



# Multi-view Fusion and View-specific-labels Learning for Multi-label Classification

Zhongliu Yang<sup>1</sup> · Jun Yin<sup>1</sup>

Accepted: 28 June 2025 / Published online: 24 October 2025  
© The Author(s) 2025

## Abstract

In multi-view multi-label learning, each sample is characterized by multiple views and simultaneously labeled with multiple semantic labels. Previous approaches typically combined information from various views to obtain a shared subspace, assuming each view has a shared set of labels. However, these methods ignore the diversity of multiple views, and each view can only capture a fraction of the complete label space. Moreover, the current approaches in this field often rely on either global label structures or local smoothness for learning label correlations, disregarding scenarios where both global and local correlations are crucial. To tackle these issues, we present a new method to improve multi-label classification through Multi-view Fusion and View-specific-labels Learning (MFVL). More concretely, we first get the shared subspace to capture shared information across the various views. Next, we combine it with the original multi-view dataset to construct a new multi-view dataset. To enhance multi-label classification performance, we learn view-specific labels and simultaneously consider global and local label correlations. Finally, our model is extended to nonlinear version to effectively handle linearly inseparable problem. In numerous experiments conducted on nine multi-view multi-label datasets, the proposed method achieves superior performance compared to state-of-the-art algorithms.

**Keywords** Multi-label classification · View-specific-labels learning · Multi-view fusion · Label correlation

## 1 Introduction

With the arrival of the information age, a huge amount of data is generated every day. These data are not only obtained from diverse perspectives but can also be represented in multiple forms. For instance, a news report may be presented in the form of text, images, and videos. Similarly, an image can be described using some feature types like GIST, RGB, SIFT, and HSV. This type of data is called multi-view data. In contrast to single-view data, multi-view

---

✉ Jun Yin  
junyin@shmtu.edu.cn

Zhongliu Yang  
yang\_zhongliu@163.com

<sup>1</sup> College of Information Engineering, Shanghai Maritime University, Shanghai 201306, Shanghai, People's Republic of China

data has richer feature information and can solve different data problems more effectively. Recently, multi-view learning [1–3] has gained significant attention in research and is now widely utilized in tasks such as classification [4] and clustering [5].

Furthermore, a sample is no longer restricted to a single semantic label, but can be annotated with multiple labels. For instance, a movie can be simultaneously tagged as horror, suspense, and drama, while a landscape photo can be labeled with different categories like sky, snow-capped mountains, and forests [6]. Compared to traditional single-label learning, multi-label learning provides a more accurate representation of the multiple semantic information contained in an object. Over the past few decades, research on multi-label learning has quickly gained extensive attention. Some mature methods [7–9] have been presented and widely used in various fields like image annotation [10], protein categorization [11], gene annotation [12], and text classification [13].

As an integration of the two types of learning, multi-view multi-label learning presents a more effective solution to tackle the intricacies of real-world data classification problems. In this learning paradigm, each target has several views and is assigned several labels. Currently, there are three key difficulties in this field:

1. Multi-view learning involves consistency and complementarity. Consistency means learning consistent information from multiple views, while complementarity indicates the diverse information learned from different views. To address multi-view multi-label learning problem, it is important to exploit the consistency and complementarity of multi-view data comprehensively.
2. In Multi-label Learning, different semantic labels are interrelated and exhibit strong correlations. Therefore, exploiting label correlation can effectively improve multi-label classification.
3. Most real-world data is not linearly separable. To ensure the robustness of the proposed approach, it is essential to handle nonlinear data more effectively.

To address the first challenge, current methods can be roughly classified into two categories. The first category utilizes matrix factorization techniques to extract shared subspace information from multi-view data. Subsequently, discriminative prediction classifiers are trained by considering the varying contributions of different views to the labels [14–18]. Nevertheless, this approach primarily emphasizes extracting shared information from multiple views while disregarding their individual discrepancies. In addition, while subspace learning techniques for multi-view multi-label approaches can effectively handle high-dimensional problems, they ignore the inherent architectural information in the original feature space. The second category involves training separate classifier for each view and assigning them different contribution weights, followed by integrating them into a unified prediction classifier [19–21]. Nonetheless, this method does not take into account consistency across multiple views. As for the second challenge, exploiting label correlation has been proven to be an effective approach [22–27]. Considering kernel functions can map the original data into higher-dimensional spaces, For the third challenge, existing research methods address the problem of linear inseparability by using kernel functions to extend linear models to nonlinear ones [20, 28, 29].

Based on the aforementioned analysis and discussion, this paper presents an innovative framework aimed at addressing the three challenges mentioned in learning with multiple views and multiple labels. Fig. 1 depicts the model framework of the proposed approach. Specifically, we extract common low-dimensional subspace from multi-view multi-label dataset. The subspace is then fused as a complementary view with the original dataset. By applying label correlation information over a shared label dataset along with the view

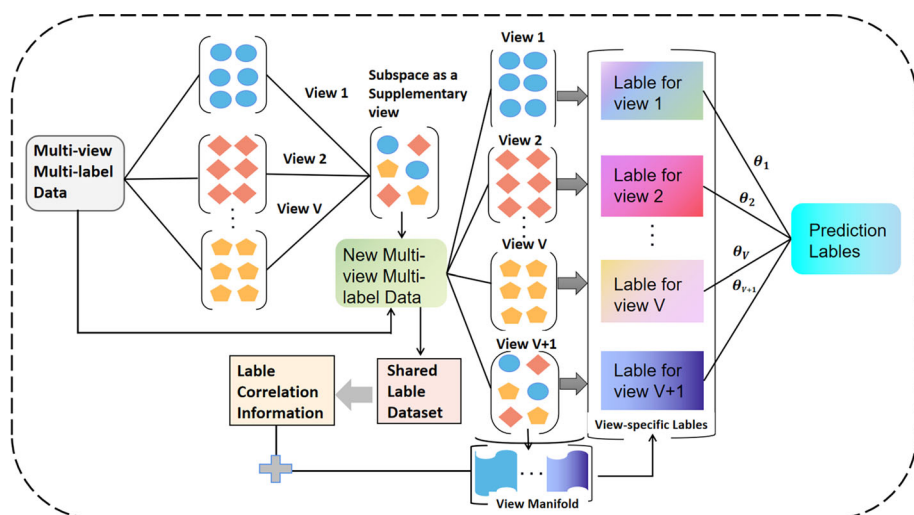


**Fig. 1** The framework of the proposed MFVL method

manifold, we obtain view-specific labels. We carefully consider the contribution weights of each view-specific label and integrate the predictions into the multi-label classification. To address the linear indivisibility problem, we developed the model as a non-linear one. Our optimization problem is addressed by an alternating iteration method. Through extensive experiments, we demonstrate the superiority of our approach over some relevant methods. The key contributions of this paper can be summarized as follows:

1. We propose an innovative multi-view multi-label learning approach that addresses two major challenges in the field by integrating multiple views, processing view-specific labels, and capturing label correlations.
2. We implement a nonlinear framework in the development of our model to deal with linear inseparability, which not only solves the third challenge but also improves the robustness and flexibility of the model.
3. We use the alternating iteration approach to solve the optimization problem of the proposed method. By employing a two-step optimization strategy, we first obtain the complementary subspace view, followed by optimizing each variable in a multi-view multi-label framework.

This paper is arranged in the following: Section II provides a short review of existing research in the area of multi-view multi-label learning. Section III details our proposed method, while Section IV utilizes an alternating iteration approach to solve it. Section V presents comprehensive experimental results and analyses. In Section VI, we summarize the paper and suggest potential directions for future research.

## 2 Related work

In this section, we will primarily introduce the latest research related to our proposed approach, which can be further categorized into three parts: multi-view multi-label learning, label correlation learning, and kernel extension learning.

## 2.1 Multi-view multi-label learning

In this learning area, each instance has many views and is associated with many semantic labels simultaneously. Currently, there are two main research methods: One is solving the problem from a multi-view perspective and the other is solving the problem from a multi-label perspective. In the first category, the primary focus is processing the multiple views and subsequently performing multi-label classification. This category can be further divided into three subcategories: feature concatenation, early fusion, and late fusion.

Feature concatenation refers to transforming the multi-view multi-label problem into a single-view multi-label problem by combining the features from all views into a single view. For example, Ren et al. [30] merged all views into a hybrid feature matrix and expanded the original label matrix using a correlation matrix to improve the accuracy of the labels. However, this direct concatenation method not only leads to the curse of dimensionality but also overlooks the potential differences between views.

Early fusion involves learning a shared subspace from multiple views through matrix factorization. As an example, Liu et al. [14] presented an approach that incorporates low-rank matrix constraints to achieve a low-dimensional common representation and investigated the complementarity of different views by allocating individual weight to each view. Subsequently, they embedded the low-dimensional shared representation into the matrix completion framework for multi-label classification. Zhang et al. [16] utilized matrix factorization techniques to obtain a latent subspace that effectively captures the complementarity among various views. By preserving a coherent latent semantic basis in the kernel space, they were able to improve the consistency of shared representations across multiple views. Tan et al. [18] employed non-negative matrix factorization to learn a discriminative low-rank common subspace from incomplete views and solved the problem of missing label by considering label correlation. Liu et al. [17] learned a shared subspace across heterogeneous incomplete views and utilized the shared subspace and disambiguated labels to learn the prediction model. Tan et al. [15] first learned a shared subspace that captures collaborative patterns across different views. They then explored the features of individual views using multiple separate classifiers followed by an ensemble strategy for prediction. The methods mentioned have demonstrated promising results in effectively exploring both the individual characteristics and shared patterns in multi-view multi-label data. Nevertheless, when the number of views increases, obtaining shared semantic information between multiple views using subspace-based methods becomes more challenging.

Late fusion methods typically involve constructing individual classifiers for each view, and subsequently combining the projected results for all views with corresponding view contribution weights. For example, Zhang et al. [19] constructed a support vector machine classifier for each view and addressed the multi-view multi-label learning problem with a uniform learning pattern. Huang et al. [21] first learned label-specific representations for each view and constructed a model using label correlation and view consensus. They then obtained the final prediction by aggregating the predictions from all classifiers using the contribution weights. Zhao et al. [28] employed a unified feed-forward neural network model to build a separate classifier for each view to achieve a consistent label prediction. They integrated the Hilbert Schmidt criterion of independence with an exploration of the differences between perspectives. Eventually, they obtained a classification prediction model by combining label relevance and the contribution of views. The drawback of these methods is that they only consider the differences between multiple views without considering their consistency.

In the second category, they typically make the assumption that the label sets for each view are uniform and aligned. However, there are privacy protection issues in the real world.

For instance, in social communities, users may register several accounts on Twitter, Weibo, and WeChat, but linking these social accounts to the same user is a challenge for privacy reasons. Therefore, each view can only capture partial information in the label space, leading to the phenomenon of view non-aligned. To address this problem, multiple techniques have been presented for handling view non-aligned. Houthuys and Suykens [31] were the first to introduce the concept of non-aligned multi-views. They also proposed a method to enhance learning efficiency by incorporating must-link and cannot-link constraints into kernel spectral clustering. Li et al. [32] tackled the challenge of non-aligned views by assuming that, despite lacking explicit alignment, the views can still be indirectly linked through the presence of shared labels. Zhao et al. [33] highlighted that each view may not learn a complete set of labels. Therefore, the authors learned specific labels for each view and maximized the dependency between view-specific labels and their corresponding features. Zhao et al. [28] also utilized view-specific labels to learn information about agreement and variety between views and labels. From these methods, we can realize the fact that each view can only learn a subset of labels in the real-world scenario.

## 2.2 Label correlation learning

In multi-label learning, an object is related to more than one semantic label. But at an earlier stage of research, it was assumed that each label was mutually exclusive and independent of the others. Therefore a first-order strategy [22] was used to study label correlation. Later, researchers found that there are certain relationships between labels. Intuitively, labels with strong correlations tend to have similar classifier outputs, which led to the creation of a second-order strategy [23] to address label correlation. Indeed, label correlation exists not only between label pairs, but also has more complicated and higher dimensional connections. Therefore, to address more complex scenarios problems and enhance the accuracy of multi-label classification, higher-order label correlation are used to reconstruct and enrich the original label information [24, 25].

