Applied Soft Computing Journal 102 (2021) 107120

Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

Two-step multi-view and multi-label learning with missing label via subspace learning

Dawei Zhao [a,b], Qingwei Gao [a,b,*], Yixiang Lu [b], Dong Sun [b]

[a] School of Computer Science and Technology, Anhui University, Hefei 230601, PR China
[b] School of Electrical Engineering and Automation, Anhui University, Hefei 230601, PR China

ARTICLE INFO

Article history:
Received 29 January 2020
Received in revised form 8 December 2020
Accepted 17 January 2021
Available online 22 January 2021

Keywords:
Multi-view and multi-label learning
Subspace learning
Missing label
Matrix completion
Kernel extreme learning machine

ABSTRACT

In multi-view and multi-label learning, each example can be represented by multiple data view features and annotated with a set of discrete non-exclusive labels. Missing label learning is an important branch of multi-label learning, which can handle incomplete labels with annotations. Previous work on multi-label learning with missing labels mainly considered data in a single view representation. Based on intuitive understanding, we propose a **T**wo-step **M**ulti-view and **M**ulti-label **M**issing **L**abel learning optimization solution(TM3L). The first step is to solve the multi-view learning problem by finding the data representation of the common low-dimensional space of all views through subspace learning. While fully considering the complementary information between multiple views, the different degrees of contribution combined with different views are weighted differently. The second step is to solve the multi-label missing label learning problem by using the label matrix completion method in combination with the kernel extreme learning machine classifier. The kernel extreme learning machine can effectively enhance the robustness of the algorithm to missing labels. The experimental results and analysis on multiple benchmark multi-view and multi-label data sets verify the effectiveness of TM3L compared with the state-of-the-art solutions.

1. Introduction

With the rapid development of Internet technology and digital devices in recent years, web users can easily upload their multimedia data, such as photos and videos [1]. This data is generated in large quantities daily and most of the data is unlabeled. The annotation of diverse information is very helpful for data management and search, which has drawn an increasing research interest. Similarly, as data collection and feature extraction methods become more and more diversified. The form of a single view description problem cannot effectively solve the problem of diverse data. In the real world, there are often multiple views descriptions of instance. For example, in image analysis, images of natural scenes can usually be represented by their visual features (such as RGB and HSV color histograms, globe feature(Gist), and scale-invariant feature transform(SIFT)), and they can also be annotated with labels. In video analysis, video data has both images, audio, and text [2,3]. At present, it is difficult to describe something in detail from a single view. Multi-view learning has gradually become a hot research content and has attracted widespread attention in machine learning and various application fields [4–7].

In addition, multi-label learning is an essential framework for dealing with a single example and multiple labels simultaneously [8,9]. One of its core challenges is how to use the correlation between class labels to reconstruct an effective learning model that can predict the set of unknown instance labels. According to the types of label correlations used, the existing multi-label methods can generally be divided into three categories. First-order strategies: each tag has its unique attributes, and the relationship between tags is ignored. But the results are often suboptimal, for example, BR [10] and ML-*k*NN [11]; Second-order strategy: consider the correlation between paired labels, such as CLR [12] and MLRL [13], and the real-world application label relationship is often more complicated; High-order strategy: mining the correlation between all category labels, and can better reflect the multi-faceted relationship between real-world objects, such as MLSF [14] and MLMF [15]. To solve the more complex classification problem of natural data, a multi-view and multi-label learning framework is proposed.

The existing multi-label learning perspective-based multi-view and multi-label learning algorithm methods can be intuitively divided into two simple ways [16]. The first category: directly connecting multi-view data into single-view data, is

* Corresponding author at: School of Computer Science and Technology, Anhui University, Hefei 230601, PR China.
E-mail address: qingweigao@ahu.edu.cn (Q. Gao).

https://doi.org/10.1016/j.asoc.2021.107120
1568-4946/

solved using traditional multi-label learning methods. The problem is that it will ignore the unique physical interpretation between features, and it will cause over-fitting because of too high dimensions. The second category: a multi-label classifier is constructed for each multi-view data, and the results of each view are combined to obtain the final predictive label [17]. The complementarity between different views may be ignored. The most common way is to find a method to deal with multi-view and multi-label problems from multi-view learning models, which can be divided into the following categories:

1. Multi-view and multi-label methods combined with Co-training [18]: Maximize consensus by alternating training on two different views of unlabeled data.
2. Multi-view and multi-label method combined with multi-kernel learning [19]: The way to improve learning performance is to take advantage of the kernels that naturally correspond to each other in different views, and combine the kernels either linear or non-linear way.
3. Multi-view and multi-label method combined with subspace learning [1,2,20–22]: By assuming that the input view is generated from this potential subspace, observe the potential subspace shared by multiple views.

Furthermore, in previous multi-label learning studies, researchers previously assumed that all class labels for each sample were known as training data. In practical applications, the acquisition of class labels basically depends on manual annotation, which requires a lot of human and material resources. Manual annotation is difficult to fully annotate all labels, usually only partial labels can be observed. Annotating unlabeled samples directly not only wastes a lot of human resources, more important but also requires expert experience. Therefore, improving the performance of the classifier with a small number of labeled samples or defects is one of the main problems facing multi-view and multi-label learning. Based on the above studies, considering that subspace learning can solve multi-view problems well, and combined with the advantages of existing multi-label learning. We propose a separate two-step algorithm to solve the problem of missing multi-view multi-label data. A method similar to lrMMC [20] uses a two-step separate learning scheme. The first step uses multi-view subspace learning to extract multi-view shared subspace features. In the second step, a multi-label classification method of kernel extreme learning machine with label correlation is added to predict the unknown instance label set. The contributions of the methods in this paper are:

- A separate two-step solution is proposed to solve the problem of the missing label under multi-view and multi-label. Learning the shared subspace and subsequent label prediction in two separate steps. First, it can get shared subspaces information from different views. Secondly, in the multi-label learning of the kernel extreme learning machine, the correlation between labels is added to solve the problem of the missing label.
- In the process of subspace learning mapping, the Hilbert–Schmidt Independence Criterion is used to maintain the consistency of multi-view latent space further. At the same time, we consider the different contributions of different views to weight each view.
- TM3L is a two-step iterative optimization model that combines the advantages of multi-view subspace learning and multi-label learning. A large number of empirical results on the benchmark data set prove that TM3L and some related and competitive methods (such as LSML [16], ICM2L [21], iMvWL [23] and WcML [24]) have certain advantages.

The remainder of this paper is organized as follows. Section 2 reviews previous work on multi-view and multi-label learning, and missing label learning. Section 3 presents the details of the proposed TM3L method. Comparative experimental results and analysis are shown in Section 4. Finally, we conclude the paper in Section 5.

2. Related work

2.1. Multi-view and multi-label learning

Due to the widespread existence of multi-view and multi-label data, multi-view and multi-label learning has become an active research area in many practical applications [1,25,26].

In the existing multi-view learning methods, there are already a large number of strategies for acquiring the internal feature structure of multiple views and transforming feature expressions. For examples, MDBP [27] projects multi-view data into a shared subspace through a view-specific bilinear projection, which retains the structure of a multivariate time series and incorporates supervised regularization learning discriminative features; MLRA [28] is a multi-view low-rank analysis method. It first obtains the internal structure of multi-view data by performing cross-view low-rank analysis, and secondly, estimates the outliers of each test sample to identify outliers.

Besides, there have been some attempts to analyze multi-view data with multi-view and multi-label learning methods. Such as Xing et al. [29] proposed a predictive reliability measure to select samples for sharing label information with other views in a co-training manner. LSA-MML [22] solves the multi-view and multi-label learning problem based on the premise that there is a common representation between different views and obtains undiscovered latent semantics through alignment between different views in the kernel space. CSMSC [30] is a multi-view subspace learning method that can jointly extract the consistency and specificity of heterogeneous features for subspace representation learning. Liu et al. [20] proposed a multi-view framework lrMMC based on matrix factorization. This framework first seeks a shared representation of multiple views and then performs classification based on matrices on the shared feature space. Furthermore, Zhu et al. [31] map each view to a shared space to eliminate noise and redundancy, while maintaining the sparse and manifold structure of the image data, respectively. The goal of LSA-MML and MVLE [32] is to use the Hilbert–Schmidt Independence Criterion during the mapping process to further maintain consensus on the multi-view potential space. MSFS [33] captures higher-level label concepts and the correlation among multiple labels by decomposing label space information into reduced potential label representations. Further, the visual similarity and relationship of different views are used to construct multiple local geometric structures. SIMM [2] jointly minimizes confusion adversarial loss and multi-label loss to utilize shared information from all views. SIMM shares subspace and view-specific information extraction that are used together for model induction. ICM2L [21] is an individual and commonality-based approach for explicitly exploring the personality and commonality information of multi-label and multi-view data in a unified model.

