriately set to balance the importance of their corresponding terms in the proposed CFMSC. According to the above analysis, we can first use a fixed β and then tune λ and γ by the grid search to achieve better performance.

The objective function of CFMSC is iteratively solved in Algorithm 1. In this part, we experimentally illustrate the convergence of CFMSC on all multi-view datasets. Here, the ratio of labeled samples is also set to 20% or 2%, and parameters γ, λ, and β are set to 1. Fig. 9 shows the variations of the objective function with the number of iterations. We find that the objective function monotonically decreases at each iteration and quickly converges within 15 iterations, demonstrating that the adopted optimization solution is effective.

5. Conclusions

In this paper, we present an adaptive collaborative fusion method for semi-supervised multi-view classification problem. Different from existing methods that perform the multi-view fusion from the level of graphs or projections independently, our formulation can adaptively discriminate different views and fuse multiple graphs as well as projections within a unified framework simultaneously, obtaining more informative knowledge from multi-view data. Benefiting from this collaborative fusion scheme, the proposed CFMSC can learn a joint feature projection and a unified similarity graph compatible across different views, in which the correlation and diversity among views are fully considered and effectively utilized. Moreover, CFMSC coalesces different views in an adaptive-weighting manner without extra weight-related parameters. Inspired by the anchor-based bipartite graph, an acceleration strategy is designed to reduce the computational complexity of CFMSC by using a small size of anchor points. Experimental results on different datasets have validated the superiority of CFMSC compared to the state-of-the-art competitors over the classification performance and time cost.

Although CFMSC achieves its objectives, some interesting directions are worthwhile in future research. First, we will extend CFMSC to be a more general multi-view learning framework that can work in unsupervised scenarios. Second, it is possible to use kernel tricks to further enhance performance. Additionally, how to extend CFMSC to effectively tackle the task that selects a minimal or optimal view subset from multiple views will be studied in the future.

CRediT authorship contribution statement

Bingbing Jiang: Writing – original draft, Methodology, Software, Formal analysis, Investigation, Data curation, Visualization. **Chenglong Zhang:** Investigation, Writing – editing. **Yan Zhong:** Writing – review & editing, Validation. **Yi Liu:** Conceptualization. **Yingwei Zhang:** Writing – review. **Xingyu Wu:** Writing – review, Validation. **Weiguo Sheng:** Project administration, Funding acquisition.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

This work is supported in part by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang Province, China (Grant No. 2023C01022), the National Natural Science Foundation of China (Grant No. 62006065), National Key Research and Development Program of China (Grant No. 2022YFC3601200), and the Zhejiang Provincial Natural Science Foundation of China (Grant No. LY22F030004).

Appendix A. Proof of Eq. (13)

Proof. Since $\sum_{i=1}^{n} q_i = n$, thus the first term of Eq. (13) can be rewritten by the Cauchy inequality:

$$\sum_{i=1}^{n} \frac{1}{q_i} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2^2 \geq \frac{1}{n} \Big(\sum_{i=1}^{n} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2 \Big)^2 . \tag{35}$$

According to Eq. (35), we have:

$$\frac{1}{n} \Big(\sum_{i=1}^{n} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2 \Big)^2 = \min_{q_i \geq 0, \mathbf{1}^T \boldsymbol{q} = n} \sum_{i=1}^{n} \frac{1}{q_i} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2^2 . \tag{36}$$

Therefore, we can further infer that:

$$\min_{\boldsymbol{W}_v, \boldsymbol{\alpha}} \frac{1}{n} \Big(\sum_{i=1}^{n} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2 \Big)^2 \Leftrightarrow \min_{\boldsymbol{W}_v, \boldsymbol{\alpha}} \frac{1}{n} \sum_{i=1}^{n} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2$$

$$\Leftrightarrow \min_{\boldsymbol{W}_v, \boldsymbol{\alpha}, q_i \geq 0, \mathbf{1}^T \boldsymbol{q} = n} \sum_{i=1}^{n} \frac{1}{q_i} \left\| \mathbf{f}_i - \sum_{v=1}^{V} \alpha_v \boldsymbol{W}_v^T \boldsymbol{x}_i^v \right\|_2^2 . \tag{37}$$

Hence we prove that Eq. (12) can be equivalent to Eq. (13).