In previous research, most methods considered either the global structure of label correlations or only the local smoothness of labels. For instance, Xu et al. [34] introduced a novel approach for tackling missing labels, where they integrated a complementary label matrix with a low-rank structure to capture and leverage global label correlations. Huang et al. [35] highlighted that label correlations may only be applicable to a specific subset of data. However, multiple research studies have provided evidence that global and local label correlations can coexist, and taking into account both factors together can produce favorable results. For instance, Zhu et al. [26] addressed the problem considering both complete and missing labels by optimizing label manifolds through learning latent label representations and utilizing both global and local label correlations. Ren et al. [30] integrated data from all views into a hybrid feature matrix, learned global label correlations using kernel norms, and introduced manifold regularization to enhance local smoothness. Che et al. [27] assigned labels to corresponding label groups while simultaneously computing local and global label correlations.

## 2.3 Kernel extension learning

In machine learning, traditional linear classifiers such as support vector machines often fail to separate different classes in linearly inseparable datasets. Kernel extension addresses this issue by transforming the data from a lower-dimensional space to a higher-dimensional

space to achieve linear separability. This mapping relationship can be represented by kernel functions. In recent researches, scholars have incorporated nonlinear mappings into models. For instance, Zhao et al. [20, 28] presented a number of effective studies and highlighted the importance of nonlinear techniques in addressing these challenges. Zhu et al. [29] applied the technique of maximizing correlation to train their model, and further extended it to a nonlinear model.

### 3 The proposed method

In this section, we first provide a conceptual explanation of the proposed approach, followed by a comprehensive description of our model.

#### 3.1 Problem definition and notation

We define a collection of  $n$  instances and  $V$  views as  $\mathcal{X} = \{X_v\}_{v=1}^V$ . Here,  $X_v = [x_v^1, x_v^2, \dots, x_v^n]^T \in R^{n \times d_v}$  is the data matrix of the  $v$ th view.  $Y = [y_1, y_2, \dots, y_n]^T \in \{0, 1\}^{n \times q}$  represents the label matrix, while  $q$  refers to the total number of class labels. The value of  $y_{ij} = 1$  ( $j = 1, \dots, q$ ) represents the association between the  $j$ -th label and the  $i$ -th sample, otherwise it is irrelevant. By utilizing the partially labeled matrix  $Y$  and the feature matrix  $\mathcal{X}$ , our proposed method aims to accurately predict the label matrix for unlabeled samples.

#### 3.2 Problem formulation

Through the analysis discussed above, subspace learning methods have been shown to be effective in obtaining valuable multi-view information. Among these methods, Non-negative Matrix Factorization (NMF) stands out due to its unique properties. Therefore, in this paper, we leverage NMF to derive a shared and distinct latent subspace. Specifically, given a multi-view dataset  $\mathcal{X}$ , we apply standard NMF to discover a shared subspace denoted by  $B$ , as presented below:

$$\min_{B, Q_v} \sum_{v=1}^V \|X_v - BQ_v^T\|_F^2 \quad \text{s.t. } B \geq 0, Q_v \geq 0, \quad (1)$$

where  $B \in R^{n \times d}$  represents the shared low-dimensional subspace that captures the shared complementary information across all views, and  $Q_v \in R^{d_v \times d}$  represents the basic mapping matrix corresponding to the  $v$ th view. Here,  $d_v$  refers to the size of the  $v$ th view, while  $d$  represents the dimensionality of the lower-dimensional subspace. In this paper, we set  $d = d_{\min}$ , where  $d_{\min}$  is the smallest dimensionality among the multiple views.  $\|\cdot\|_F$  refers to the Frobenius norm, and the constraints  $Q_v \geq 0$  and  $B \geq 0$  impose non-negativity on the matrices.

The initial prediction model is a standard regression problem represented by the least squares loss function defined below:

$$\begin{aligned} \min_{W_v, \theta_v} L &= \frac{1}{2} \sum_{v=1}^V \left\{ \theta_v \|X_v W_v - Y\|_F^2 + \lambda_1 \|W_v\|_F^2 \right\} + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } \theta_v &\geq 0, \sum_{v=1}^V \theta_v = 1, \end{aligned} \quad (2)$$

In equation (2), where  $\theta = [\theta_1, \theta_2, \dots, \theta_V]$ ,  $W_v \in R^{d_v \times q}$  represents the confidence matrix,  $\theta_v$  represents the weight contribution of the  $v$ th view, and  $L$  denotes the objective function. Here, we impose non-negativity and normalization constraints on  $\theta$  to ensure that each view has a certain contribution to the predicted labels and to obtain a consistent label prediction.

Next, we use the common low-dimensional subspace  $B$  obtained from equation (1) as a complementary view, forming a new multi-view dataset  $\widehat{X}_v$  with the original multi-view dataset  $X_v$ , i.e.  $\widehat{X}_v = \{X_1, X_2, \dots, X_V, B\}$ . The updated multi-view dataset now contains  $\widehat{V} = V + 1$  views. After this modification, the preceding equations can be restated in the following manner:

$$\begin{aligned} \min_{W_v, \theta_v} L &= \frac{1}{2} \sum_{v=1}^{\widehat{V}} \left\{ \theta_v \|\widehat{X}_v W_v - Y\|_F^2 + \lambda_1 \|W_v\|_F^2 \right\} + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } \theta_v &\geq 0, \sum_{v=1}^{\widehat{V}} \theta_v = 1, \end{aligned} \quad (3)$$

In equation (3), we assume that all samples in all views share a common label set, implying that all views are aligned. However, as mentioned earlier, this assumption is an idealized scenario and may not be applicable in real-world situations. In practical scenarios, we can only observe a partial set of labels for each view, rather than the complete label space. Therefore, we introduce a view-specific label matrix  $U_v$  to address the issue of inconsistent observable information among different views. Similarly, in order to maintain the consistency of the semantic information between the feature space and different views, we introduce the graph Laplace regularization. The representation is as follows:

$$\begin{aligned} \min_{U_v, W_v, \theta_v} L &= \frac{1}{2} \sum_{v=1}^{\widehat{V}} \left\{ \theta_v \left( \|\widehat{X}_v W_v - U_v\|_F^2 + \lambda_3 \text{tr} \left( U_v^\top L_v U_v \right) \right) + \lambda_1 \|W_v\|_F^2 \right. \\ &\quad \left. + \lambda_4 \|U_v - Y\|_F^2 \right\} + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } \theta_v &\geq 0, \sum_{v=1}^{\widehat{V}} \theta_v = 1, \end{aligned} \quad (4)$$

In equation (4),  $L_v$  represents the graph Laplacian matrix, which satisfies  $L_v = D_v - S_v$ . Here, the matrix  $S_{ij}$  denotes the similarity matrix between sample  $X_i$  and sample  $X_j$ . Additionally, the diagonal matrix  $D_v$  is formed with elements  $D_{i,i}^v$  as  $D_{i,i}^v = \sum_{j=1}^n S_{i,j}^v$ . To capture the local geometric structure in the feature space, we construct a graph based on the nearest neighbor technique. The computation of similarity between two samples in the  $v$ th view is as follows:

$$S_{i,j}^v = \begin{cases} e^{-\frac{\|x_i^v - x_j^v\|^2}{2\sigma^2}}, & \text{if } x_j^v \in N_k(x_i^v) \text{ or } x_i^v \in N_k(x_j^v), \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where the term  $N_k()$  represents the  $k$  nearest neighbors.

Utilizing label propagation dependency and considering global label correlations and local smoothness assumptions concurrently can effectively enhance multi-label classification performance. Hence, in our work, we first learn the global label correlations. This is realized by obtaining a complementary label matrix  $YC$  through the use of a label correlation matrix  $C$  to complement the initial label matrix  $Y$ . To ensure reliable label correlations, we introduce a reconstruction loss term that constrains the original and supplemental label matrices. The modified objective function is as follows:

$$\begin{aligned} \min_{U_v, W_v, \theta_v, C} L = & \frac{1}{2} \sum_{v=1}^{\widehat{V}} \left\{ \theta_v \left( \|\widehat{X}_v W_v - U_v\|_F^2 + \lambda_3 \operatorname{tr} \left( U_v^\top L_v U_v \right) \right) + \lambda_1 \|W_v\|_F^2 \right. \\ & \left. + \lambda_4 \|U_v - Y\|_F^2 \right\} + \frac{\lambda_5}{2} \|YC - Y\|_F^2 + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } & \theta_v \geq 0, \sum_{v=1}^{\widehat{V}} \theta_v = 1, \end{aligned} \quad (6)$$

To ensure the robustness of our model, it is essential to take into account label correlations at both the global and local levels. Local label correlations can be explored using the smoothness assumption, which states that the distance between any two samples in the feature space reflects the similarity of their corresponding class labels. Additionally, if two original label vectors  $y_i$  and  $y_j$  are strongly correlated in the geometric space, then there is also a strong correlation between their corresponding supplemented true label vectors. Thus, the representation is as follows:

$$\begin{aligned} \Omega(C) &= \sum_{i,j=1}^n \|YC_i - YC_j\|^2 S_{i,j} \\ &= \operatorname{Tr} \left( (YC)^\top L_v YC \right). \end{aligned} \quad (7)$$

In equation (7), the label similarity matrix  $C$  preserves the local smoothness of the samples. Here,  $S_{ij}$  is the sample similarity matrix, which can be calculated using equation (5), and  $L_v$  is the corresponding graph Laplacian matrix. Considering the previous discussions and analyses, we can express the final objective function in the following manner:

$$\begin{aligned} \min_{U_v, W_v, \theta_v, C} L = & \frac{1}{2} \sum_{v=1}^{\widehat{V}} \left\{ \theta_v \left( \|\widehat{X}_v W_v - U_v\|_F^2 + \lambda_3 \operatorname{tr} \left( U_v^\top L_v U_v \right) \right) + \lambda_1 \|W_v\|_F^2 \right. \\ & \left. + \lambda_4 \|U_v - Y\|_F^2 + \lambda_6 \operatorname{tr} \left( (YC)^\top L_v (YC) \right) \right\} \\ & + \frac{\lambda_5}{2} \|YC - Y\|_F^2 + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } & \theta_v \geq 0, \sum_{v=1}^{\widehat{V}} \theta_v = 1. \end{aligned} \quad (8)$$

To address the linear indivisibility problem of the given data, we extend our approach to nonlinear models for improved performance. We introduce a mapping function  $\phi(\cdot)$  that maps features from the original space to a higher-dimensional Hilbert space. By leveraging the representation theorem [23] and assuming the existence of a kernel matrix  $K$  satisfies

$K = \phi(\hat{x})\phi(\hat{x})^T$ , with  $\kappa(\cdot, \cdot)$  denoting the kernel function, we can express the initial linear variable  $W$  as  $W = \phi(\hat{x})^T A$ . Thus, Equation (8) is accordingly revised:

$$\begin{aligned} \min_{U_v, A_v, \theta_v, C} L = & \frac{1}{2} \sum_{v=1}^{\hat{V}} \left\{ \theta_v \left( \|K_v A_v - U_v\|_F^2 + \lambda_3 \operatorname{tr} \left( U_v^T L_v U_v \right) \right) + \lambda_1 \operatorname{tr} \left( A_v^T K_v A_v \right) \right. \\ & + \lambda_4 \|U_v - Y\|_F^2 + \lambda_6 \operatorname{tr} \left( (YC)^T L_v (YC) \right) \Big\} \\ & + \frac{\lambda_5}{2} \|YC - Y\|_F^2 + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } & \theta_v \geq 0, \sum_{v=1}^{\hat{V}} \theta_v = 1. \end{aligned} \quad (9)$$

In summary, we can obtain an innovative multi-view multi-label learning model MFVL, which can be optimized by minimizing a loss function  $L$  containing multiple parts, each as follows:

- Reconstruction error:  $\|K_v A_v - U_v\|_F^2$  denotes the reconstruction error of the feature matrix  $\hat{X}_v$  in view  $v$ , where  $K_v$  is the kernel matrix, and  $A_v$  and  $U_v$  are the parameters to be learned.
- Graph regularization terms:  $\lambda_3 \operatorname{tr}(U_v^T L_v U_v)$ ,  $\lambda_1 \operatorname{tr}(A_v^T L_v A_v)$  and  $\lambda_6 \operatorname{tr}((YC)^T L_v (YC))$  use graph structure information to enhance the generalization of the model.
- Label consistency:  $\lambda_4 \|U_v - Y\|_F^2$  ensures consistency between predicted labels and true labels.
- Label correlation:  $\frac{\lambda_5}{2} \|YC - Y\|_F^2$  reinforces the correlation between labels.
- Weight regularization:  $\frac{\lambda_2}{2} \|\theta\|_2^2$  regularizes the weight vector  $\theta$  to avoid overfitting.