It can be seen that subspace learning(SL) has gradually become one of the essential methods to solve the multi-view and multi-label learning problem. SL not only effectively solves the problem of dimensional disasters in multi-view learning, but also can mine common and private information from different perspectives.

2.2. Missing label learning

In practical applications, the way to obtain labels is often manually annotated, which may result in that when the label set is large, we can only observe a subset of labels. Since the incompleteness of the label set has a significant impact on the effect of multi-label learning, many scholars have proposed methods to mitigate performance degradation. For example, learning the label matrix as a preprocessing step before multi-label learning for classification can restore the integrity of the training data, such as [34–36]. The recovered label matrix may be suboptimal, which independently processes incomplete training data and predicted invisible category labels. Besides, some work combines label matrix recovery and multi-label classifier construction for joint learning. Usually, the number and location of lost labels are unknown in advance. For example, ML-LEML [37] handles missing label classification and multi-label learning under a unified framework, simultaneously. ML-LEML assumes that the label sets sharing the same cluster are highly correlated with each other, while the label sets of different groups are loosely connected. ML-LEML uses low-rank and sparse attributes to estimate defective labels and extracts potential relationships between features and labels in a low-dimensional shared subspace. The MLMF method realizes not only joint learning of independent binary classification but also considers joint learning of multi-label classification and label correlation. Glocal [38] solves the multi-label learning problem of defective labels by modeling global and local label correlations, learning potential class label representations, and optimizing label manifold regularization. MLR-GL [39] adds a low-rank structure to the predictions of all instances from the same label and adds the maximum separation structure to instances from different labels. ALSM [40] learns the intermediate feature space of labeled and unlabeled training samples through low-rank matrix restoration and uses an adaptive semi-supervised learning strategy to train a multi-label classifier. LSML performs joint learning of the recovery of the defective label set and label-specific features to achieve multi-label classification with missing labels. Another label recovery method is Maxide [36], which uses two-sided information matrices to accelerate matrix completion. Maxide assumes that the target and edge information matrices have the same potential information. Based on these existing missing label algorithms, some scholars have applied it to multi-view and multi-label learning. McWL is a multi-view weak label learning method based on matrix completion, which can model multi-feature fusion and matrix-completed prediction functions simultaneously. McWL describes the relationships between instances collected from different views by graphs and then uses kernel object alignment techniques to combine these graphs into a composite graph. iMvWL jointly solves the problems of incomplete views and missing labels in multi-view and multi-label learning. iMvWL can learn predictive labels and share subspaces simultaneously from incomplete views of weak labels, label correlations, and subspaces.

In multi-view and multi-label learning, the following challenges are faced:

1. Most multi-view subspace learning transforms multi-view learning problems into shared subspace learning problems. How to make full and effective use of shared and private information among views to improve the performance of the algorithm becomes the key.
2. In multi-label learning, the correlation among labels is often considered and utilized. Current research has fully confirmed that applying tag correlation can significantly improve the performance of multi-label classification learning. In the face of missing labels, how to effectively mine the relationship among labels and improve the classification performance of the algorithm significantly is a key issue for missing label learning.

Because of these problems, this paper proposes a two-step multi-view subspace learning to perform a multi-label classification of missing labels. Similar work has the lrMMC method and CSMSC method, and both are a two-step solution. The difference is that TM3L pays more attention to the ambiguity of private space and the effectiveness of classification.

3. Proposed approach

The learning framework of our proposed method TM3L is shown in Fig. 1. Our method framework is mainly composed of three parts. The first part is to learn the data representation of shared space and private space among multi-view. In the second part, determine the contribution weights of different views. The third part, multi-label classification learning under missing labels, gives the classification results.

Let $\boldsymbol{D} = \left\{\boldsymbol{X}_1^v, \boldsymbol{X}_2^v, \ldots, \boldsymbol{X}_n^v\right\}_{v=1}^m \in \mathbb{R}^{d_v \times n}$, $1 \le v \le m$, where n is the number of samples in a data set and m is the number of views, and d_v is the sample dimension of the vth view. $\boldsymbol{X}_n^v$ represents the feature space of the nth sample under the vth view. $\boldsymbol{Y} \in \{0, 1\}^{n \times l}$ indicates the label matrix, where $\boldsymbol{Y}_{ij} = 1$ indicates that the ith sample has the jth label. $\boldsymbol{Y}_{ij} = 0$ indicates that the jth label of the ith sample does not provide any information, and l indicates the number of class labels. Our goal is to predict the labels of unknown instances through feature space $\boldsymbol{D}$ and missing label space $\boldsymbol{Y}$ of known examples.

3.1. Multi-label subspace learning model

$$\min_{\boldsymbol{Q}^v, \boldsymbol{Z}} \frac{1}{mn} \sum_{i=1}^{n} \sum_{v=1}^{m} \left(\left\| \boldsymbol{X}_i^v - \boldsymbol{Q}^v \boldsymbol{Z}_i \right\|_F^2 + \frac{mC}{2} \left\| \boldsymbol{Z}_i \right\|_F^2 \right) \tag{1}$$

where $\boldsymbol{Q}^v \in \mathbb{R}^{d_v \times r}$ is the private information matrix corresponding to the vth view; $\boldsymbol{Z} \in \mathbb{R}^{r \times n}$ represents a multi-view shared information matrix, which encodes supplementary information from different views; r is the low-order matrix dimension; C is a non-negative constant, which is a trade-off between generating errors and penalizing regular terms in private low-dimensional spaces. $\|\cdot\|_F$ is denoted as Frobenius norm, and its simplicity and wide application are the reasons why we use it.

Furthermore, to enhance the diversity of private subspaces between different views, we approximate the quantification of diversity based on the dependencies between different private spaces. Considering that the smaller the correlation coefficient between the data matrices, the greater their diversity relationship, because the correlation between different private spaces is lower. Furthermore, we added the following regular penalty terms to punish the basic independence of different private spaces:

$$\min_{\boldsymbol{Q}^v, \boldsymbol{Z}} \frac{1}{mn} \sum_{i=1}^{n} \sum_{v=1}^{m} \left(\left\| \boldsymbol{X}_i^v - \boldsymbol{Q}^v \boldsymbol{Z}_i \right\|_F^2 + \frac{mC}{2} \left\| \boldsymbol{Z}_i \right\|_F^2 \right) + \frac{\lambda}{m} \boldsymbol{\Phi} \left(\left\{ \boldsymbol{Q}^v \right\}_{v=1}^m \right) \tag{2}$$

Various measurement methods can be used to assess the dependencies between variables. Here we use the Hilbert–Schmidt Independence Criterion (HSIC) [32,41] to constrain the consistence across different views because it is simple and has a solid theoretical foundation and the ability to measure linearity and non-linearity between variables. HSIC calculates the square norm

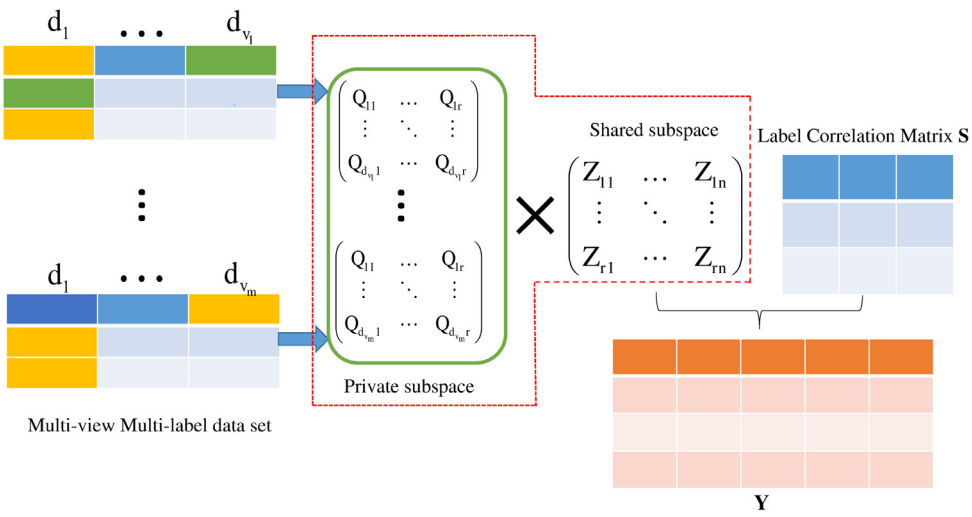


Fig. 1. The learning framework of the proposed method TM3L.