Appendix B. Proof of Eq. (16)

Proof. We first denote the objective function in Eq. (16) as $g(\widetilde{\boldsymbol{W}}, \boldsymbol{F})$, then the constant term $\mathrm{Tr}(\boldsymbol{Y}^T \boldsymbol{U}_n \boldsymbol{Y})$ can be removed from $g(\widetilde{\boldsymbol{W}}, \boldsymbol{F})$. Thus $g(\widetilde{\boldsymbol{W}}, \boldsymbol{F})$ can be rewritten in matrix form as:

$$g(\widetilde{\boldsymbol{W}}, \boldsymbol{F}) = \mathrm{Tr}\left(\begin{bmatrix}\boldsymbol{F}\\ \widetilde{\boldsymbol{W}}\end{bmatrix}^T \boldsymbol{Z} \begin{bmatrix}\boldsymbol{F}\\ \widetilde{\boldsymbol{W}}\end{bmatrix}\right) - \mathrm{Tr}\left(\begin{bmatrix}\boldsymbol{F}\\ \widetilde{\boldsymbol{W}}\end{bmatrix}^T \begin{bmatrix}2\boldsymbol{U}_n \boldsymbol{Y}\\ \boldsymbol{0}\end{bmatrix}\right), \tag{38}$$

where

$$\boldsymbol{Z} = \begin{bmatrix}\boldsymbol{Q} + \beta \boldsymbol{L}_S + \boldsymbol{U}_n & -\boldsymbol{Q}\boldsymbol{X}^T\\ -\boldsymbol{X}\boldsymbol{Q} & \boldsymbol{X}\boldsymbol{Q}\boldsymbol{X}^T + \lambda \boldsymbol{A}^{-1}\end{bmatrix} \in \mathbb{R}^{(n+d)\times(n+d)}. \tag{39}$$

To prove that $g(\widetilde{\boldsymbol{W}}, \boldsymbol{F})$ is jointly convex w.r.t. $\widetilde{\boldsymbol{W}}$ and $\boldsymbol{F}$, we only need to prove that the matrix $\boldsymbol{Z} \in \mathbb{R}^{(n+d)\times(n+d)}$ is positive semi-definite. For an arbitrary vector $\boldsymbol{v} = [\boldsymbol{v}_1^T, \boldsymbol{v}_2^T]^T \in \mathbb{R}^{(n+d)\times 1}$, where $\boldsymbol{v}_1 \in \mathbb{R}^{n\times 1}$ and $\boldsymbol{v}_2 \in \mathbb{R}^{d\times 1}$, we have

$$\boldsymbol{v}^T \boldsymbol{Z} \boldsymbol{v} = \boldsymbol{v}_1^T (\beta \boldsymbol{L}_S + \boldsymbol{U}_n)\boldsymbol{v}_1 + \lambda \boldsymbol{v}_2^T \boldsymbol{A}^{-1} \boldsymbol{v}_2 + (\boldsymbol{v}_1 - \boldsymbol{X}^T \boldsymbol{v}_2)^T \boldsymbol{Q} (\boldsymbol{v}_1 - \boldsymbol{X}^T \boldsymbol{v}_2). \tag{40}$$

Since $\boldsymbol{U}_n$, $\boldsymbol{A}$ and $\boldsymbol{Q}$ are all the nonnegative diagonal matrices, and the Laplacian matrix $\boldsymbol{L}_S$ is positive semi-definite [40], we have $\boldsymbol{v}^T \boldsymbol{Z} \boldsymbol{v} \geq 0$, proving that $\boldsymbol{Z}$ is positive semi-definite. Therefore, Eq. (16) is jointly convex w.r.t. $\widetilde{\boldsymbol{W}}$ and $\boldsymbol{F}$.