By optimizing the above loss function, the model learns the weights of each view  $\theta_v$  and other parameters  $A_v$ ,  $U_v$  and  $C$ . Finally, it is then possible to predict the label by fusing the features of multiple views with the learned weights and parameters. The fused features are  $F_p = \sum_{v=1}^{\hat{V}} \theta_v (K_v^{test}) A_v$ , where  $K_v^{test}$  is the kernel matrix of the test data. A threshold  $\delta$  is set as 0.5 by cross-validation. With  $\delta$ , the fused features  $F_p$  are converted to the predictive label matrix  $Y_p$  as follows:

$$Y_p = \operatorname{sign}(F_p - \delta). \quad (10)$$

where the  $\operatorname{sign}(\cdot)$  function maps the value of each element to 0 or 1, indicating the presence or absence of a label. The sign function  $\operatorname{sign}(\cdot)$  satisfies the following conditions:

$$\operatorname{sign}(p) = \begin{cases} 1, & \text{if } p > 0 \\ 0, & \text{otherwise} \end{cases}. \quad (11)$$

### 3.3 Optimization

In this article, we first obtain a low-dimensional subspace as an additional complementary view using the original multi-view data. Then we use the newly acquired dataset and label matrix to facilitate multi-label learning across multiple views. Therefore, the optimization contains two steps.

### 3.3.1 The first step in optimization

Firstly, we use non-negative matrix factorization to obtain a low-dimensional subspace  $B$ . The objective function in equation (1) is a convex optimization problem involving two variables,  $B$  and  $Q_v$ . We adopt an alternating iterative update method to optimize the variables. Specifically, during each iteration, we optimize one variable while fixing the others. The iterative update optimization process is outlined as follows:

- (1) Update  $B$  while fixing  $\{Q_v\}_{v=1}^V$  The partial derivative of  $L$  with respect to  $B$  as:

$$\nabla B = 2B \sum_{v=1}^V Q_v^\top Q_v - 2 \sum_{v=1}^V X_v Q_v. \quad (12)$$

By setting  $\nabla B = 0$ , the closed-form solution for  $B$  can be derived, and it can be expressed as:

$$B = \left( \sum_{v=1}^V X_v Q_v \right) \left( \sum_{v=1}^V Q_v^\top Q_v \right)^{-1}. \quad (13)$$

- (2) Update  $\{Q_v\}_{v=1}^V$  while fixing  $B$  For each view  $v$ , we can obtain the gradient of the objective function with respect to  $Q_v$ , which can be formulated as:

$$\nabla Q_v = 2Q_v B^\top B - 2X_v^\top B. \quad (14)$$

By setting  $\nabla Q_v = 0$ , we can deduce the solution for  $Q_v$  using a closed-form expression as:

$$Q_v = X_v^\top B \left( B^\top B \right)^{-1}. \quad (15)$$

---

**Algorithm 1** The first step of the learning process of MFVL
 

---

**Require:** Training dataset:  $\mathcal{X} = \{X_v\}_{v=1}^V \in R^{n \times dv}$

**Ensure:** Updating Variables:  $B, \{Q_v\}_{v=1}^V$

```

1: Randomly initialize  $B$  as  $rand(n, d)$ 
2: while not converged do
3:   for  $v = 1$  to  $V$  do
4:     Update variable  $Q_v$  according to equation (15)
5:   end for
6:   Update variable  $B$  according to equation (13)
7: end while
```

---

### 3.3.2 The second step in optimization

After obtaining the low-dimensional subspace  $B$ , we obtain a new multi-view dataset  $\widehat{X}_v$ . By analysis, the objective function in equation (9) involves four variables:  $A_v$ ,  $U_v$ ,  $\theta$  and  $C$ , making it a non-convex problem overall. To address this issue, we utilize an alternative optimization approach in the second step of the optimization process. The specific update process is summarized as follows:

(1) Update  $A_v$  while fixing  $U_v$ ,  $\theta$  and  $C$  We find the equation of  $L$  with respect to  $A_v$  as:

$$L(A_v) = \frac{1}{2} \sum_{v=1}^{\hat{V}} \left\{ \theta_v \|K_v A_v - U_v\|_F^2 + \lambda_1 \text{tr} \left( A_v^T K_v A_v \right) \right\}. \quad (16)$$

By differentiating  $A_v$  in  $L(A_v)$ , we get the following derivatives:

$$\nabla A_v = \theta_v K_v^T (K_v A_v - U_v) + \frac{\lambda_1}{2} K_v A_v + \frac{\lambda_1}{2} K_v^T A_v. \quad (17)$$

By setting  $\nabla A_v = 0$ , we can obtain the following equation,

$$K_v^T (\theta_v K_v A_v - \theta_v U_v + \lambda_1 A_v) = 0. \quad (18)$$

In this article, further simplification of the above equation leads to a closed-form solution for  $A_v$ , expressed as follows:

$$A_v = (\theta_v K_v + \lambda_1 I)^{-1} (\theta_v U_v). \quad (19)$$

(2) Update  $U_v$  while fixing  $A_v$ ,  $\theta$  and  $C$  We find the equation of  $L$  with respect to  $U_v$  as:

$$L(U_v) = \frac{1}{2} \sum_{v=1}^{\hat{V}} \left\{ \theta_v \left( \|K_v A_v - U_v\|_F^2 + \lambda_3 \text{tr} \left( U_v^T L_v U_v \right) \right) + \lambda_4 \|U_v - YC\|_F^2 \right\}. \quad (20)$$

We can denote the derivative of  $L$  with respect to  $U_v$  as:

$$\nabla U_v = \frac{1}{2} \theta_v \lambda_3 L_v^T U_v + \frac{1}{2} \theta_v \lambda_3 L_v U_v + \lambda_4 (U_v - YC) - \theta_v (K_v A_v - U_v). \quad (21)$$

In this article, as it is stated that,  $L_v^T = L_v$ , the aforementioned equation can be reformulated as follows:

$$\nabla U_v = \lambda_3 \theta_v L_v U_v + \lambda_4 (U_v - YC) + \theta_v (U_v - K_v A_v). \quad (22)$$

By setting  $\nabla U_v = 0$ , we can obtain the closed-form solution for  $U_v$ , which is expressed as follows:

$$U_v = (\lambda_3 \theta_v L_v + (\theta_v + \lambda_4) I)^{-1} (\lambda_4 YC + \theta_v K_v A_v). \quad (23)$$

(3) Update  $C$  while fixing  $U_v$ ,  $\theta$  and  $A_v$

We find the equation of  $L$  with respect to  $C$  as:

$$L(C) = \frac{1}{2} \sum_{v=1}^{\hat{V}} \left\{ \lambda_4 \|U_v - YC\|_F^2 + \lambda_6 \text{tr}((YC)^T L_v (YC)) \right\} + \frac{\lambda_5}{2} \|YC - Y\|_2^2. \quad (24)$$

By differentiating  $L(C)$  with respect to  $C$ , we can obtain the derivative as:

$$\nabla C = \sum_{v=1}^{\hat{V}} \left\{ \lambda_6 Y^T L_v YC + \lambda_4 Y^T (YC - U_v) \right\} + \lambda_5 Y^T (YC - Y). \quad (25)$$

By setting  $\nabla C = 0$ , the closed-form solution for  $C$  can be derived, and it can be represented as:

$$C = \left( \sum_{v=1}^{\widehat{V}} \lambda_6 Y^T L_v Y + \lambda_4 Y^T Y + \lambda_5 Y^T Y \right)^{-1} \left( \sum_{v=1}^{\widehat{V}} \lambda_4 Y^T U_v + \lambda_5 Y^T Y \right). \quad (26)$$

(4) Update  $\theta$  while fixing  $U_v$ ,  $A_v$  and  $C$

We find the objective function with respect to  $\theta_v$ , which can be formulated as follows:

$$\begin{aligned} L(\theta_v) &= \frac{1}{2} \sum_{v=1}^{\widehat{V}} \left\{ \theta_v \left( \|K_v A_v - U_v\|_F^2 + \lambda_3 \operatorname{tr}(U_v^T L_v U_v) \right) \right\} \\ &\quad + \frac{\lambda_2}{2} \|\theta\|_2^2 \\ \text{s.t. } \theta_v &\geq 0, \sum_{v=1}^{\widehat{V}} \theta_v = 1. \end{aligned} \quad (27)$$

Considering the constraints on the parameter  $\theta_v$  in the objective function, with the total sum of  $\theta_v$  constrained to be 1, according to [20], we utilize the Lagrangian function method to update  $\theta_v$ . We set  $H_v = \frac{1}{2} (\|K_v A_v - U_v\|_F^2 + \lambda_3 \operatorname{tr}(U_v^T L_v U_v))$ . Then the specific representation is as follows:

$$\theta_v = \begin{cases} \frac{\sum_{v=1}^{\widehat{V}} H_v + \lambda_2 - \widehat{V} H_v}{\widehat{V} \lambda_2} & \text{if } \varphi > 0 \\ \epsilon & \text{if } \varphi \leq 0 \end{cases}. \quad (28)$$

where  $\varphi = \sum_{v=1}^{\widehat{V}} H_v + \lambda_2 - \widehat{V} H_v$ , and  $\epsilon$  is a very small non-negative number.

---

### Algorithm 2 The second step of the learning process of MFVL

---

**Require:** The training dataset:  $\mathcal{X} = \{\widehat{X}_v\}_{v=1}^{\widehat{V}} \in R^{n \times dv}$ ; The label matrix:  $Y \in R^{n \times q}$ ; The parameters:  $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6$ ;

**Ensure:** The predictive labeling matrix for unlabeled samples:  $Y_p$

```

1: Randomly initialize  $U_v = \operatorname{rand}(n, q)$  and  $C = \operatorname{zeros}(q)$ ;
2: Calculate the Laplacian matrix  $L_v$  according to equation (5);
3: Let  $i = 0$ ;
4: while not converged do
5:   for  $v = 1$  to  $\widehat{V}$  do
6:      $i = i + 1$ ;
7:     Update variable  $A_v$  according to equation (19);
8:     Update variable  $U_v$  according to equation (23);
9:     Update variable  $C$  according to equation (26);
10:    Update variable  $\theta$  according to equation (28);
11:   end for
12: end while
13: Calculate the predicted label matrix  $Y_p$  according to equation (10).
```

---

### 3.4 Complexity analysis

In our method, the time complexity of MFVL is mainly influenced by multiple variables, i.e., the update of  $B$ ,  $Q_v$ ,  $A_v$ ,  $U_v$  and  $C$ . In the first optimization, the computation complexity

**Table 1** The summary of datasets.

| Datasets  | Views | Dimension in each view      | Samples | Labels | Domain  |
|-----------|-------|-----------------------------|---------|--------|---------|
| Yeast     | 2     | 79/24                       | 2417    | 14     | biology |
| Emotions  | 2     | 8/64                        | 593     | 6      | music   |
| Pascal07  | 4     | 100/512/1000/4096           | 9963    | 20     | image   |
| Mirflickr | 4     | 100/512/1000/4096           | 4000    | 38     | image   |
| Scene     | 6     | 49/49/49/49/49/49           | 2407    | 6      | image   |
| Espgame3k | 6     | 100/512/1000/4096/4096/4096 | 3000    | 268    | image   |
| Birds     | 4     | 65/65/65/65                 | 645     | 20     | audio   |
| Arts      | 5     | 100/100/100/100/62          | 5000    | 26     | text    |
| Espgame   | 6     | 100/512/1000/1000/100/5184  | 20770   | 268    | image   |

of updating  $B$  and  $Q_v$  are  $O(\sum_{v=1}^V nd_vd + \sum_{v=1}^V d_vd^2 + d^3 + nd^2)$  and  $\sum_{v=1}^V O(d_vnd + d^3 + nd^2 + d_vd^2)$ , where  $d_v$  denotes the dimensionality of the  $v$ th view. In the second optimization, the computation complexity of updating  $A_v$ ,  $U_v$  and  $C$  are  $\widehat{V}O(n^3 + n^2q)$ ,  $\widehat{V}O(n^3 + n^2q + nq^2)$  and  $\widehat{V}O(n^2q + nq^2 + q^3)$ . Owing to  $n \gg d$  and  $n \gg q$ , we can obtain the total time complexity of MFVL is  $(t_1 \sum_{v=1}^V O(d_vnd + nd^2) + t_2 \widehat{V}O(n^3))$ .  $t_1$  denotes the required number of iterations of the algorithm1 and  $t_2$  represents the number of the algorithm2.