of the cross-covariance on $\boldsymbol{Q}^v$ and $\boldsymbol{Q}^h$ in Hilbert space to estimate the correlation. The empirical HSIC does not need to calculate the joint distribution of $\boldsymbol{Q}^v$ and $\boldsymbol{Q}^h$ explicitly. The HSIC in the method in this paper can be regarded as a punishment for the inconsistency of different views based on the basic similarity diagram, which can be expressed as:

$$HSIC\left(\boldsymbol{Q}^v, \boldsymbol{Q}^h\right) = (r-1)^{-2} tr\left(\boldsymbol{K}^v \boldsymbol{H} \boldsymbol{K}^h \boldsymbol{H}\right) \tag{3}$$

where $\boldsymbol{K}^v, \boldsymbol{K}^h, \boldsymbol{H} \in \mathbb{R}^{r\times r}$, $\boldsymbol{K}^v$ and $\boldsymbol{K}^h$ were used to measure kernel-induced similarity between $\boldsymbol{Q}^v$ and $\boldsymbol{Q}^h$, respectively. $\boldsymbol{H} = \delta_{ij} - 1/r$, $\delta_{ij} = 1$ if $i = j$, $\delta_{ij} = 0$ otherwise. In this paper, we use the inner matrix product as: $\boldsymbol{K}^v = (\boldsymbol{Q}^v)^{\mathrm{T}} \boldsymbol{Q}^v, \forall v = \{1, 2, \ldots, m\}$. Then we maximize the overall HSIC of the v perspective matrices to reduce the redundancy between them and specify $\boldsymbol{\Phi}(\boldsymbol{Q}^v)$ as follows:

$$\begin{aligned}
\boldsymbol{\Phi}\left(\left\{\boldsymbol{Q}^v\right\}_{v=1}^m\right) &= -\sum_{v=1, v\neq h}^{m} HSIC\left(\boldsymbol{Q}^v, \boldsymbol{Q}^h\right) \\
&= -\sum_{v=1, v\neq h}^{m} (r-1)^{-2} tr\left(\boldsymbol{Q}^v \boldsymbol{H} \boldsymbol{K}^h \boldsymbol{H} \left(\boldsymbol{Q}^h\right)^{\mathrm{T}}\right) \\
&= \sum_{v=1}^{m} tr\left(\boldsymbol{Q}^v \tilde{\boldsymbol{K}}^v \left(\boldsymbol{Q}^v\right)^{\mathrm{T}}\right)
\end{aligned} \tag{4}$$

where $\tilde{\boldsymbol{K}}^v = -(r-1)^{-2} \sum_{h=1, h\neq v}^{m} \boldsymbol{H} \boldsymbol{K}^h \boldsymbol{H}$.

In addition, we also try to consider the problem from the perspective that all data views may be useful for multi-label learning tasks, but with different contributions. We try to learn the view weights of all views, where $\boldsymbol{\theta}^v$ represents the degree of contribution of the vth data view. Then express the optimization problem as:

$$\begin{aligned}
&\min_{\boldsymbol{Q}^v, \boldsymbol{Z}, \boldsymbol{\theta}^v} \frac{1}{mn} \sum_{i=1}^{n} \sum_{v=1}^{m} \boldsymbol{\theta}^v \left(\left\| \boldsymbol{X}_i^v - \boldsymbol{Q}^v \boldsymbol{Z}_i \right\|_F^2 + \frac{mC}{2} \left\| \boldsymbol{Z}_i \right\|_F^2 \right) + \\
&\frac{\lambda}{m} tr\left(\boldsymbol{Q}^v \tilde{\boldsymbol{K}}^v \left(\boldsymbol{Q}^v\right)^{\mathrm{T}}\right) + \frac{\alpha}{2n} \left\| \boldsymbol{\theta} \right\|_2^2 \\
&\text{s.t.} \boldsymbol{\theta}^v \geq 0, \sum_{v=1}^{m} \boldsymbol{\theta}^v = 1
\end{aligned} \tag{5}$$

3.2. Missing label multi-label learning

In this section, we model the shared subspace $\boldsymbol{Z}$ and the label space $\boldsymbol{Y}$ to predict the label set of unknown samples. Effective use of label correlation information can undoubtedly complement the missing information in the original label space [42]. We use the label correlations among the missing labels to estimate the likelihood score, which can be expressed as:

$$\min_{\boldsymbol{S}, \boldsymbol{W}} \frac{1}{2} \left\| \boldsymbol{Z}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{S} - \boldsymbol{Y} \right\|_F^2 \tag{6}$$

Generally, label correlation matrix $\boldsymbol{S}$ is estimated by the known label matrix $\boldsymbol{Y}$, but in the problem of missing labels, we cannot directly obtain the label correlation matrix through prior knowledge. Besides, because the labeled samples are not sufficient, we advocate learning it using the features of the training data and the known labels [9,42]. Therefore, we need to obtain $\boldsymbol{S}$ through learning.

Furthermore, we need to consider such a widespread situation where local characteristics [43] similarity among labels, so the label correlations matrix has a low-rank trend. For this reason, many scholars [38,44] solve this problem by assuming that the label correlation matrix has a local structure and set $\boldsymbol{S}$ as a low-rank matrix. In other words, there are multiple label subsets in a label set, and these label subsets have complex correlations and are closely related to each other and are independent of the remaining label subsets. This local structure in practical applications usually means that $\boldsymbol{S}$ is a low-rank matrix structure [20,22,23]. To obtain the correlation of local labels, we add a constraint of the low-rank matrix on $\boldsymbol{S}$. We rewrite Eq. (6) to make it more suitable for missing label problems:

$$\min_{\boldsymbol{S}, \boldsymbol{W}} \frac{1}{2} \left\| \boldsymbol{Z}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{S} - \boldsymbol{Y} \right\|_F^2 + \beta rank\left(\boldsymbol{S}\right) \tag{7}$$

where β is a trade-off parameter that balances the relative importance of low-rank constraints to matrix $\boldsymbol{S}$. By adding sorting terms, our model can obtain local label correlations between defect signatures, making it more suitable for practical applications. It is worth noting that the work in lrMMC assumes that the association among labels is also low-rank, but the usage is different. lrMMC multiplies the low-rank label correlation matrix with $\boldsymbol{Y}$ to complete the missing labels. Similarly, LSML performs the same operation, but LSML does not consider local label correlation. Instead, we multiply the low-rank label correlation matrix with the predicted likelihood label vector, which is similar to iMvWL.

iMvWL considers that the estimated value of label correlations may not necessarily be reliable in practice. Therefore, using the low-rank correlation matrix to multiply the predicted likelihood label vector is less affected by the method of multiplying the label matrix and the label correlation matrix. The low-rank matrix minimization problem is an NP-hard problem. In this paper, the nuclear norm $\|\cdot\|_*$ is used as a convex approximation of the rank function. We rewrite Eq. (7) as:

$$\min_{\boldsymbol{S},\boldsymbol{W}} \frac{1}{2}\left\|\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{S}-\boldsymbol{Y}\right\|_F^2+\beta\|\boldsymbol{S}\|_* \tag{8}$$

In summary, we decompose the multi-view and multi-label missing label problem into a two-step learning problem for optimal solution. This two-step learning method can well combine the advantages of the two learning frameworks. Next, we need to solve Eq. (5) and Eq. (8) separately.

3.3. Optimized solution for the first step

First, we consider the problem of optimizing Eq. (5). The objective function in Eq. (5) involves $\boldsymbol{Q}$, $\boldsymbol{Z}$, and $\boldsymbol{\theta}$. In general, we use an iterative optimization technique to optimize these three parameters separately. This technique optimizes the objective function by iteratively solving a variable while fixing other variables, and the entire process is performed alternately.