Appendix C. Optimization of Eq. (23)

According to [41], Eq. (23) can be solved by tackling its counterpart:

$$\min_{\boldsymbol{\alpha}\geq 0, \boldsymbol{\alpha}^T \mathbf{1}=1, \boldsymbol{z}} \boldsymbol{\alpha}^T \boldsymbol{M} \boldsymbol{z} - \boldsymbol{\alpha}^T \boldsymbol{h} + \frac{\mu}{2} \left\| \boldsymbol{\alpha} - \boldsymbol{z} + \frac{\boldsymbol{\tau}}{\mu} \right\|_2^2, \tag{41}$$

where $\boldsymbol{z} \in \mathbb{R}^{V\times 1}$ denotes a slack variable, $\mu > 0$ is a penalty parameter, and $\boldsymbol{\tau} \in \mathbb{R}^{V\times 1}$ is a Lagrangian multiplier. Eq. (41) can be iteratively optimized by the augmented Lagrangian multiplier method. Specifically, μ is gradually increased with step δ ($1 < \delta < 2$) during each iteration, making $\boldsymbol{\alpha}$ and $\boldsymbol{z}$ closer, then $\boldsymbol{\tau}$ is updated by $\boldsymbol{\tau} \leftarrow \boldsymbol{\tau} + \mu(\boldsymbol{\alpha} - \boldsymbol{z})$. When the updated $\boldsymbol{\alpha}$ and $\boldsymbol{z}$ are sufficiently close to each other, Eq. (41) converges, and finally the optimal $\boldsymbol{\alpha}$ can be obtained. The solution steps are as follows:

Step 1. Update $\boldsymbol{z}$: When $\boldsymbol{\alpha}$ is fixed, Eq. (41) is an unconstrained optimization problem. By setting the derivative of Eq. (41) w.r.t. $\boldsymbol{z}$ to zero, we update $\boldsymbol{z}$ by:

$$\boldsymbol{z} = \boldsymbol{\alpha} - \frac{1}{\mu}\left(\boldsymbol{M}\boldsymbol{\alpha} - \boldsymbol{\tau}\right). \tag{42}$$

Step 2. Update $\boldsymbol{\alpha}$: When $\boldsymbol{z}$ is fixed with its current value of $\boldsymbol{z}$ (i.e., $\boldsymbol{z}^*$), $\boldsymbol{\alpha}$ can be updated by minimizing the following problem:

$$\min_{\boldsymbol{\alpha}^T \mathbf{1}=1, \boldsymbol{\alpha}\geq 0} \left\| \boldsymbol{\alpha} - \boldsymbol{z}^* + \frac{1}{\mu}(\boldsymbol{\tau} + \boldsymbol{M}\boldsymbol{z}^* - \boldsymbol{h}) \right\|_2^2, \tag{43}$$

which can be solved with a closed-form solution [35]. In this way, $\boldsymbol{\alpha}$ is adaptively updated according to the aforementioned steps.

References

[1] Shiliang Sun, A survey of multi-view machine learning, Neural Comput. Appl. 23 (7–8) (2013) 2031–2038.

[2] Shiliang Sun, John Shawe-Taylor, Liang Mao, PAC-Bayes analysis of multi-view learning, Inf. Fusion 35 (2017) 117–131.

[3] Shiliang Sun, Mengran Yu, John Shawe-Taylor, Liang Mao, Stability-based PAC-Bayes analysis for multi-view learning algorithms, Inf. Fusion 86–87 (2022) 76–92.

[4] Jinxing Li, Bob Zhang, Guangming Lu, David Zhang, Generative multi-view and multi-feature learning for classification, Inf. Fusion 45 (2019) 215–226.