## 4 Experiment

### 4.1 Datasets

In this paper, we use nine widely used multi-labeled datasets for experiments. These datasets cover a wide range of domains, including biology, music, images, audio, and text. Most of these datasets are taken from the Mulan dataset website [36]<sup>1</sup>, and Pascal07, Espgame and Mirflickr are three of the widely used image datasets from the Lear website [37, 38]<sup>2</sup>. Table 1 details the characteristics of each dataset.

### 4.2 Comparison methods

To evaluate the performance of our proposed method, we compare it with eight state-of-the-art multi-view multi-label learning methods.

1. VLSF<sup>3</sup> [21]: It addresses the problem of multi-view multi-label learning by learning features that are specific to each view and label. We utilize the parameter settings recommended by the original paper, where  $\lambda_1 = 10^{-1}$ ,  $\lambda_2 = 10^2$ ,  $\lambda_3 = 10^5$ ,  $\lambda_4 = 10^4$ .
2. ICM2L<sup>4</sup> [15]: It learns individuality and commonality information in multi-view multi-label data by constructing a unified model for multi-view multi-label classification. The

<sup>1</sup> datasets: <http://mulan.sourceforge.net/datasets-mlc.html>.

<sup>2</sup> datasets: <http://lear.inrialpes.fr/people/guillaumin/data.php>.

<sup>3</sup> code: <https://www.escience.cn/people/huangjun/index.html>.

<sup>4</sup> code: <https://mlda.swu.edu.cn/codes.php?name=ICM2L>.

recommended parameter settings from the original paper are used with  $\alpha = 0.6$ ,  $\beta = 0.7$  and  $k = 0.5dim$ .

3. iMVWL <sup>5</sup> [18]: An effective method for addressing incomplete view and missing label learning. The recommended parameter settings from the original paper are used with  $\alpha = 10^{-2}$  and  $\beta = 10^{-2}$ .
4. LSML <sup>6</sup> [39]: It fills the missing label matrix by jointly learning label-specific features and utilizing label correlations. The parameter settings recommended in the original paper are used for comparison experiments with  $\lambda_1 = 10^2$ ,  $\lambda_2 = 10^{-2}$ ,  $\lambda_3 = 10^{-2}$ ,  $\lambda_4 = 10^{-3}$  and  $\lambda_5 = 10^{-5}$ .
5. CDMM <sup>7</sup> [20]: A new neural network-based framework for multi-view multi-label learning. We use the parameter settings recommended in the original paper for comparison experiments, where  $\alpha = 10^{-4}$ ,  $\lambda = 10^5$ ,  $C = 1.0$ ,  $\sigma$  is searched in  $\{10^i \mid i = -2, -1, \dots, 2\}$ , and  $\eta$  is searched in 0.5, 0.6, 0.7, 0.8.
6. LVSL <sup>8</sup> [28]: It addresses non-aligned multi-view multi-label classification by learning view-specific labels. We adopt the parameter settings suggested in the original paper, where  $\lambda_1 = 10^{-1}$ ,  $\lambda_2 = 10^5$ ,  $\lambda_3 = 10^{-1}$ ,  $\lambda_4 = 10^2$  and  $\lambda_5 = 10^{-1}$ .
7. NAIM3L <sup>9</sup> [32]: It addresses the challenges of missing labels, incomplete views, and view misalignment in multi-view multi-label learning by exploiting view consensus and labeling structure. We adopt the parameter settings suggested in the original paper, where  $\lambda$  is in  $[0.1, 1]$ .
8. DICNet <sup>10</sup> [40]: It introduces a deep instance-level comparison network that uses deep neural networks to extract high-level semantic representations of samples, aiming at solving the problem of dual incompleteness of views and labels in multi-view multi-label classification. We use the parameter settings recommended in the original paper.

### 4.3 Evaluation metrics

To assess the performance of each algorithm, we select six widely used evaluation metrics for multi-label classification: Hamming Loss (HL), Ranking Loss (RL), Average Precision (AP), Area Under Curve (AUC), One Error (OE) and Coverage (CV). In our experiments, we utilize 1-HL and 1-RL as substitutes for HL and RL. For the six metrics, a higher value of 1-HL, 1-RL, AP and AUC indicates better performance, whereas lower values of OE and CV represents better performance. The specific definitions of metrics can be found in [41].

### 4.4 Experimental results and analysis

In this paper, we conduct experiments on a Windows 10 system equipped with an Intel(R) Core(TM) i5-8500 CPU and 16GB RAM. The experiments were implemented in MATLAB R2022a. To ensure the reliability of our experimental results, we employ five-fold cross-validation on each dataset and repeat each experiment five times. The final results are presented by the mean value and standard deviation of each metric, and these results are sum-

<sup>5</sup> code: <https://mlda.swu.edu.cn/codes.php?name=iMVWL>.

<sup>6</sup> code: <http://www.escience.cn/people/huangjun/index.html>.

<sup>7</sup> code: <https://github.com/chengyshaq/CDMM>.

<sup>8</sup> code: <https://github.com/zhaoawahu/LVSL>.

<sup>9</sup> code: <https://github.com/EverFAITH/NAIM3L>.

<sup>10</sup> code: <https://github.com/justsmart/DICNet>.

marized in Tables 2–3. Based on the results presented in Tables 2–3, several key observations can be found:

1. MFVL achieves the best performance in 66.7% of the experimental results (54 metrics (9 datasets  $\times$  6 metrics)), significantly outperforming the second place DICNet (11.1%).
2. MFVL consistently performs optimally on key metrics such as classification, ranking, coverage, etc. Specifically, for metrics 1-HL, OE, AP, AUC, MFVL ranks 1st on 6 datasets; for metric CV, MFVL ranks 1st on 5 datasets; for metric 1-RL, MFVL ranks 1st on 7 datasets.
3. MFVL performs robustly on datasets with different characteristics, especially in complex scenarios. On Espgame dataset, MFVL ranks 1st for all 6 metrics, outperforming DICNet, demonstrating that the multi-view fusion strategy effectively integrates the complementary information from multiple views; on Scene dataset, MFVL is optimal for all metrics, verifying the adaptability of global-local labeling coalition modeling to multi-level labeling; on Yeast dataset, MFVL ranks 1st for 5 metrics, and is only slightly inferior to CDMM for CV.

#### 4.5 Statistical hypothesis test

To further evaluate the performance differences between different algorithms on dataset, we conduct a statistical hypothesis test. Specifically, the Friedman test method [42] was used to evaluate the performance of each algorithm. Firstly, we propose the null hypothesis that all algorithms have the same performance. The Friedman test statistic  $F_F$  and the corresponding critical value for each evaluation metric are presented in Table 4. From Table 4, we observe that for each metric, the calculated  $F_F$  is greater than the critical value of 2.087 at a confidence level of  $\alpha = 0.05$ . Therefore, we reject the null hypothesis, indicating that all algorithms do not have the same performance.

Next, to compare the significance differences between two algorithms, we use the Nemenyi test as the post-hoc test. At a confidence level of  $\alpha = 0.05$ , we found that  $q_\alpha = 3.102$ . By the formula  $CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$ , we can calculate  $CD = 4.0047$ . In Figure. 2, we plot the CD diagrams for different algorithms under different evaluation metrics. In Nemenyi test, if average rank difference between two algorithms is less than the CD value, it means that there is no significant difference between the performances of these two algorithms, and they will be connected by a red solid line in the CD diagram. Finally, according to all the experimental results (Tables 2–4 and Figure 2), we can draw the following conclusions:

1. By observing the average rank values of each algorithm in Tables 2–3 and the ranking positions of different algorithms in Figure. 2, we find that our proposed method MFVL consistently has the lowest average rank value and is always ranked at the rightmost position in the CD diagrams. This indicates that MFVL outperforms other methods and has the best performance.
2. MFVL and ICM2L are significantly different for all evaluation metrics. For 5 metrics, MFVL is significantly different from VLSF, iMVWL, and LSML. This strongly suggests that multi-view fusion is more effective than subspace learning or learning individual and common information from multiple views. However, for all metrics, MFVL does not show significant differences from LVSL, CDMM, NAIM3L, and DICNet. This may be because MFVL, LVSL, and CDMM are all nonlinear models.

**Table 2** The experimental results of nine methods across nine datasets using six metrics. The best results are highlighted in bold,  $\uparrow(\downarrow)$  indicates that a higher(lower) value represents better performance. The average ranking across all datasets is listed in the last row