(1) Update $\boldsymbol{Q}$ With Fix $\boldsymbol{Z}$ and $\boldsymbol{\theta}$. Eq. (5) is a convex optimization problem, and all views are considered to be equally relevant during initialization. Our optimization goal can be rewritten as:

$$f\left(\boldsymbol{Q}^v\right)=\min_{\boldsymbol{Z}}\frac{1}{mn}\sum_{v=1}^{m}\left(\left\|\boldsymbol{X}^v-\boldsymbol{Q}^v\boldsymbol{Z}\right\|_F^2\right)+\frac{\lambda}{m}tr\left(\boldsymbol{Q}^v\tilde{\boldsymbol{K}}^v\left(\boldsymbol{Q}^v\right)^{\mathrm{T}}\right) \tag{9}$$

For each view v, we obtain the following equation by taking the derivative of Eq. (9) for $\boldsymbol{Q}^v$:

$$\frac{\partial f\left(\boldsymbol{Q}^v\right)}{\partial\left(\boldsymbol{Q}^v\right)}=\frac{1}{mn}\sum_{v=1}^{m}\boldsymbol{\theta}^v\left(\boldsymbol{Q}^v\boldsymbol{Z}\boldsymbol{Z}^{\mathrm{T}}-\boldsymbol{X}\boldsymbol{Z}^{\mathrm{T}}\right)+\frac{\lambda}{m}\boldsymbol{Q}^v\tilde{\boldsymbol{K}}^v \tag{10}$$

Using the Karush–Kuhn–Tucker(KKT) condition [45], we can derive the following updating rule:

$$\left(\boldsymbol{Q}^v\right)_{ij}\leftarrow\left(\boldsymbol{Q}^v\right)_{ij}\frac{\sum_{v=1}^{m}\boldsymbol{\theta}^v\boldsymbol{X}\boldsymbol{Z}^{\mathrm{T}}}{\sum_{v=1}^{m}\boldsymbol{\theta}^v\boldsymbol{Q}^v\boldsymbol{Z}\boldsymbol{Z}^{\mathrm{T}}+n\lambda\boldsymbol{Q}^v\tilde{\boldsymbol{K}}^v} \tag{11}$$

Besides, to avoid the value of $\boldsymbol{Z}$ can be arbitrarily large, we normalize the value of $\boldsymbol{Q}^v$ by $\frac{\boldsymbol{Q}^v}{\|\boldsymbol{Q}^v\|_F}$ to prevent the appearance of trivial solutions.

(2) Update $\boldsymbol{Z}$ With Fix $\boldsymbol{Q}$ and $\boldsymbol{\theta}$. We obtain the following equation by taking the derivative of Eq. (5) for $\boldsymbol{Z}$ to zero:

$$\left(Cm\boldsymbol{I}+\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{Q}\right)\boldsymbol{Z}-\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{X}=0 \tag{12}$$

According to the closed-form solution of Eq. (12), the update rule of the shared subspace $\boldsymbol{Z}$ can be expressed as:

$$\boldsymbol{Z}=\left(Cm\boldsymbol{I}+\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{Q}\right)^{-1}\times\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{X} \tag{13}$$

(3) Update $\boldsymbol{\theta}$ With Fix $\boldsymbol{Q}$ and $\boldsymbol{Z}$. When $\boldsymbol{Z}$ and $\boldsymbol{Q}$ are fixed, the Eq. (5) optimization target can be rewritten as:

$$\begin{aligned}&\min_{\boldsymbol{\theta}^v}\frac{1}{mn}\sum_{i=1}^{n}\sum_{v=1}^{m}\boldsymbol{\theta}^v\left(\left\|\boldsymbol{X}_i^v-\boldsymbol{Q}^v\boldsymbol{Z}_i\right\|_F^2+\frac{mC}{2}\|\boldsymbol{Z}_i\|_F^2\right)+\frac{\alpha}{2n}\|\boldsymbol{\theta}\|_2^2\\&\text{s.t.}\boldsymbol{\theta}^v\geq 0,\sum_{v=1}^{m}\boldsymbol{\theta}^v=1\end{aligned} \tag{14}$$

Coordinate descent method is used to update $\boldsymbol{\theta}^v$. Especially, in each iteration, only two elements $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$ are selected for update, while the other elements are fixed. By using the Lagrangian multiplier method for Eq. (14) and considering it as a constraint, $\boldsymbol{\theta}$ is updated by the following rules:

$$\begin{cases}\boldsymbol{\theta}_i^*=0,\boldsymbol{\theta}_j^*=\boldsymbol{\theta}_i+\boldsymbol{\theta}_j,\ if\ \lambda m\left(\boldsymbol{\theta}_i+\boldsymbol{\theta}_j\right)+\left(\boldsymbol{u}_j-\boldsymbol{u}_i\right)\leq 0,\\ \boldsymbol{\theta}_j^*=0,\boldsymbol{\theta}_i^*=\boldsymbol{\theta}_i+\boldsymbol{\theta}_j,\ if\ \lambda m\left(\boldsymbol{\theta}_i+\boldsymbol{\theta}_j\right)+\left(\boldsymbol{u}_i-\boldsymbol{u}_j\right)\leq 0,\\ \boldsymbol{\theta}_i^*=\frac{\lambda m\left(\boldsymbol{\theta}_i+\boldsymbol{\theta}_j\right)+\left(\boldsymbol{u}_j-\boldsymbol{u}_i\right)}{2\lambda m},\boldsymbol{\theta}_j^*=\boldsymbol{\theta}_i+\boldsymbol{\theta}_j-\boldsymbol{\theta}_i^*,\\ otherwise\end{cases} \tag{15}$$

where $\boldsymbol{u}=[\boldsymbol{u}_1,\ldots,\boldsymbol{u}_m]^{\mathrm{T}}$, with each $\boldsymbol{u}_i=\left\|\boldsymbol{X}^{(i)}-\boldsymbol{Q}^{(i)*}\boldsymbol{Z}^*\right\|_F^2+\frac{mC}{2}\left\|\boldsymbol{Z}^*\right\|_F^2$.

The first step learning process of the TM3L method is outlined in Algorithm 1. The stopping criterion of the algorithm is the difference between the target values of two consecutive steps. Since each of the above subproblems is convex, the TM3L algorithm can guarantee convergence to the local optimum of Eq. (5).

Algorithm 1: Multi-view subspace learning based on matrix factorization

Input: Training data matrix: $\left\{\boldsymbol{X}^v\right\}_{v=1}^m$;
Trade-off parameters: C, α, and λ;
Dimensionality of the shared subspace: r;
Minimum convergence error: ϵ;
Number of iterations: t;
Output: Shared subspace matrix: $\boldsymbol{Z}$;

```
Randomly initialize Q^v, Z, θ^v, and S;
for j = 1, 2, ···, t do
    for v = 1, 2, ···, m do
        Update Q^v by Eq. (11);
        Normalize Q^v;
        Update θ^v by Eq. (15);
    Update Z by Eq. (13);
    if convergence then
        break;
```

3.4. Optimized solution for the second step

Next, we consider the optimization Eq. (8) problem. The objective function in Eq. (8) involves $\boldsymbol{W}$ and $\boldsymbol{S}$. Similarly, and we use an alternating optimization technique.

(1) Update $\boldsymbol{S}$ With Fixed $\boldsymbol{W}$. When $\boldsymbol{W}$ is fixed, Eq. (8) optimized for $\boldsymbol{S}$ can be re-expressed as:

$$\min_{\boldsymbol{S}}\frac{1}{2}\left\|\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{S}-\boldsymbol{Y}\right\|_F^2+\beta\|\boldsymbol{S}\|_* \tag{16}$$

Eq. (16) can be regarded as a matrix completion problem, and many algorithms have been proposed to solve this problem in the past few decades. In this paper, an efficient acceleration algorithm Maxide [36], which only needs to estimate an $l\times l$ matrix, is used to solve it. Similar methods such as iMvWL and lrMMC has been adopted.

(2) Update $\boldsymbol{W}$ With Fixed $\boldsymbol{S}$. When fixed $\boldsymbol{S}$, Eq. (8) is a convex optimization problem about the least-squares loss. The optimization objective can be rewritten as:

$$\min_{\boldsymbol{W}}\frac{1}{2}\left\|\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{S}-\boldsymbol{Y}\right\|_F^2 \tag{17}$$

Eq. (17) has many optimization methods. In this paper, we utilize extreme learning machines, a single hidden layer feedforward neural network (SLFN) algorithm to solve this problem. In the traditional neural network algorithm, more parameter settings are needed initially, and the optimal local problem will appear when solving the optimal solution, and the global optimal solution

cannot be obtained. The extreme learning machine (ELM) [46,47] is an efficient optimization learning algorithm. Initially, only the number of nodes in the hidden layer needs to be set, and the weights and deviations are randomly initialized to obtain the optimal global solution. Eq. (17) can be rewritten as:

$$\min_{\boldsymbol{W}} \|\boldsymbol{HWS} - \boldsymbol{Y}\|_{\mathrm{F}}^2 \tag{18}$$

where $\boldsymbol{H}$ represents the hidden layer output matrix:

$$\boldsymbol{H} = \begin{bmatrix} g\left(\boldsymbol{Z}_1^{\mathrm{T}}\right) \\ \vdots \\ g\left(\boldsymbol{Z}_n^{\mathrm{T}}\right) \end{bmatrix} = \begin{bmatrix} g_1\left(\boldsymbol{Z}_1^{\mathrm{T}}\right) & \cdots & g_l\left(\boldsymbol{Z}_1^{\mathrm{T}}\right) \\ \vdots & \vdots & \vdots \\ g_1\left(\boldsymbol{Z}_n^{\mathrm{T}}\right) & \cdots & g_l\left(\boldsymbol{Z}_n^{\mathrm{T}}\right) \end{bmatrix} \tag{19}$$