[5] Shiliang Sun, Daoming Zong, LCBM: A multi-view probabilistic model for multi-label classification, IEEE Trans. Pattern Anal. Mach. Intell. 43 (8) (2021) 2682–2696.

[6] Jing Zhao, Xijiong Xie, Xin Xu, Shiliang Sun, Multi-view learning overview: Recent progress and new challenges, Inf. Fusion 38 (2017) 43–54.

[7] Lai Tian, Feiping Nie, Xuelong Li, A unified weight learning paradigm for multi-view learning, in: International Conference on Artificial Intelligence and Statistics, 2019, pp. 2790–2800.

[8] Zhongheng Li, Qianyao Qiang, Bin Zhang, Fei Wang, Feiping Nie, Flexible multi-view semi-supervised learning with unified graph, Neural Netw. 142 (2021) 92–104.

[9] Guoqing Chao, Shiliang Sun, Semi-supervised multi-view maximum entropy discrimination with expectation Laplacian regularization, Inf. Fusion 45 (2019) 296–306.

[10] Shiliang Sun, Wenbo Dong, Qiuyang Liu, Multi-view representation learning with deep Gaussian processes, IEEE Trans. Pattern Anal. Mach. Intell. 43 (12) (2021) 4453–4468.

[11] Sally El Hajjar, Fadi Dornaika, Fahed Abdallah, Multi-view spectral clustering via constrained nonnegative embedding, Inf. Fusion 78 (2022) 209–217.

[12] Xijiong Xie, Yanfeng Li, Shiliang Sun, Deep multi-view multiclass twin support vector machines, Inf. Fusion 91 (2023) 80–92.

[13] Saeedeh Bahrami, Fadi Dornaika, Alireza Bosaghzadeh, Joint auto-weighted graph fusion and scalable semi-supervised learning, Inf. Fusion 66 (2021) 213–228.

[14] Bingbing Jiang, Junhao Xiang, Xingyu Wu, Yadi Wang, Huanhuan Chen, Weiwei Cao, Weiguo Sheng, Robust multi-view learning via adaptive regression, Inform. Sci. 610 (2022) 916–937.

[15] Bingbing Jiang, Xingyu Wu, Xiren Zhou, Anthony G Cohn, Yi Liu, Weiguo Sheng, Huanhuan Chen, Semi-supervised multi-view feature selection with adaptive graph learning, IEEE Trans. Neural Netw. Learn. Syst. early access (2022) 1–15.

[16] Bin Zhang, Qianyao Qiang, Fei Wang, Feiping Nie, Fast multi-view semi-supervised learning with learned graph, IEEE Trans. Knowl. Data Eng. 34 (1) (2022) 286–299.

[17] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, Bernhard Schölkopf, Learning with local and global consistency, in: Advances in Neural Information Processing Systems, 2004, pp. 321–328.

[18] Feiping Nie, Dong Xu, Ivor Wai Hung Tsang, Changshui Zhang, Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction, IEEE Trans. Image Process. 19 (7) (2010) 1921–1932.

[19] Vikas Sindhwani, Partha Niyogi, Mikhail Belkin, A co-regularization approach to semi-supervised learning with multiple views, in: Proceedings of ICML Workshop on Learning with Multiple Views, Vol. 2005, 2005, pp. 74–79.

[20] Shiliang Sun, Multi-view Laplacian support vector machines, in: International Conference on Advanced Data Mining and Applications, Springer, 2011, pp. 209–222.

[21] Xijiong Xie, Shiliang Sun, Multi-view Laplacian twin support vector machines, Appl. Intell. 41 (4) (2014) 1059–1068.

[22] Xijiong Xie, Shiliang Sun, General multi-view semi-supervised least squares support vector machines with multi-manifold regularization, Inf. Fusion 62 (2020) 63–72.

[23] Masayuki Karasuyama, Hiroshi Mamitsuka, Multiple graph label propagation by sparse integration, IEEE Trans. Neural Netw. Learn. Syst. 24 (12) (2013) 1999–2012.