| Dataset            | VLSF                   | ICM2L                                | iMVWL                                | LSML                 | CDMM                   | LVSL                   | MFVL                                 | NAIM3L                               | DICNet                               |
|--------------------|------------------------|--------------------------------------|--------------------------------------|----------------------|------------------------|------------------------|--------------------------------------|--------------------------------------|--------------------------------------|
| 1-HL( $\uparrow$ ) |                        |                                      |                                      |                      |                        |                        |                                      |                                      |                                      |
| Yeast              | 0.735 $\pm$ 0.004(7)   | 0.686 $\pm$ 0.017(9)                 | 0.724 $\pm$ 0.015(8)                 | 0.754 $\pm$ 0.009(6) | 0.816 $\pm$ 0.003(2.5) | 0.815 $\pm$ 0.003(4)   | <b>0.819<math>\pm</math>0.006(1)</b> | 0.811 $\pm$ 0.003(5)                 | 0.816 $\pm$ 0.001(2.5)               |
| Emotions           | 0.702 $\pm$ 0.007(6)   | 0.651 $\pm$ 0.020(7)                 | 0.603 $\pm$ 0.015(8)                 | 0.432 $\pm$ 0.059(9) | 0.705 $\pm$ 0.013(5)   | 0.753 $\pm$ 0.006(3)   | 0.757 $\pm$ 0.011(2)                 | 0.747 $\pm$ 0.011(4)                 | <b>0.759<math>\pm</math>0.001(1)</b> |
| Espgame3k          | 0.975 $\pm$ 0.000(6)   | 0.951 $\pm$ 0.000(9)                 | 0.965 $\pm$ 0.000(8)                 | 0.969 $\pm$ 0.001(7) | 0.977 $\pm$ 0.000(5)   | 0.978 $\pm$ 0.000(4)   | <b>0.981<math>\pm</math>0.000(1)</b> | 0.980 $\pm$ 0.000(2)                 | 0.979 $\pm$ 0.000(3)                 |
| Scene              | 0.827 $\pm$ 0.003(6)   | 0.623 $\pm$ 0.004(9)                 | 0.677 $\pm$ 0.012(8)                 | 0.722 $\pm$ 0.012(7) | 0.877 $\pm$ 0.005(3)   | 0.875 $\pm$ 0.004(4)   | <b>0.886<math>\pm</math>0.003(1)</b> | 0.878 $\pm$ 0.013(2)                 | 0.846 $\pm$ 0.001(5)                 |
| Mirflickr          | 0.886 $\pm$ 0.002(5)   | 0.827 $\pm$ 0.004(7)                 | 0.748 $\pm$ 0.001(8)                 | 0.352 $\pm$ 0.016(9) | 0.895 $\pm$ 0.002(2.5) | 0.895 $\pm$ 0.002(2.5) | <b>0.900<math>\pm</math>0.002(1)</b> | 0.881 $\pm$ 0.000(6)                 | 0.888 $\pm$ 0.002(4)                 |
| Pascal07           | 0.934 $\pm$ 0.000(3.5) | 0.836 $\pm$ 0.000(8)                 | 0.885 $\pm$ 0.005(7)                 | 0.590 $\pm$ 0.009(9) | 0.934 $\pm$ 0.001(3.5) | 0.936 $\pm$ 0.000(2)   | <b>0.938<math>\pm</math>0.000(1)</b> | 0.928 $\pm$ 0.000(6)                 | 0.929 $\pm$ 0.001(5)                 |
| Birds              | 0.023 $\pm$ 0.001(9)   | <b>0.871<math>\pm</math>0.008(1)</b> | 0.836 $\pm$ 0.002(2)                 | 0.028 $\pm$ 0.003(6) | 0.026 $\pm$ 0.001(8)   | 0.029 $\pm$ 0.002(5)   | 0.027 $\pm$ 0.002(7)                 | 0.047 $\pm$ 0.001(4)                 | 0.127 $\pm$ 0.012(3)                 |
| Arts               | 0.001 $\pm$ 0.000(9)   | <b>0.883<math>\pm</math>0.002(1)</b> | 0.877 $\pm$ 0.004(2)                 | 0.015 $\pm$ 0.000(5) | 0.004 $\pm$ 0.000(7)   | 0.002 $\pm$ 0.000(8)   | 0.005 $\pm$ 0.000(6)                 | 0.035 $\pm$ 0.001(4)                 | 0.101 $\pm$ 0.003(3)                 |
| Espgame            | 0.983 $\pm$ 0.000(3)   | 0.964 $\pm$ 0.002(9)                 | 0.971 $\pm$ 0.000(8)                 | 0.981 $\pm$ 0.000(7) | 0.982 $\pm$ 0.000(5.5) | 0.983 $\pm$ 0.000(3)   | <b>0.984<math>\pm</math>0.000(1)</b> | 0.982 $\pm$ 0.000(5.5)               | 0.983 $\pm$ 0.000(3)                 |
| Average rank       | 6.0555                 | 6.6667                               | 6.5556                               | 7.2222               | 4.6667                 | 3.9444                 | 2.3333                               | 4.2777                               | 3.2777                               |
| OE( $\downarrow$ ) |                        |                                      |                                      |                      |                        |                        |                                      |                                      |                                      |
| Yeast              | 0.350 $\pm$ 0.017(9)   | 0.251 $\pm$ 0.003(7)                 | 0.311 $\pm$ 0.029(8)                 | 0.238 $\pm$ 0.023(6) | 0.217 $\pm$ 0.016(5)   | 0.205 $\pm$ 0.015(2)   | <b>0.203<math>\pm</math>0.013(1)</b> | 0.206 $\pm$ 0.003(3)                 | 0.213 $\pm$ 0.011(4)                 |
| Emotions           | 0.509 $\pm$ 0.081(7)   | 0.487 $\pm$ 0.076(6)                 | 0.550 $\pm$ 0.059(8)                 | 0.554 $\pm$ 0.040(9) | 0.478 $\pm$ 0.048(5)   | 0.428 $\pm$ 0.021(4)   | <b>0.392<math>\pm</math>0.033(1)</b> | 0.412 $\pm$ 0.006(3)                 | 0.393 $\pm$ 0.013(2)                 |
| Espgame3k          | 0.624 $\pm$ 0.019(8)   | 0.713 $\pm$ 0.030(9)                 | 0.435 $\pm$ 0.038(6)                 | 0.504 $\pm$ 0.014(7) | 0.376 $\pm$ 0.021(2)   | 0.394 $\pm$ 0.011(5)   | <b>0.375<math>\pm</math>0.019(1)</b> | 0.385 $\pm$ 0.011(3)                 | 0.393 $\pm$ 0.009(4)                 |
| Scene              | 0.835 $\pm$ 0.019(9)   | 0.806 $\pm$ 0.031(8)                 | 0.676 $\pm$ 0.032(7)                 | 0.631 $\pm$ 0.024(6) | 0.342 $\pm$ 0.020(5)   | 0.327 $\pm$ 0.021(4)   | <b>0.308<math>\pm</math>0.002(1)</b> | 0.311 $\pm$ 0.012(2)                 | 0.318 $\pm$ 0.012(3)                 |
| Mirflickr          | 0.335 $\pm$ 0.013(6)   | 0.600 $\pm$ 0.036(9)                 | 0.546 $\pm$ 0.001(8)                 | 0.352 $\pm$ 0.016(7) | 0.301 $\pm$ 0.019(4)   | 0.307 $\pm$ 0.013(5)   | 0.284 $\pm$ 0.012(3)                 | <b>0.279<math>\pm</math>0.002(1)</b> | 0.281 $\pm$ 0.012(2)                 |
| Pascal07           | 0.507 $\pm$ 0.005(6)   | 0.589 $\pm$ 0.002(7)                 | 0.623 $\pm$ 0.060(9)                 | 0.590 $\pm$ 0.009(8) | 0.492 $\pm$ 0.016(5)   | 0.484 $\pm$ 0.011(3)   | <b>0.451<math>\pm</math>0.010(1)</b> | 0.473 $\pm$ 0.011(2)                 | 0.491 $\pm$ 0.016(4)                 |
| Birds              | 0.347 $\pm$ 0.030(9)   | <b>0.101<math>\pm</math>0.005(1)</b> | 0.106 $\pm$ 0.006(2)                 | 0.276 $\pm$ 0.056(7) | 0.303 $\pm$ 0.040(8)   | 0.262 $\pm$ 0.018(5)   | 0.259 $\pm$ 0.025(4)                 | 0.266 $\pm$ 0.015(6)                 | 0.253 $\pm$ 0.021(3)                 |
| Arts               | 0.564 $\pm$ 0.015(9)   | 0.200 $\pm$ 0.000(2)                 | <b>0.134<math>\pm</math>0.005(1)</b> | 0.456 $\pm$ 0.018(5) | 0.496 $\pm$ 0.013(7)   | 0.508 $\pm$ 0.078(8)   | 0.469 $\pm$ 0.006(6)                 | 0.419 $\pm$ 0.016(4)                 | 0.369 $\pm$ 0.001(3)                 |
| Espgame            | 0.572 $\pm$ 0.005(8)   | 0.994 $\pm$ 0.002(9)                 | 0.489 $\pm$ 0.003(6)                 | 0.544 $\pm$ 0.014(7) | 0.465 $\pm$ 0.009(4.5) | 0.465 $\pm$ 0.009(4.5) | <b>0.429<math>\pm</math>0.003(1)</b> | 0.433 $\pm$ 0.013(3)                 | 0.430 $\pm$ 0.001(2)                 |
| Average rank       | 7.8889                 | 6.4444                               | 6.1111                               | 6.8889               | 5.0555                 | 4.5                    | 2.1111                               | 3                                    | 3                                    |

**Table 2** continued

| Dataset      | VLSF             | ICM2L            | iMVWL          | LSML           | CDMM           | LVSL           | MFVL                  | NAIM3L                | DICNet                |
|--------------|------------------|------------------|----------------|----------------|----------------|----------------|-----------------------|-----------------------|-----------------------|
| 1-RL(†)      |                  |                  |                |                |                |                |                       |                       |                       |
| Yeast        | 0.676±0.010(9)   | 0.713±0.043(8)   | 0.780±0.020(7) | 0.812±0.009(6) | 0.848±0.004(4) | 0.852±0.006(2) | <b>0.854±0.006(1)</b> | 0.845±0.003(5)        | 0.851±0.001(3)        |
| Emotions     | 0.698±0.019(6)   | 0.624±0.028(7)   | 0.583±0.024(9) | 0.589±0.018(8) | 0.723±0.030(5) | 0.767±0.008(4) | 0.782±0.015(2)        | <b>0.791±0.015(1)</b> | 0.778±0.011(3)        |
| Espgame3k    | 0.663±0.010(9)   | 0.797±0.002(8)   | 0.834±0.002(6) | 0.810±0.003(7) | 0.872±0.003(3) | 0.867±0.005(4) | <b>0.876±0.002(1)</b> | 0.866±0.001(5)        | 0.874±0.012(2)        |
| Scene        | 0.526±0.013(9)   | 0.533±0.007(8)   | 0.662±0.019(7) | 0.683±0.019(6) | 0.865±0.008(5) | 0.869±0.007(4) | <b>0.881±0.008(1)</b> | 0.871±0.001(3)        | 0.879±0.003(2)        |
| Mirflickr    | 0.872±0.005(4)   | 0.743±0.016(8)   | 0.765±0.032(7) | 0.357±0.008(9) | 0.883±0.001(3) | 0.885±0.004(2) | <b>0.897±0.004(1)</b> | 0.844±0.001(6)        | 0.863±0.004(5)        |
| Pascal07     | 0.875±0.003(4.5) | 0.875±0.003(4.5) | 0.753±0.000(8) | 0.295±0.006(9) | 0.887±0.005(2) | 0.883±0.001(3) | <b>0.891±0.000(1)</b> | 0.783±0.001(7)        | 0.799±0.008(6)        |
| Birds        | 0.903±0.008(5.5) | 0.788±0.065(8)   | 0.016±0.001(9) | 0.91±0.018(4)  | 0.900±0.005(7) | 0.932±0.017(2) | <b>0.933±0.012(1)</b> | 0.931±0.002(3)        | 0.903±0.006(5.5)      |
| Arts         | 0.861±0.003(7)   | 0.833±0.042(8)   | 0.094±0.005(9) | 0.866±0.005(6) | 0.869±0.003(5) | 0.883±0.004(4) | 0.891±0.004(2)        | 0.889±0.003(3)        | <b>0.893±0.001(1)</b> |
| Espgame      | 0.846±0.001(4)   | 0.796±0.001(9)   | 0.803±0.001(8) | 0.816±0.013(7) | 0.876±0.002(2) | 0.854±0.001(3) | <b>0.885±0.000(1)</b> | 0.818±0.001(6)        | 0.832±0.001(5)        |
| Average rank | 6.4444           | 7.6111           | 7.7777         | 6.8888         | 4              | 3.1111         | 1.2222                | 4.3333                | 3.6111                |

**Table 3** The remainder of Table 2

| Dataset      | VLSF             | ICM2L          | iMVWL            | LSML                  | CDMM                  | LVSL                  | MFVL                    | NAIM3L         | DICNet                  |
|--------------|------------------|----------------|------------------|-----------------------|-----------------------|-----------------------|-------------------------|----------------|-------------------------|
| <b>CV(↓)</b> |                  |                |                  |                       |                       |                       |                         |                |                         |
| Yeast        | 0.598±0.009(8)   | 0.623±0.070(9) | 0.497±0.022(7)   | 0.475±0.009(6)        | <b>0.422±0.010(1)</b> | 0.424±0.007(2.5)      | 0.424±0.007(2.5)        | 0.429±0.003(4) | 0.434±0.001(5)          |
| Emotions     | 0.415±0.010(6)   | 0.474±0.009(7) | 0.524±0.030(9)   | 0.518±0.014(8)        | 0.393±0.025(5)        | 0.348±0.008(4)        | <b>0.330±0.011(1)</b>   | 0.333±0.013(2) | 0.334±0.014(3)          |
| Espgame3k    | 0.819±0.011(9)   | 0.479±0.001(7) | 0.476±0.006(6)   | 0.550±0.006(8)        | 0.407±0.006(4)        | 0.414±0.013(5)        | <b>0.374±0.005(1)</b>   | 0.394±0.005(3) | 0.383±0.000(2)          |
| Scene        | 0.408±0.011(9)   | 0.404±0.008(8) | 0.298±0.016(7)   | 0.279±0.017(6)        | 0.127±0.006(5)        | 0.122±0.005(4)        | <b>0.112±0.006(1)</b>   | 0.116±0.001(3) | 0.114±0.003(2)          |
| Mirflickr    | 0.341±0.012(7)   | 0.539±0.004(8) | 0.766±0.002(9)   | <b>0.016±0.004(1)</b> | 0.314±0.007(4)        | 0.317±0.007(5)        | 0.298±0.006(2)          | 0.318±0.001(6) | 0.301±0.006(3)          |
| Pascal07     | 0.168±0.004(7)   | 0.558±0.001(9) | 0.308±0.000(8)   | <b>0.074±0.012(1)</b> | 0.155±0.006(4)        | 0.160±0.001(5)        | 0.152±0.003(2)          | 0.162±0.001(6) | 0.154±0.013(3)          |
| Birds        | 0.146±0.013(6)   | 0.929±0.003(9) | 0.696±0.003(8)   | 0.141±0.016(5)        | 0.151±0.008(7)        | <b>0.108±0.027(1)</b> | 0.109±0.021(2)          | 0.119±0.011(4) | 0.111±0.001(3)          |
| Arts         | 0.204±0.007(6)   | 0.867±0.000(9) | 0.595±0.004(8)   | 0.207±0.008(7)        | 0.196±0.007(5)        | 0.176±0.006(4)        | <b>0.168±0.007(1)</b>   | 0.173±0.004(3) | 0.171±0.017(2)          |
| Espgame      | 0.396±0.005(6.5) | 0.999±0.004(9) | 0.396±0.006(6.5) | 0.506±0.004(8)        | 0.337±0.005(4)        | 0.386±0.004(5)        | <b>0.305±0.001(1)</b>   | 0.315±0.001(3) | 0.314±0.004(2)          |
| Average rank | 7.1666           | 8.3333         | 7.6111           | 5.5555                | 4.3333                | 3.9444                | 1.5                     | 3.7777         | 2.7777                  |
| <b>AP(↑)</b> |                  |                |                  |                       |                       |                       |                         |                |                         |
| Yeast        | 0.622±0.009(9)   | 0.634±0.037(8) | 0.693±0.021(7)   | 0.736±0.014(6)        | 0.779±0.009(4)        | 0.787±0.010(3)        | <b>0.790±0.009(1.5)</b> | 0.771±0.004(5) | <b>0.790±0.019(1.5)</b> |
| Emotions     | 0.650±0.014(6)   | 0.626±0.026(7) | 0.573±0.020(9)   | 0.574±0.016(8)        | 0.669±0.027(5)        | 0.713±0.007(3.5)      | <b>0.736±0.014(1)</b>   | 0.733±0.004(2) | 0.713±0.014(3.5)        |
| Espgame3k    | 0.241±0.010(8)   | 0.219±0.130(9) | 0.326±0.012(7)   | 0.349±0.005(6)        | 0.413±0.009(4)        | 0.408±0.005(5)        | <b>0.424±0.004(1)</b>   | 0.414±0.001(3) | 0.423±0.000(2)          |
| Scene        | 0.426±0.015(9)   | 0.438±0.013(8) | 0.550±0.021(7)   | 0.577±0.018(6)        | 0.786±0.011(5)        | 0.796±0.011(4)        | <b>0.810±0.013(1)</b>   | 0.801±0.03(3)  | 0.807±0.009(2)          |
| Mirflickr    | 0.615±0.001(6)   | 0.415±0.019(8) | 0.232±0.004(9)   | 0.829±0.002(2)        | 0.629±0.005(4)        | 0.622±0.009(5)        | 0.650±0.007(3)          | 0.551±0.019(7) | <b>0.849±0.004(1)</b>   |
| Pascal07     | 0.597±0.004(5)   | 0.185±0.000(9) | 0.458±0.029(8)   | <b>0.765±0.006(1)</b> | 0.616±0.011(4)        | 0.618±0.006(3)        | 0.640±0.004(2)          | 0.488±0.003(7) | 0.505±0.012(6)          |
| Birds        | 0.731±0.015(7)   | 0.576±0.066(8) | 0.534±0.005(9)   | 0.770±0.042(5)        | 0.752±0.019(6)        | 0.798±0.017(3)        | 0.797±0.017(4)          | 0.799±0.017(2) | <b>0.801±0.001(1)</b>   |
| Arts         | 0.560±0.010(7)   | 0.266±0.055(8) | 0.208±0.001(9)   | 0.624±0.014(2)        | 0.599±0.007(4.5)      | 0.599±0.005(4.5)      | <b>0.627±0.005(1)</b>   | 0.617±0.005(3) | 0.597±0.003(6)          |
| Espgame      | 0.321±0.002(5)   | 0.220±0.002(9) | 0.236±0.001(8)   | 0.379±0.013(4)        | 0.400±0.003(2)        | 0.392±0.001(3)        | <b>0.430±0.002(1)</b>   | 0.246±0.001(7) | 0.297±0.00(6)           |
| Average rank | 6.8888           | 8.2222         | 8.1111           | 4.4444                | 4.2777                | 3.7777                | 1.5                     | 4.3333         | 3.2222                  |