In Eq. (19), g_i is the activation function which can be expressed as:

$$g_i\left(\boldsymbol{Z}_j\right) = g\left(\boldsymbol{Z}_j^{\mathrm{T}} \cdot \boldsymbol{\omega}_i + \boldsymbol{b}_i\right) \tag{20}$$

where ω_i is the input weight, and b_i is the bias of the ith hidden neuron. From Eqs. (18) and (19), we know that the solution of least-squares can be expressed as:

$$\widehat{\boldsymbol{W}} = \boldsymbol{H}^{\dagger}\boldsymbol{YS}^{-1} \tag{21}$$

where is $\boldsymbol{H}^{\dagger}$ the Moore–Penrose generalized inverse matrix of $\boldsymbol{H}$.

$$\text{s.t.}\boldsymbol{H}^{\dagger} = \begin{cases} \left(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}\right)^{-1}\boldsymbol{H}^{T}, \boldsymbol{H}^{T}\boldsymbol{H} \textit{ is a nonsingular matrix} \\ \boldsymbol{H}\left(\boldsymbol{HH}^{\mathrm{T}}\right)^{-1}, \boldsymbol{HH}^{T} \textit{ is a nonsingular matrix} \end{cases} \tag{22}$$

According to the Ridge Regression Theory, adding the regularization term P to the diagonal of $\boldsymbol{HH}^{\mathrm{T}}$ or $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ can improve the stability and generalization ability of the algorithm. The minimization goal of formula (18) is:

$$\begin{aligned} &\min_{\boldsymbol{W}} g\left(\boldsymbol{Z}\right) = \|\boldsymbol{W}\|^2 + P\sum_{i=1}^{n}\left\|\boldsymbol{\xi}_i\right\|^2; \\ &\text{s.t.}\boldsymbol{\xi}_i = \boldsymbol{Y}_i - g(\boldsymbol{Z}_i), i = 1, 2, \ldots, n \end{aligned} \tag{23}$$

According to the KKT condition (Karush–Kuhn–Tucker, KKT), the hidden output weight $\boldsymbol{W}$ is expressed as:

$$\boldsymbol{W} = \boldsymbol{H}^{\mathrm{T}}(\frac{\boldsymbol{I}}{P} + \boldsymbol{HH}^{\mathrm{T}})^{-1}\boldsymbol{YS}^{-1} \tag{24}$$

According to Eqs. (19) and (24), the final label set can be predicted:

$$\widehat{\boldsymbol{Y}} = \boldsymbol{H}\widehat{\boldsymbol{W}}\boldsymbol{S} \tag{25}$$

In the traditional ELM method, the calculation result is easily affected by the random set value. To this end, a kernel matrix is introduced to solve this problem:

$$\begin{aligned} &\boldsymbol{\Omega}_{\mathrm{ELM}} = \boldsymbol{HH}^{\mathrm{T}} : \boldsymbol{\Omega}_{\mathrm{ELM}(i,j)} = \boldsymbol{K}(z_i, z_j); \\ &\boldsymbol{K}(z_i, z_j) = \exp(-\gamma\left\|z_i - z_j\right\|) \end{aligned} \tag{26}$$

Eq. (25) can be rewritten as:

$$\begin{aligned} \widehat{\boldsymbol{Y}} &= h(z)\boldsymbol{H}^{\mathrm{T}}(\frac{\boldsymbol{I}}{P} + \boldsymbol{HH}^{\mathrm{T}})^{-1}\boldsymbol{YS}^{-1} \\ &= \begin{bmatrix} \boldsymbol{K}(z,z_1) \\ \vdots \\ \boldsymbol{K}(z,z_n) \end{bmatrix}(\frac{\boldsymbol{I}}{P} + \boldsymbol{\Omega}_{\mathrm{ELM}})^{-1}\boldsymbol{YS}^{-1} \end{aligned} \tag{27}$$

The second step learning process of the TM3L method is outlined in Algorithm 2.

Algorithm 2: Two-step Multi-view and Multi-label Learning with Missing Label via Subspace Learning

Input: Training data matrix: $\left\{\boldsymbol{X}^{v}\right\}_{v=1}^{m}$;
Training label data set: $\boldsymbol{Y}$;
Trade-off parameters: C, α, λ, β, P, and σ;
Dimensionality of the shared subspace: r;
Output: Predicted likelihood score matrix: $\widehat{\boldsymbol{Y}}$;

```
Randomly initialize Q^v, Z, θ^v, and S;
while not convergence do
    Obtain the shared subspace matrix Z through Algorithm 1;
    Update S by Maxide;
    Update W by Eq. (24);
Return the predicted likelihood score matrix by Eq. (27).
```

3.5. Complexity analysis

The time complexity of TM3L is mainly composed of the following parts. The complexity of solving $\boldsymbol{Q}_v$ and $\boldsymbol{Z}$ in the first step are $\mathcal{O}\left(m\left(nr^2 + r^3 + nd_{\max}r\right)\right)$ and $\mathcal{O}\left(d_{\max}r^2 + r^3 + nd_{\max}r\right)$, respectively. $d_{\max}$ represents the largest dimensionality of the views. The complexity of solving $\boldsymbol{W}$ and $\boldsymbol{S}$ in the second step are $\mathcal{O}\left(n^3 + n^2l + nd_{\max}l\right)$ and $\mathcal{O}\left(ql\ln\left(l\right)\ln\left(n\right)\right)$, respectively, where q is the rank of $\boldsymbol{S}$. Sicne $n \gg l$ and $n \gg r$ the overall time complexity of TM3L is $\mathcal{O}\left(tmnd_{\max}r\right)$, where t is the number of iterations to converge.

4. Experimental content

4.1. Comparison algorithm

1. ICM2L[1]: A method to explicitly explore the individuality and commonality information of multi-view and multi-label data in a unified model. According to the parameters given in the paper, we recommend setting $\alpha = 0.6$, $\beta = 0.7$, and $k = 0.5d_{\min}$.
2. LSML[2]: The label-specific feature learning method for multi-label missing label classification, which jointly learning classification tasks and recovery of label matrices. All parameters of the proposed method are adjusted in $\left\{10^{-5}, 10^{-4}, \ldots, 10^{3}\right\}$.
3. McWL[3]: Multi-view weak label learning method based on matrix completion. McWL performs multi-view integration and MC-based classification optimization in a unified objective function. The parameters α, β and k are searched in $\left\{2^{-5}, 2^{-4}, \ldots, 2^{5}\right\}$, $\{0.1, 0.2, \ldots, 0.5\}$, and $\{1, 2, \ldots, 10\}$, respectively.
4. iMvWL[4]: Incomplete multi-view weak label learning. In experiments, complete view information is available. The parameters α and β are adjusted within $\left\{10^{-5}, 10^{-4}, \ldots, 10^{0}\right\}$.
5. TM3L: Two-step multi-view and multi-label missing label learning via subspace learning. The parameters α, C, and λ are adjusted within the range of $\left\{10^{-5}, 10^{-4}, \ldots, 10^{5}\right\}$. The parameter β is fixedly set to $10^{-}3$. The kernel extreme learning machine regularization coefficient P and the kernel parameter σ are both fixedly set to 1.

[1] code: http://mlda.swu.edu.cn/codes.php?name=ICM2L.
[2] code: http://www.escience.cn/people/huangjun/index.html.
[3] code: http://mlda.swu.edu.cn/codes.php?name=McWL.
[4] code: http://mlda.swu.edu.cn/codes.php?name=iMvWL.

Table 1
Data set description.