[24] Xiao Cai, Feiping Nie, Weidong Cai, Heng Huang, Heterogeneous image features integration via multi-modal semi-supervised learning model, in: IEEE International Conference on Computer Vision, 2013, pp. 1737–1744.

[25] Shiping Wang, Zhewen Wang, Wenzhong Guo, Accelerated manifold embedding for multi-view semi-supervised classification, Inform. Sci. 562 (2021) 438–451.

[26] Feiping Nie, Guohao Cai, Jing Li, Xuelong Li, Auto-weighted multi-view learning for image clustering and semi-supervised classification, IEEE Trans. Image Process. 27 (3) (2018) 1501–1511.

[27] Feiping Nie, Lai Tian, Rong Wang, Xuelong Li, Multiview semi-supervised learning model for image classification, IEEE Trans. Knowl. Data Eng. 32 (12) (2020) 2389–2400.

[28] Feiping Nie, Guohao Cai, Xuelong Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2017, pp. 2408–2414.

[29] Najmeh Ziraki, Fadi Dornaika, Alireza Bosaghzadeh, Multiple-view flexible semi-supervised classification through consistent graph construction and label propagation, Neural Netw. 146 (2022) 174–180.

[30] Hong Tao, Chenping Hou, Feiping Nie, Jubo Zhu, Dongyun Yi, Scalable multi-view semi-supervised classification via adaptive regression, IEEE Trans. Image Process. 26 (9) (2017) 4283–4296.

[31] Wenzhang Zhuge, Chenping Hou, Shaoliang Peng, Dongyun Yi, Joint consensus and diversity for multi-view semi-supervised classification, Mach. Learn. 109 (2020) 445–465.

[32] Bingbing Jiang, Xingyu Wu, Kui Yu, Huanhuan Chen, Joint semi-supervised feature selection and classification through Bayesian approach, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2019, pp. 3983–3990.

[33] Xiaojun Chang, Feiping Nie, Yi Yang, Heng Huang, A convex formulation for semi-supervised multi-label feature selection, in: Proceedings of the 28th AAAI Conference on Artificial Intelligence, 2014, pp. 1171–1177.

[34] Xiao Dong, Lei Zhu, Xuemeng Song, Jingjing Li, Zhiyong Cheng, Adaptive collaborative similarity learning for unsupervised multi-view feature selection, in: International Joint Conference on Artificial Intelligence, 2018, pp. 2064–2070.

[35] Jin Huang, Feiping Nie, Heng Huang, A new simplex sparse learning model to measure data similarity for clustering, in: International Joint Conference on Artificial Intelligence, 2015, pp. 3569–3575.
[36] Xuelong Li, Han Zhang, Rong Wang, Feiping Nie, Multi-view clustering: A scalable and parameter-free bipartite graph fusion method, IEEE Trans. Pattern Anal. Mach. Intell. 44 (1) (2022) 330–344.
[37] Yeqing Li, Feiping Nie, Heng Huang, Junzhou Huang, Large-scale multi-view spectral clustering via bipartite graph, in: Proceedings of AAAI Conference on Artificial Intelligence, 2015, pp. 2750–2756.
[38] Hao Wang, Yan Yang, Bing Liu, GMC: Graph-based multi-view clustering, IEEE Trans. Knowl. Data Eng. 32 (6) (2020) 1116–1129.
[39] Bingbing Jiang, Junhao Xiang, Xingyu Wu, Wenda He, Libin Hong, Weiguo Sheng, Robust adaptive-weighting multi-view classification, in: Proceedings of the ACM International Conference on Information and Knowledge Management, 2021, pp. 3117–3121.
[40] Fan R.K. Chung, Spectral graph theory, in: Proc. CBMS Regional Conf. Series in Math., vol. 92, 1997.
[41] Dimitri P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, 2014.