**Table 3** continued

| Dataset      | VLSF             | ICM2L          | iMVWL          | LSML                  | CDMM             | LVSL           | MFVL                  | NAIM3L                | DICNet                |
|--------------|------------------|----------------|----------------|-----------------------|------------------|----------------|-----------------------|-----------------------|-----------------------|
| AUC(↑)       |                  |                |                |                       |                  |                |                       |                       |                       |
| Yeast        | 0.662±0.009(9)   | 0.727±0.041(8) | 0.793±0.019(7) | 0.799±0.008(6)        | 0.834±0.004(5)   | 0.853±0.006(3) | <b>0.855±0.006(1)</b> | 0.854±0.006(2)        | 0.851±0.001(4)        |
| Emotions     | 0.670±0.021(7)   | 0.677±0.023(6) | 0.644±0.026(8) | 0.555±0.018(9)        | 0.696±0.027(5)   | 0.822±0.010(4) | 0.833±0.014(3)        | <b>0.836±0.004(1)</b> | 0.834±0.001(2)        |
| Espgame3k    | 0.676±0.008(8)   | 0.466±0.000(9) | 0.836±0.001(6) | 0.806±0.004(7)        | 0.872±0.002(3)   | 0.870±0.004(4) | <b>0.876±0.002(1)</b> | 0.856±0.002(5)        | 0.874±0.002(2)        |
| Scene        | 0.509±0.011(9)   | 0.621±0.011(8) | 0.742±0.019(7) | 0.869±0.001(5)        | 0.837±0.008(6)   | 0.901±0.004(4) | <b>0.910±0.005(1)</b> | 0.905±0.005(3)        | 0.909±0.000(2)        |
| Mirflickr    | 0.839±0.004(5)   | 0.743±0.006(7) | 0.413±0.001(9) | <b>0.888±0.005(1)</b> | 0.852±0.005(4)   | 0.870±0.004(3) | 0.881±0.003(2)        | 0.837±0.000(6)        | 0.589±0.005(8)        |
| Pascal07     | 0.870±0.002(4)   | 0.480±0.001(9) | 0.778±0.005(8) | 0.869±0.001(5)        | 0.880±0.004(3)   | 0.900±0.000(2) | <b>0.905±0.001(1)</b> | 0.811±0.001(6)        | 0.809±0.006(7)        |
| Birds        | 0.867±0.008(6.5) | 0.654±0.045(8) | 0.423±0.044(9) | 0.879±0.015(5)        | 0.867±0.009(6.5) | 0.912±0.020(3) | <b>0.917±0.015(1)</b> | 0.911±0.011(4)        | 0.913±0.005(2)        |
| Arts         | 0.824±0.005(6.5) | 0.532±0.027(8) | 0.484±0.001(9) | 0.824±0.004(6.5)      | 0.833±0.005(5)   | 0.866±0.005(4) | 0.871±0.005(2)        | 0.867±0.003(3)        | <b>0.873±0.001(1)</b> |
| Espgame      | 0.848±0.002(3)   | 0.803±0.001(9) | 0.808±0.001(8) | 0.817±0.001(7)        | 0.835±0.006(5)   | 0.855±0.001(2) | <b>0.888±0.000(1)</b> | 0.824±0.001(6)        | 0.836±0.001(4)        |
| Average rank | 6.4444           | 8              | 7.8888         | 5.7222                | 4.7222           | 3.2222         | 1.4444                | 4                     | 3.5555                |

**Table 4** The summary of the Friedman Statistics  $F_F$  and the critical value in terms of each evaluation metric ( $K=9$ ,  $N=9$ ,  $\alpha = 0.05$ )

| Metric | $F_F$   | Critical value( $\alpha = 0.05$ ) |
|--------|---------|-----------------------------------|
| 1-HL   | 4.9989  | 2.087                             |
| OE     | 9.0414  |                                   |
| 1-RL   | 17.8081 |                                   |
| CV     | 20.5253 |                                   |
| AP     | 16.4294 |                                   |
| AUC    | 14.7701 |                                   |

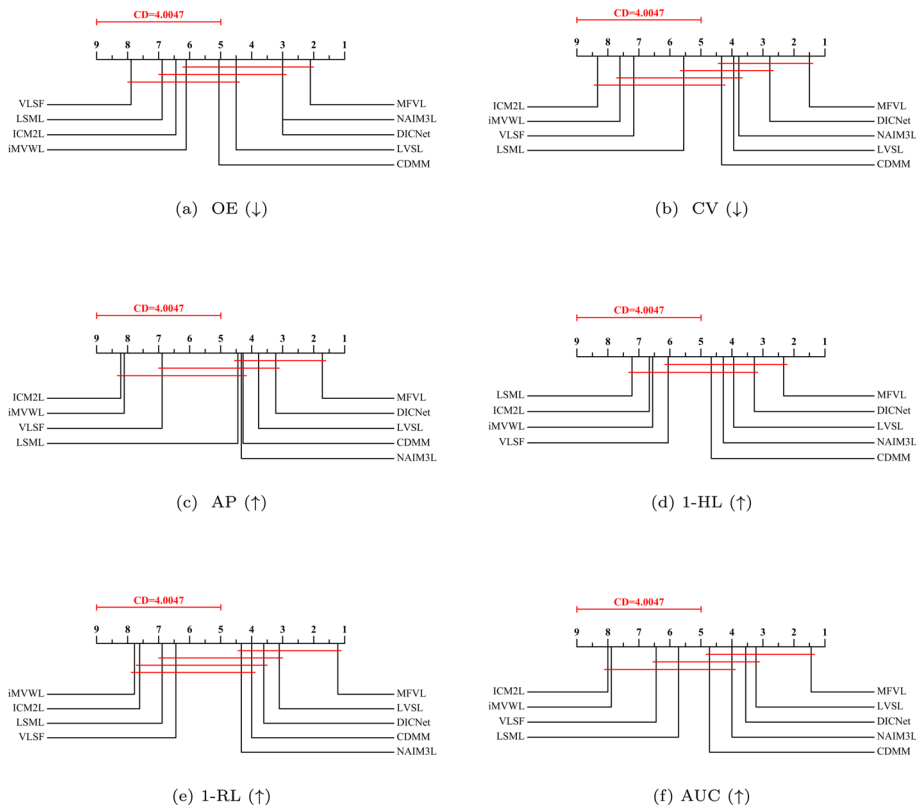


**Fig. 2** Comparing the significance differences in performance between MFVL and other comparative methods using the Nemenyi test ( $\alpha = 0.05$ )

## 4.6 Sensitivity analysis

In our proposed method, we have eight key parameters that significantly affect the model's performance and behavior. Here's a concise explanation of each parameter:

- $\lambda_1$  controls the model's complexity. Its value range is  $\{10^i | i = -4, -3, \dots, 3\}$ . A larger  $\lambda_1$  increases the model's regularization strength, reducing complexity and preventing overfitting, while a smaller  $\lambda_1$  allows the model to learn more complex features.

- $\lambda_2$  balances the contributions from different views. Its value range is  $\{10^i | i = -1, 0, \dots, 6\}$ . A larger  $\lambda_2$  enhances the influence of certain views, while a smaller  $\lambda_2$  makes the model rely more on other views.
- $\lambda_3$  controls the structural diversity of view-specific labels. Its value range is  $\{10^i | i = -3, -2, \dots, 4\}$ . A larger  $\lambda_3$  enhances label relationships, while a smaller  $\lambda_3$  reduces these relationships.
- $\lambda_4$  controls the information consistency between view-specific labels and ground truth labels. Its value range is  $\{10^i | i = -4, -3, \dots, 5\}$ . A larger  $\lambda_4$  strengthens label consistency, while a smaller  $\lambda_4$  weakens label consistency.
- $\lambda_5$  balances the impact of reconstructing loss by utilizing label correlation to supplement the original label matrix. Its value range is  $\{10^i | i = -5, -4, \dots, 3\}$ . A larger  $\lambda_5$  enhances the role of label correlation, while a smaller  $\lambda_5$  reduces the model's label correlation.
- $\lambda_6$  controls the influence of local label smoothing. Its value range is  $\{10^i | i = -3, -2, \dots, 4\}$ . A larger  $\lambda_6$  enhances the local smoothing effect, while a smaller  $\lambda_6$  reduces the model's robustness to noisy labels.
- $k$  controls the neighborhood size. Its value range is  $\{2, 5, 10, 15, 20, 50\}$ . A larger  $k$  introduces noise, while a smaller  $k$  leads to sparsity.
- $\sigma$  controls the kernel width. Its value range is  $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ . A larger  $\sigma$  may result in all data points being considered "near neighbors", while a smaller  $\sigma$  may result in only very close data points being considered "near neighbors".

To verify the impact of parameter selection on experimental results, we conduct parameter sensitivity experiments. Due to space limitations, we only present the results on the Yeast and Mirflickr datasets, and the similar findings can observe on the other datasets. Figures 3–4 illustrate the effects of parameter variations for evaluation metrics.

During the experiments, we found that the parameters significantly influence the model's performance. For example, a smaller  $\lambda_1$  value allows the model to learn more complex feature representations, thereby enhancing the model's expressive power but potentially leading to overfitting. In contrast, a larger  $\lambda_1$  value enhances the model's generalization ability, but it may reduce the model's precision. Similarly, adjusting  $\lambda_2$  optimizes the fusion of multi-view information, thereby improving the model's ability to fit multi-source data. The adjustments of  $\lambda_3$  and  $\lambda_4$  directly affect the model's label information, thereby influencing the model's prediction accuracy. Moreover, the adjustments of  $\lambda_5$  and  $\lambda_6$  optimize the model's robustness to label noise and local smoothing effects, thereby enhancing the model's robustness and stability. Meanwhile, the similarity graph forms the basis for feature and label manifold regularization, where the selection of neighborhood size  $k$  and kernel width  $\sigma$  critically influences its construction.