Views	Yeast	Pascal07	Corel5k	Espgame	Iaprtc12	Mirflickr
1	Genetic Expression (79)	DenseSift (1000)	DenseHue (100)	DenseHue (100)	DenseHue (100)	DenseHue (100)
2	Phylogenetic Profile (24)	HarrisSift (1000)	DenseSift (1000)	DenseSift (1000)	DenseSift (1000)	DenseSift (1000)
3	–	Gist(512)	Gist(512)	Gist(512)	Gist(512)	Gist(512)
4	–	HSV(4096)	HSV(4096)	HSV(4096)	HSV(4096)	HSV(4096)
5	–	RGB(4096)	–	Lab(4096)	Lab(4096)	Lab(4096)
6	–	Tags(804)	–	RGB(4096)	RGB(4096)	RGB(4096)
l	14	20	260	268	291	457
n	2417	9963	4999	20 770	19 627	25 000

Table 2
Summary of the Friedman Statistics F_F ($k = 5$, $N = 24$) and the critical value in terms of each evaluation metric (k: Comparing Algorithms; N: Data sets).

Metric	F_F	Critical Value($\rho = 0.05$)
Hamming Loss	55.0902	1.2452
Subset Accuracy	53.6445	
Average Precision	28.1703	
One Error	30.8975	
Ranking Loss	20.3508	
Coverage	13.6574	
AUC	8.6484	

Both McWL and iMvWL are designed to solve the multi-view and multi-label problem of missing labels directly. The difference between TM3L and iMvWL and McWL is the unified learning strategy adopted by iMvWL and McWL. But neither of them considers the dependency information among views and the difference in each view's contribution. ICM2L is designed to solve the problem of multi-view and multi-label learning. We use it to solve the problem of multi-view and multi-label of missing labels. The difference between TM3L and ICM2L is that ICM2L ignores the problem of different contributions from different views. The LSML method is not directly utilized to solve the multi-view multi-label with missing label learning problems. In this paper, we build a multi-label missing label learning model based on each view data. LSML predicts the final result by combining the output of m multi-view models with equal contribution weights.

For all comparison methods, we use a five-fold cross-validation method on the training set, and select the optimal parameter values from the range suggested by the original paper. For our method, the parameters we choose are α, C, and λ. To avoid errors caused by random effects, all experiments were independently repeated 10 times, and the mean and standard deviation were reported.

4.2. Data sets

In order to verify the effectiveness of the TM3L algorithm, we tested it on six multi-view and multi-label benchmark data sets with five-fold cross-validation, which can be downloaded from [48][5] and Mulan.[6] Details are summarized in Table 1.

4.3. Evaluation metrics

We use seven widely used multi-label evaluation metrics for performance comparison [49]: (1) Hamming Loss (HL); (2) Subset Accuracy (SA); (3) Average Precision (AP); (4) One Error (OE); (5) Ranking Loss (RL); (6) Coverage (CV) and (7) AUC. These metrics can be divided into two categories based on different types of reference: (1) example-based criteria; (2) label-based criteria. HL, SA, AP, OE, RL, and CV are example-based criteria, while AUC is a label-based criterion. Performance can be evaluated from the perspective of ranking and classification, where AP, OE, RL, CV, and AUC are ranking-based metrics, while HL and CV are example-based classification metrics. Formal definitions of these seven metrics can be found in the literature [49–51]. Among them, the larger the value of SA, AUC and AP, the better the performance, and the smaller the values of HL, OE, RL and CV, the better the performance.

4.4. Experimental results and analysis

All the experiments are implemented using Matlab 2016a on a standard Windows PC with an Intel 4.2-GHz CPU and 16-GB RAM. In order to create a missing label scenario, we randomly delete all positive labels used for training data according to the preset loss rate $w\%$, and for each instance, keep at least one positive class to avoid empty instances or labels. In this article $w\%$ is set to 0%, 40%, 60% and 90%. Tables A.3 to A.6 list the average results (mean $\pm$ standard deviation) of each comparison algorithm for each comparison index on 6 multi-view and multi-label benchmark data sets under various missing labels rates, the best results are shown in bold. Because McWL algorithm consumes a lot of memory during training, it will run out of memory on some data sets (OM means algorithm out of memory).

In addition, the Friedman test [52] was used to compare statistical performance between the comparison methods. Since the loss rate varies from 0% to 90%, there are 24 (4×6) points in total. Table 2 summarizes the Friedman statistic F_F and the critical difference (CD) corresponding to each evaluation criterion, where the uppermost row is the critical difference CD = 1.2452. As shown in Table 2, at the significance level $\rho = 0.05$, for each evaluation index, the null hypothesis that all comparison algorithms are equivalently executed is explicitly rejected.

The Nemenyi test [16] with $q_\alpha = 2.728$ at a significance level of 5% is used as a post-test. When the difference between the average ranking of the two comparison algorithms on all data sets is greater than the critical difference, it is considered that the two algorithms have significant differences. Otherwise, no significant difference is considered. Fig. 2 shows a comparison between each algorithm under different evaluation indicators. For algorithms with no significant difference, they are connected by colored solid lines. From the left to the right of each evaluation index submap, the performance of the algorithms decreases in order. Comprehensively reporting the experimental results, the following conclusions are drawn:

1. TM3L algorithm achieves the best performance on almost every evaluation metrics under all data sets. Fig. 2 clearly shows the advantages of our method in exploring multi-view and multi-label missing label data. In detail, Tables A.3 to A.6 show that when the missing rates are

[5] data sets: http://lear.inrialpes.fr/people/guillaumin/data.php.
[6] data sets: http://mulan.sourceforge.net/datasets-mlc.html.

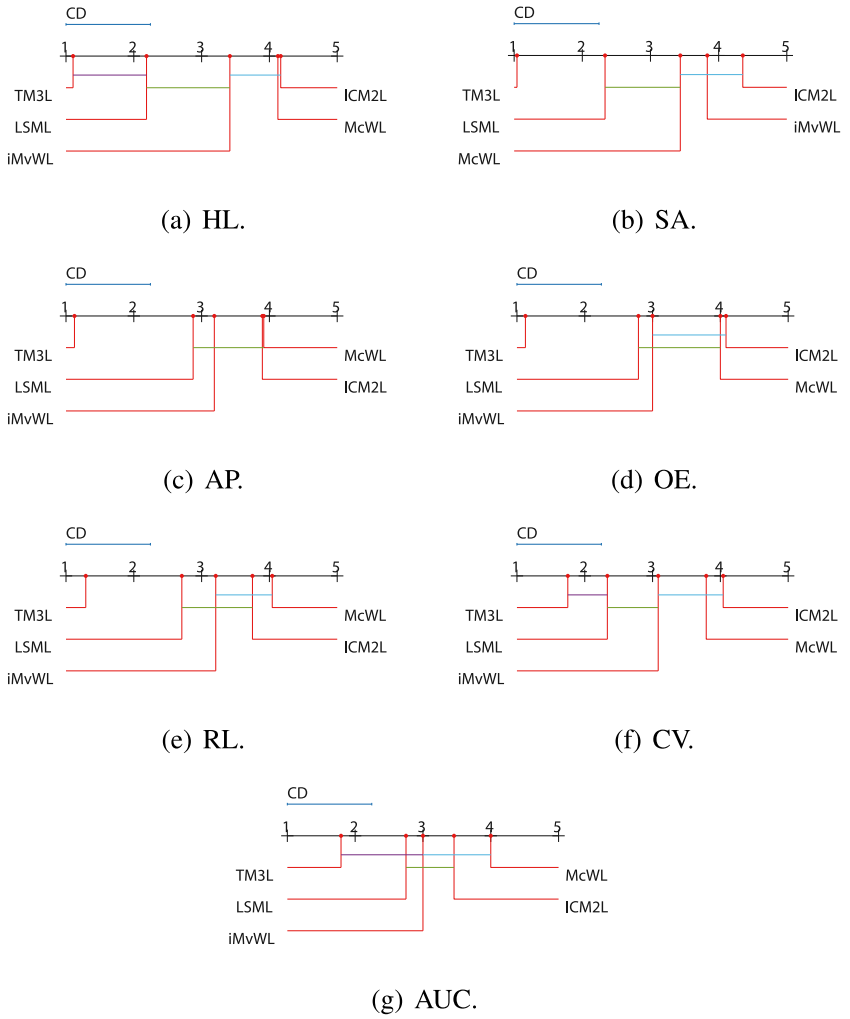


Fig. 2. The performance comparison of algorithms.