Through experimentation, we determine the optimal parameter settings as follows:  $\lambda_1 = 10^{-1}$ ,  $\lambda_2 = 10^5$ ,  $\lambda_3 = 10^{-1}$ ,  $\lambda_4 = 10^4$ ,  $\lambda_5 = 10^2$ ,  $\lambda_6 = 10^{-1}$ ,  $k = 10$ ,  $\sigma = 0.5$ . These parameter settings achieve the best performance across multiple datasets, demonstrating their significant role in enhancing model performance.

## 4.7 Ablation experiment

In our method, MFVL consists of three main modules: multi-view fusion, view-specific labels, and global and local label correlations. We perform ablation experiments on Yeast and Emotions datasets to validate each module's effect, and the results are presented in Figure 5. The various variants of the algorithm corresponding to MFVL are defined as follows:

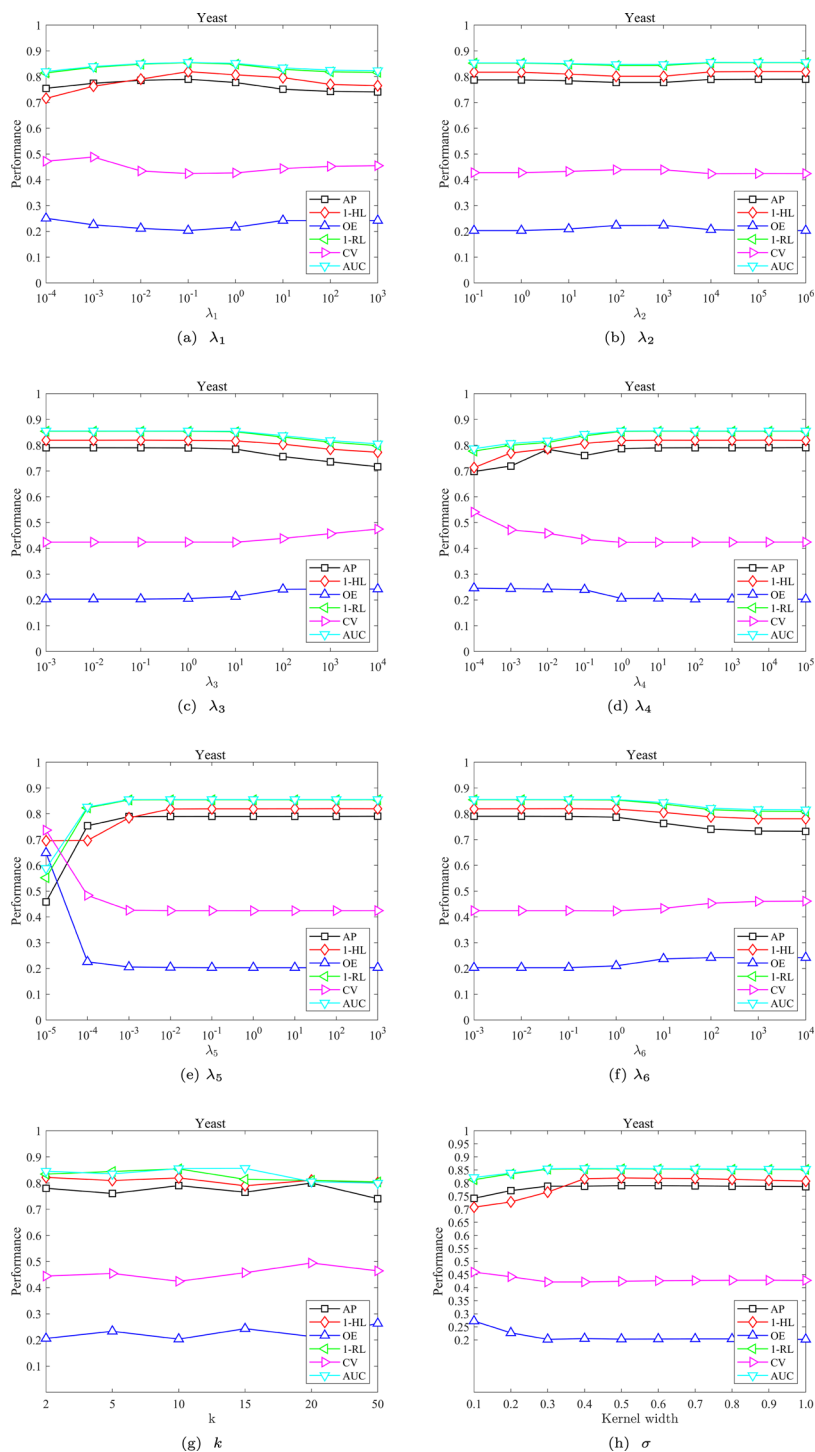


**Fig. 3** Parameter sensitivity analysis of MFVL algorithm on Yeast dataset

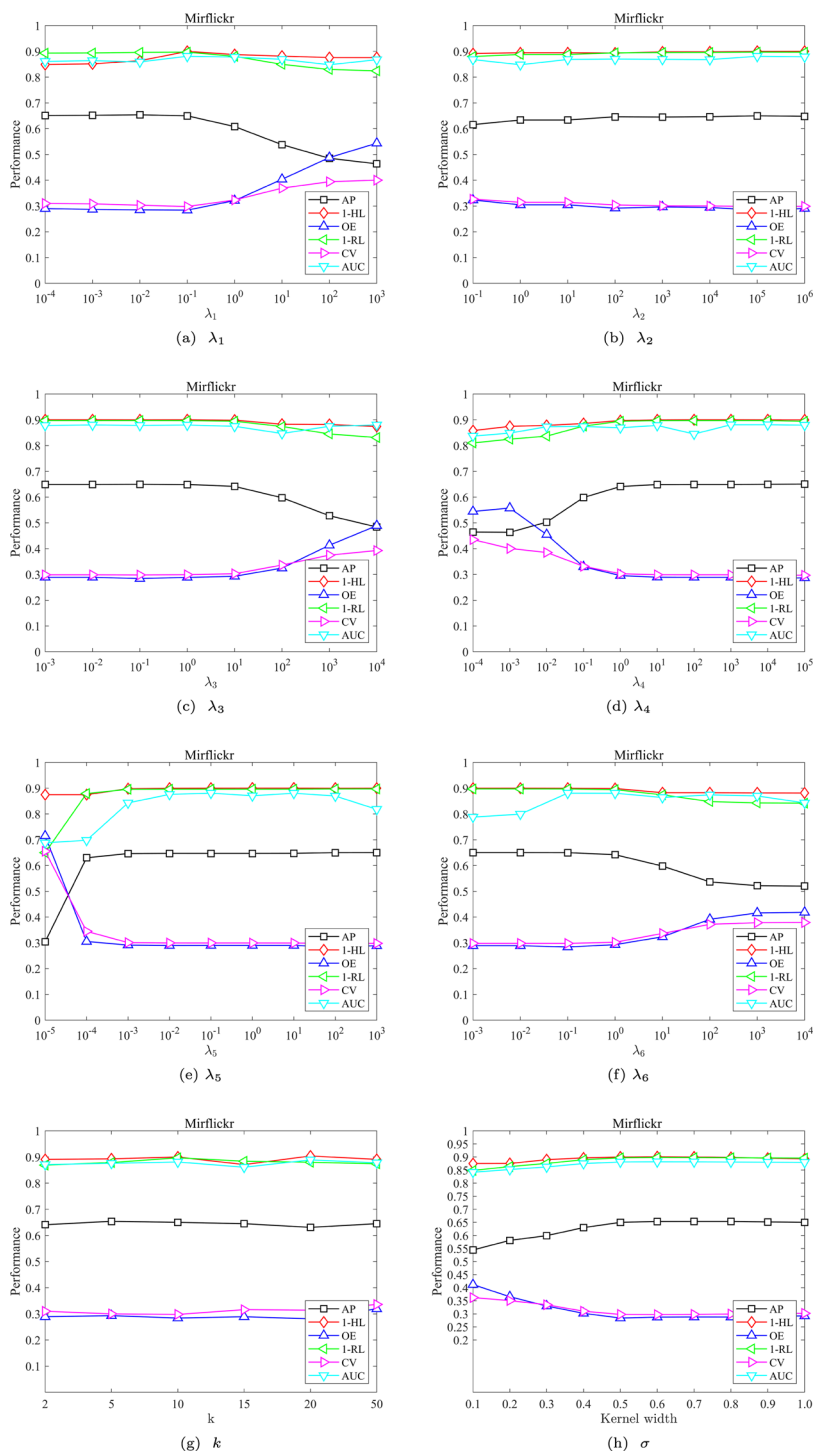


**Fig. 4** Parameter sensitivity analysis of MFVL algorithm on Mirflickr dataset

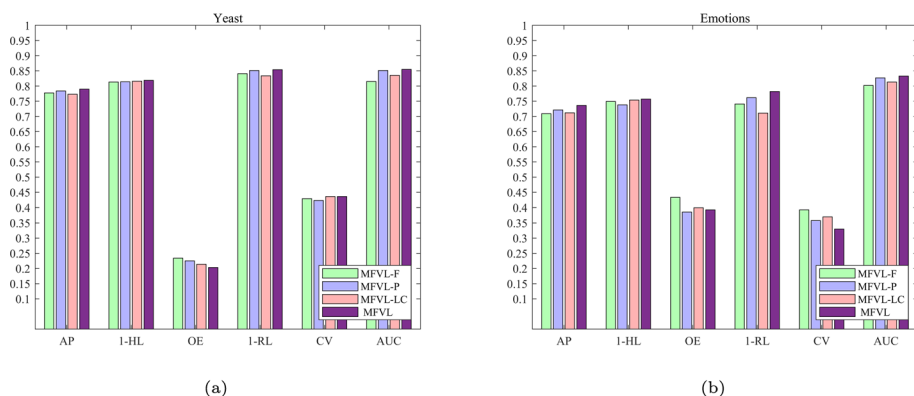


**Fig. 5** The Ablation experimental results of MFVL on Yeast and Emotions datasets

For MFVL-F, we remove the multi-view fusion step and perform multi-view multi-label learning on original dataset. For MFVL-P, we learn a shared label space. For MFVL-LC, we ignore the impact of label correlations. By comparing MFVL-F with MFVL, we find that the performance is significantly improved by fusing multiple views. This indicates that multi-view fusion can learn multi-view information more comprehensively. Comparing MFVL-P and MFVL, we found that incorporating view-specific labels enhances overall performance. However, further analysis reveals that sometimes MFVL-P performs similarly to MFVL, indicating that learning view-specific labels can be improved. We compare MFVL-LC with MFVL and observe significantly improved experimental results by incorporating global and local label correlation. These findings indicate that multi-view multi-label learning is effectively improved by considering both global and local label correlations. Similar results are obtained on other datasets.

In addition, Figure 6 presents a visualization of the prediction results on the Espgame dataset, where the numbers in parentheses represent the rankings assigned by the model to the likelihood that an image belongs to the corresponding category. A smaller number indicates a higher likelihood that the model considers the image to belong to that category. Through specific examples, it can be shown that view-specific labeling and fusion can effectively improve the accuracy of classification.

#### 4.8 Convergence analysis

In this section, we analyze the convergence of the MFVL algorithm. Figure 7 demonstrates the objective function loss values for each iteration of our method on Yeast and Mirflickr datasets. The objective function value of MFVL exhibits a sharp decrease from its highest value at about 20 iterations, and the convergence trend gradually stabilizes after 20 iterations. Overall, after 100 iterations, MFVL achieves convergence on both Yeast and Mirflickr datasets. We can get the same conclusion on other datasets as well.

#### 4.9 Runtime analysis

To verify the proposed method MFVL's computational efficiency, we compare the running time of multi-view multi-label learning methods on different datasets. The experimental



(a) Sample ID: Espgame#36063  
 Labels: {airplane, fly, plane, red, sky, wheel}  
 DICNet{1, 14, 2, 9, 13}  
 NAIM3L{1, 9, 2, 6, 7, 46}  
 MFVL{1, 5, 2, 4, 3, 6}



(b) Sample ID: Espgame#17772  
 Labels: {blue, dog, fence, hat, man}  
 DICNet{3, 4, 9, 5, 11}  
 NAIM3L{2, 4, 6, 7, 3}  
 MFVL{3, 4, 2, 5, 6}



(c) Sample ID: Espgame#20735  
 Labels: {cloud, dark, lake, sky, tree, water}  
 DICNet{3, 2, 1, 4, 6, 5}  
 NAIM3L{3, 4, 1, 2, 5, 6}  
 MFVL{2, 3, 1, 4, 5, 6}



(d) Sample ID: Espgame#16141  
 Labels: {guitar, music, white}  
 DICNet{1, 3, 7}  
 NAIM3L{1, 4, 5}  
 MFVL{1, 3, 2}

**Fig. 6** Visualization of the prediction results on the Espgame dataset

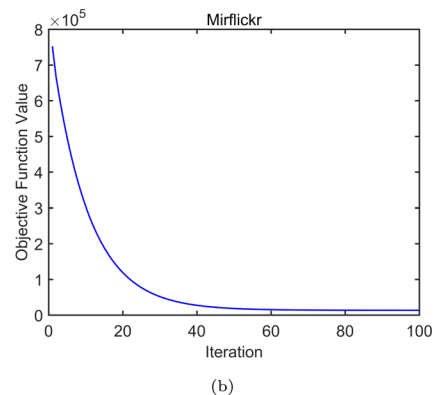
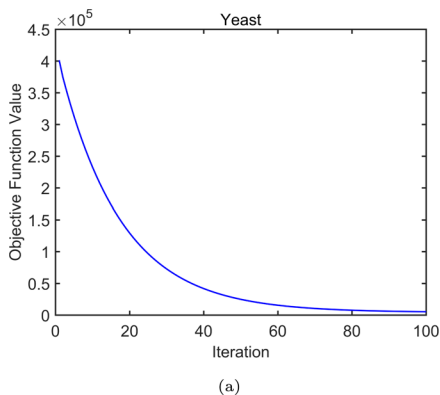


**Fig. 7** Convergence trend analysis of MFVL on Yeast and Mirflickr datasets

**Table 5** Average running time (second) of different methods on different datasets

| Datasets  | VLSF  | ICM2L  | iMVWL  | LSML   | CDMM   | LVSL   | MFVL   | NAIM3L | DICNet |
|-----------|-------|--------|--------|--------|--------|--------|--------|--------|--------|
| Yeast     | 2.2   | 29.7   | 15.6   | 12.5   | 14.6   | 25.7   | 21.1   | 8.2    | 9.8    |
| Emotions  | 0.2   | 3.3    | 2.5    | 4.3    | 19.4   | 1.2    | 1.9    | 2.4    | 3.5    |
| Espgame3k | 49.8  | 3048.5 | 945.5  | 680.2  | 259.7  | 139.9  | 128.4  | 427.7  | 3210.1 |
| Scene     | 1.2   | 14.6   | 10.7   | 9.1    | 35.3   | 68.6   | 45.1   | 12.2   | 18.5   |
| Mirflickr | 5.7   | 834.5  | 222.8  | 180.5  | 114.8  | 112.7  | 119.1  | 151.1  | 804.4  |
| Pascal07  | 3.1   | 891.0  | 292.9  | 453.5  | 1597.7 | 2106.9 | 714.9  | 607.4  | 2511.1 |
| Birds     | 0.8   | 14.0   | 11.7   | 8.1    | 4.1    | 2.9    | 7.4    | 5.6    | 6.9    |
| Arts      | 7.4   | 144.3  | 34.5   | 62.2   | 199.3  | 288.5  | 205.2  | 85.4   | 354.5  |
| Espgame   | 678.6 | 4189.2 | 3201.1 | 3451.1 | 2010.4 | 1867.8 | 6434.8 | 2211.2 | 6474.1 |

results are shown in Table 5. Here are the specific findings: For large-scale datasets, MFVL shows a significant lead in efficiency. For example, on the Pascal07 dataset, MFVL takes 714.9 seconds, while LVSL requires 2106.9 seconds and DICNet needs 2511.1 seconds. This is because MFVL's two-steps optimization strategy, which significantly reduces computational complexity. For medium-sized datasets, MFVL does really well in both efficiency and accuracy. On the Mirflickr dataset, MFVL takes 119.1 seconds, which is much lower than the label-completion method LSML (180.5 seconds). Although it is slightly slower than the lightweight network CDMM (114.8 seconds) and the non-aligned-view method LVSL (112.7 seconds), MFVL is more competitive in terms of classification accuracy. For small-scale datasets, MFVL's efficiency is close to traditional methods. On the Emotions dataset, MFVL takes 1.9 seconds, slightly higher than the VLSF method (0.2 seconds), but much lower than the neural network method CDMM (19.4 seconds) and the complex model ICM2L (3.3 seconds); on the Birds dataset, MFVL takes 7.4 seconds, close to the label-completion method LSML (8.1 seconds).

## 5 Conclusion

In this article, we propose a novel method for multi-view multi-label learning. We perform subspace learning on the original multi-view data to obtain a low-dimensional subspace and integrate it with the original data to create a new multi-view dataset, while learning view-specific label spaces instead of a shared label space. Furthermore, we consider both global and local label correlations, assigning appropriate view weight parameters for each view. To address the issue of linear inseparability, we extend our model to nonlinear version using kernel functions. Experimental results on benchmark datasets showed that our method is superior to current state-of-the-art algorithms. In the future work, we will extend our method to incomplete multi-label data and will consider utilize parallel computing and distributed processing technique to reduce the training time.

**Acknowledgements** This work is supported by the National Natural Science Foundation of China (No. 62476166) and the Shanghai Pujiang Program (No. 22PJJD029).

**Open Access** This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

## References

1. Li Y, Yang M, Zhang Z (2018) A survey of multi-view representation learning. *IEEE Trans Knowl Data Eng* 31(10):1863–1883
2. Sun S, Dong W, Liu Q (2020) Multi-view representation learning with deep gaussian processes. *IEEE Trans Pattern Anal Mach Intell* 43(12):4453–4468
3. Yin J, Sun S (2019) Multiview uncorrelated locality preserving projection. *IEEE Transactions on Neural Networks and Learning Systems* 31(9):3442–3455

4. Wu F, Jing X-Y, You X, Yue D, Hu R, Yang J-Y (2016) Multi-view low-rank dictionary learning for image classification. *Pattern Recogn* 50:143–154
5. Yin J, Sun S (2023) Incomplete multi-view clustering with reconstructed views. *IEEE Trans Knowl Data Eng* 35(3):2671–2682
6. Boutell MR, Luo J, Shen X, Brown CM (2004) Learning multi-label scene classification. *Pattern Recogn* 37(9):1757–1771
7. Zhang M-L, Wu L (2014) Lift: Multi-label learning with label-specific features. *IEEE Trans Pattern Anal Mach Intell* 37(1):107–120
8. Huang J, Qin F, Zheng X, Cheng Z, Yuan Z, Zhang W (2018) Learning label-specific features for multi-label classification with missing labels. In: 2018 IEEE Fourth International Conference on Multimedia Big Data, pp. 1–5
9. Feng L, An B, He S (2019) Collaboration based multi-label learning. *Proceedings of the AAAI Conference on Artificial Intelligence* 33:3550–3557
10. Liu Y, Wen K, Gao Q, Gao X, Nie F (2018) Svm based multi-label learning with missing labels for image annotation. *Pattern Recogn* 78:307–317
11. Tahzeeb S, Hasan S (2022) A neural network-based multi-label classifier for protein function prediction. *Engineering, Technology & Applied Science Research* 12(1):7974–7981
12. Guan R, Wang X, Yang MQ, Zhang Y, Zhou F, Yang C, Liang Y (2018) Multi-label deep learning for gene function annotation in cancer pathways. *Sci Rep* 8(1):267
13. Gargiulo F, Silvestri S, Ciampi M, De Pietro G (2019) Deep neural network for hierarchical extreme multi-label text classification. *Appl Soft Comput* 79:125–138
14. Liu M, Luo Y, Tao D, Xu C, Wen Y (2015) Low-rank multi-view learning in matrix completion for multi-label image classification. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 29
15. Tan Q, Yu G, Wang J, Domeniconi C, Zhang X (2019) Individuality-and commonality-based multiview multilabel learning. *IEEE Transactions on Cybernetics* 51(3):1716–1727
16. Zhang C, Yu Z, Hu Q, Zhu P, Liu X, Wang X (2018) Latent semantic aware multi-view multi-label classification. *Proceedings of the AAAI Conference on Artificial Intelligence* 32:1
17. Liu X, Sun L, Feng S (2022) Incomplete multi-view partial multi-label learning. *Appl Intell* 52(3):3289–3302
18. Tan Q, Yu G, Domeniconi C, Wang J, Zhang ZJ (2018) Incomplete multi-view weak-label learning. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 2703–2709
19. Zhang J, Li C, Sun Z, Luo Z, Zhou C, Li S (2019) Towards a unified multi-source-based optimization framework for multi-label learning. *Appl Soft Comput* 76:425–435
20. Zhao D, Gao Q, Lu Y, Sun D, Cheng Y (2021) Consistency and diversity neural network multi-view multi-label learning. *Knowl-Based Syst* 218:106841
21. Huang J, Qu X, Li G, Qin F, Zheng X, Huang Q (2019) Multi-view multi-label learning with view-label-specific features. *IEEE Access* 7:100979–100992
22. Zhang M-L, Zhou Z-H (2007) MI-knn: A lazy learning approach to multi-label learning. *Pattern Recogn* 40(7):2038–2048
23. Fürnkranz J, Hüllermeier E, Loza Mencía E, Brinker K (2008) Multilabel classification via calibrated label ranking. *Mach Learn* 73:133–153
24. Huang J, Li G, Wang S, Zhang W, Huang Q (2015) Group sensitive classifier chains for multi-label classification. In: 2015 IEEE International Conference on Multimedia and Expo, pp. 1–6
25. Read J, Pfahringer B, Holmes G, Frank E (2011) Classifier chains for multi-label classification. *Mach Learn* 85:333–359
26. Zhu Y, Kwok JT, Zhou Z-H (2017) Multi-label learning with global and local label correlation. *IEEE Trans Knowl Data Eng* 30(6):1081–1094
27. Che X, Chen D, Mi J (2020) A novel approach for learning label correlation with application to feature selection of multi-label data. *Inf Sci* 512:795–812
28. Zhao D, Gao Q, Lu Y, Sun D (2022) Non-aligned multi-view multi-label classification via learning view-specific labels. *IEEE Transactions on Multimedia* 1–13
29. Zhu P, Hu Q, Hu Q, Zhang C, Feng Z (2018) Multi-view label embedding. *Pattern Recogn* 84:126–135
30. Ren W, Zhang L, Jiang B, Wang Z, Guo G, Liu G (2017) Robust mapping learning for multi-view multi-label classification with missing labels. In: *Knowledge Science, Engineering and Management*, pp. 543–551
31. Houthuys L (2017) Suykens JA (2017) Unpaired multi-view kernel spectral clustering. *IEEE Symposium Series on Computational Intelligence* 85:1–7
32. Li X, Chen S (2021) A concise yet effective model for non-aligned incomplete multi-view and missing multi-label learning. *IEEE Trans Pattern Anal Mach Intell* 44(10):5918–5932

33. Zhao D, Gao Q, Lu Y, Sun D (2022) Learning view-specific labels and label-feature dependence maximization for multi-view multi-label classification. *Appl Soft Comput* 124:109071
34. Xu L, Wang Z, Shen Z, Wang Y, Chen E (2014) Learning low-rank label correlations for multi-label classification with missing labels. In: 2014 IEEE International Conference on Data Mining, pp. 1067–1072
35. Huang S-J, Zhou Z-H (2012) Multi-label learning by exploiting label correlations locally. *Proceedings of the AAAI Conference on Artificial Intelligence* 26:949–955
36. Tsoumakas G, Spyromitros-Xioufis E, Vilcek J, Vlahavas I (2011) Mulan: A java library for multi-label learning. *The Journal of Machine Learning Research* 12:2411–2414
37. Guillaumin M, Verbeek J, Schmid C (2010) Multimodal semi-supervised learning for image classification. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 902–909
38. Liu X, Yu G, Domeniconi C, Wang J, Ren Y, Guo M (2019) Ranking-based deep cross-modal hashing. *Proceedings of the AAAI Conference on Artificial Intelligence* 33:4400–4407
39. Huang J, Qin F, Zheng X, Cheng Z, Yuan Z, Zhang W, Huang Q (2019) Improving multi-label classification with missing labels by learning label-specific features. *Inf Sci* 492:124–146
40. Liu C, Wen J, Luo X, Huang C, Wu Z, Xu Y (2023) Dicnet: Deep instance-level contrastive network for double incomplete multi-view multi-label classification. *Proceedings of the AAAI Conference on Artificial Intelligence* 37:8807–8815
41. Zhang M-L, Zhou Z-H (2013) A review on multi-label learning algorithms. *IEEE Trans Knowl Data Eng* 26(8):1819–1837
42. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research* 7:1–30

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.