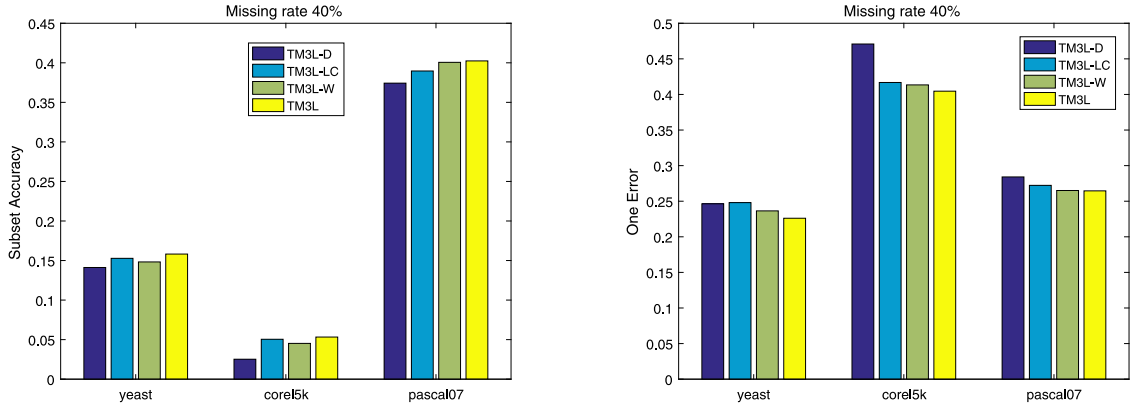
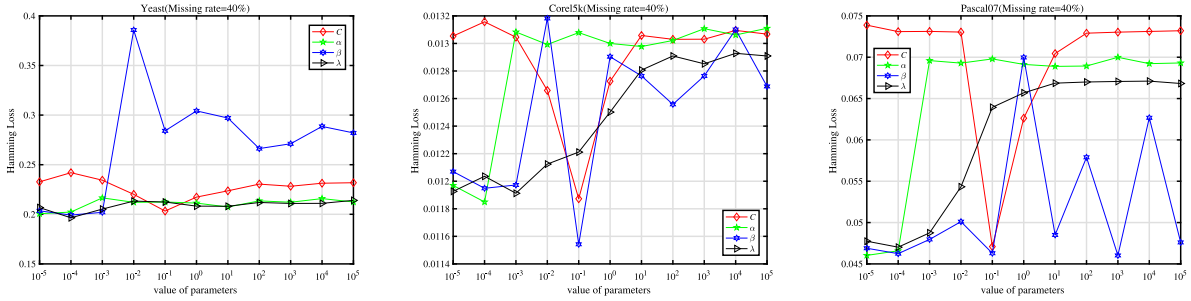
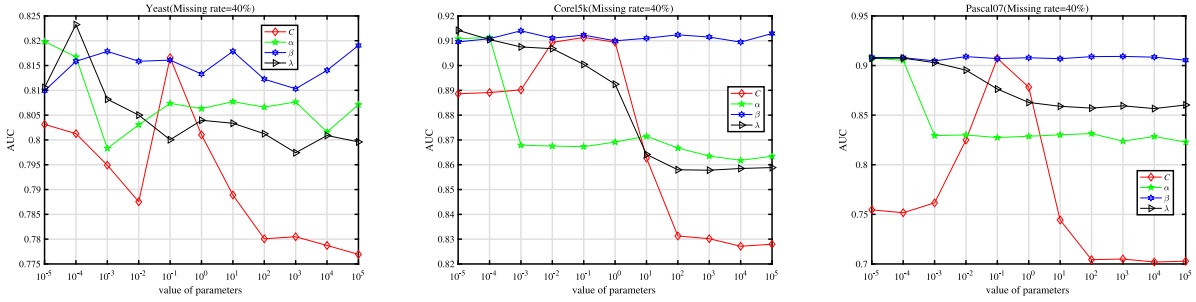


Fig. 3. The experimental results of TM3L and its variants on SA and OE with a missing labels rate of 40%.

0%, 40%, 60%, and 90%, TM3L is superior to other competitive methods in 88.1%, 83.3%, 71.4%, and 78.6%, respectively. Especially, the performance under the SA is usually much better than other algorithms. On the AUC, ICM2L and iMvWL can achieve better results on some data sets compared to TM3L. These experimental results also indicate that the results obtained by this two-step solution may be suboptimal.

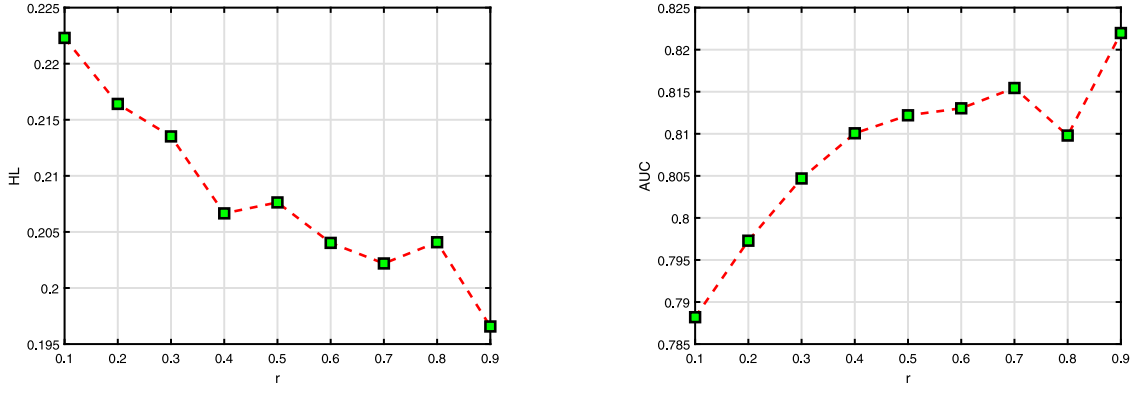


(a) HL.



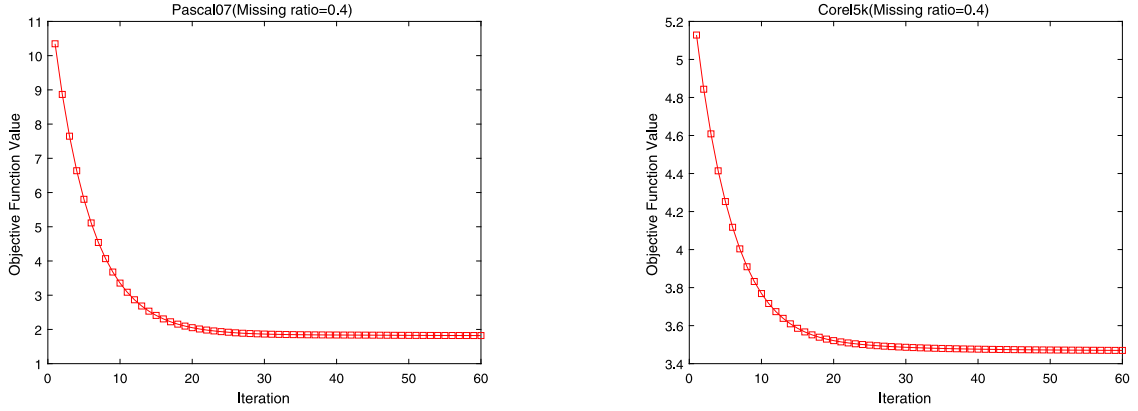
(b) AUC.

Fig. 4. Parameter sensitivity analysis of TM3L algorithm over *Yeast*, *Corel5k*, and *Pascal*07 data sets.



(a) HL. (b) AUC.

Fig. 5. Experimental results of TM3L with different parameters r.



(a) Pascal07. (b) Corel5k.

Fig. 6. Convergence trend analysis.

Table A.3
Experimental results of 5 algorithms on 6 data sets with the missing label rate of 0%.

Data set	Metric	ICM2L	LSML	McWL	iMvWL	TM3L
Yeast	HL	0.2778 ± 0.0079	0.2608 ± 0.0082	0.2246 ± 0.0101	0.2689 ± 0.0049	**0.1927 ± 0.0061**
	SA	0.0041 ± 0.0021	0.0811 ± 0.0105	0.0186 ± 0.0021	0.0075 ± 0.0038	**0.1822 ± 0.0094**
	AP	0.7078 ± 0.0137	0.6102 ± 0.0082	0.7644 ± 0.0091	0.7038 ± 0.0105	**0.7767 ± 0.0176**
	OE	0.2350 ± 0.0238	0.3583 ± 0.0192	0.2236 ± 0.0041	0.2915 ± 0.0197	**0.2120 ± 0.0354**
	RL	0.2147 ± 0.0106	0.3463 ± 0.0122	0.1653 ± 0.0109	0.2139 ± 0.0078	**0.1568 ± 0.0094**
	CV	0.5031 ± 0.0061	0.6233 ± 0.0131	0.4455 ± 0.0073	0.4937 ± 0.0092	**0.4400 ± 0.0038**
	AUC	0.7761 ± 0.0089	0.6381 ± 0.0107	0.8217 ± 0.0082	0.7753 ± 0.0062	**0.8276 ± 0.0085**
Pascal07	HL	0.1147 ± 0.0034	0.0661 ± 0.0016	0.0874 ± 0.0004	0.0861 ± 0.0017	**0.0486 ± 0.0013**
	SA	0.0379 ± 0.0098	0.1946 ± 0.0461	0.0936 ± 0.0098	0.0986 ± 0.0118	**0.3942 ± 0.0095**
	AP	0.4594 ± 0.0253	0.6627 ± 0.0088	0.6713 ± 0.0031	0.6555 ± 0.0130	**0.7880 ± 0.0040**
	OE	0.5886 ± 0.0018	0.4741 ± 0.0160	0.4199 ± 0.0028	0.3968 ± 0.0208	**0.2534 ± 0.0100**
	RL	0.2408 ± 0.0403	0.0841 ± 0.0030	0.1068 ± 0.0012	0.1381 ± 0.0115	**0.0659 ± 0.0024**
	CV	0.3085 ± 0.0480	0.1235 ± 0.0034	0.1501 ± 0.0015	0.1888 ± 0.0155	**0.1077 ± 0.0028**
	AUC	0.7574 ± 0.0381	0.9056 ± 0.0029	0.8785 ± 0.0004	0.8522 ± 0.0108	**0.9197 ± 0.0016**
Corel5k	HL	0.0223 ± 0.0001	0.0126 ± 0.0000	0.0176 ± 0.0000	0.0218 ± 0.0000	**0.0115 ± 0.0004**
	SA	0.0000 ± 0.0000	0.0104 ± 0.0024	0.0416 ± 0.0056	0.0000 ± 0.0000	**0.0655 ± 0.0055**
	AP	0.2576 ± 0.0040	0.4180 ± 0.0069	0.4324 ± 0.0076	0.2739 ± 0.0033	**0.5304 ± 0.0095**
	OE	0.6968 ± 0.0071	0.5217 ± 0.0117	0.4724 ± 0.0000	0.6872 ± 0.0026	**0.3762 ± 0.0165**
	RL	0.1494 ± 0.0017	0.0755 ± 0.0019	0.1647 ± 0.0000	0.1299 ± 0.0028	**0.0728 ± 0.0038**
	CV	0.3345 ± 0.0004	**0.1836 ± 0.0059**	0.3813 ± 0.0050	0.2858 ± 0.0041	0.1878 ± 0.0064
	AUC	0.8516 ± 0.0025	0.9239 ± 0.0013	0.8367 ± 0.0000	0.8706 ± 0.0029	**0.9268 ± 0.0043**
Espgame	HL	0.0287 ± 0.0004	0.0173 ± 0.0001		0.0283 ± 0.0001	**0.0169 ± 0.0001**
	SA	0.0000 ± 0.0000	0.0045 ± 0.0006		0.0000 ± 0.0000	**0.0115 ± 0.0020**
	AP	0.2191 ± 0.0130	0.3186 ± 0.0017		0.2366 ± 0.0016	**0.3813 ± 0.0033**
	OE	0.7133 ± 0.0301	0.5594 ± 0.0052	OM	0.6741 ± 0.0002	**0.4751 ± 0.0052**
	RL	0.2026 ± 0.0020	0.1343 ± 0.0009		0.1904 ± 0.0019	**0.1314 ± 0.0011**
	CV	0.4791 ± 0.0008	**0.3421 ± 0.0023**		0.4467 ± 0.0036	0.3552 ± 0.0027
	AUC	0.8005 ± 0.0015	0.8681 ± 0.0008		0.8119 ± 0.0013	**0.8693 ± 0.0011**
Iaprtc12	HL	0.0323 ± 0.0001	0.0194 ± 0.0000		0.0313 ± 0.0000	**0.0191 ± 0.0002**
	SA	0.0000 ± 0.0000	0.0000 ± 0.0000		0.0000 ± 0.0000	**0.0106 ± 0.0019**
	AP	0.2038 ± 0.0003	0.3282 ± 0.0014		0.2422 ± 0.0014	**0.4132 ± 0.0017**
	OE	0.7201 ± 0.0050	0.5410 ± 0.0071	OM	0.6242 ± 0.0015	**0.4493 ± 0.0030**
	RL	0.1891 ± 0.0008	0.1053 ± 0.0014		0.1646 ± 0.0022	**0.1024 ± 0.0010**
	CV	0.4970 ± 0.0010	**0.3066 ± 0.0025**		0.4353 ± 0.0022	0.3221 ± 0.0046
	AUC	0.8110 ± 0.0007	0.8938 ± 0.0000		0.8346 ± 0.0009	**0.8963 ± 0.0012**
Mirflickr	HL	0.0131 ± 0.0000	0.0058 ± 0.0001		0.0132 ± 0.0001	**0.0058 ± 0.0000**
	SA	0.0000 ± 0.0000	0.2518 ± 0.0040		0.0000 ± 0.0000	**0.2554 ± 0.0031**
	AP	0.0925 ± 0.0014	0.0962 ± 0.0022		0.0944 ± 0.0049	**0.0990 ± 0.0000**
	OE	0.9076 ± 0.0045	**0.8771 ± 0.0041**	OM	0.8869 ± 0.0035	0.8781 ± 0.0026
	RL	0.2877 ± 0.0011	0.1831 ± 0.0039		0.2848 ± 0.0087	**0.1693 ± 0.0000**
	CV	0.4985 ± 0.0015	0.3285 ± 0.0053		0.4918 ± 0.0073	**0.3072 ± 0.0011**
	AUC	**0.7283 ± 0.0013**	0.5682 ± 0.0023		0.7268 ± 0.0045	0.5758 ± 0.0031

2. From the comparison of TM3L and LSML results, we can see that the performance of the traditional single-view multi-label method directly connected to the multi-view multi-label learning algorithm is not as good as directly considering the multi-view learning problem model, which verifies that multi-view learning is more effective than single-view learning. TM3L comparing iMvWL and McWL shows that it is important to consider the measure of dependency between different views. In particular, compared to the McWL algorithm, TM3L memory usage is lower. The comparison of TM3L and ICM2L algorithms shows that the optimal solution of the neural network has achieved more effective results.
3. As can be seen from Fig. 2, the TM3L algorithm is significantly better than other algorithms at 78.6% compared to other algorithms. Specifically, on HL, SA, RL, and CV, as shown in Fig. 2, there is no significant difference compared to LSML; It is significantly superior to all other algorithms on AP and OE; On AUC, there is no significant difference from LSML and iMvWL algorithms.

Based on the above analysis, it can be seen that the TM3L algorithm has obtained a certain competitive performance compared with other popular algorithms. A large number of experimental analyzes have verified the effectiveness of considering the combination of subspace learning and multi-label neural network learning to solve multi-view and multi-label missing label learning.

4.5. Component analysis

To further verify the effectiveness of TM3L in capturing dependencies among multiple views, view contribution weights, and label correlations, we conducted additional component analysis experiments on the *Yeast*, *Core*15*k*, and *Pascal*07 data sets, and reported them in Fig. 3. SA and OE values. In Fig. 3, we set the missing labels ratio to 40%. The definitions of TM3L and its variants are as follows: TM3L-D means ignoring the dependency among views; TM3L-LC means ignoring label correlation; TM3L-W means ignoring the contribution of different views.

From Fig. 3, we can observe that TM3L outperforms its variants in all settings. Compared with TM3L-D and TM3L-W, TM3L utilizes the inter-view dependency and view weighting information, respectively, thereby improving the final performance. These results confirm our motives for explicitly using view individual dependence and weight contributions. In most cases, across three data sets, TM3L is better than TM3L-LC. The internal reason is that TM3L captures the correlation between tags, which is crucial in multi-tag learning. Furthermore, it confirms the necessity of obtaining the label correlation and also proves the rationality and validity of the label correlation matrix $\boldsymbol{S}$ learned by TM3L.

Table A.4
Experimental results of 5 algorithms on 6 data sets with the missing label rate of 40%.

Data set	Metric	ICM2L	LSML	McWL	iMvWL	TM3L
Yeast	HL	0.2683 ± 0.0140	0.2633 ± 0.0047	0.2243 ± 0.0066	0.2695 ± 0.0069	**0.1999 ± 0.0023**
	SA	0.0104 ± 0.0083	0.0678 ± 0.0066	0.0135 ± 0.0052	0.0116 ± 0.0056	**0.1631 ± 0.0082**
	AP	0.7061 ± 0.0126	0.6060 ± 0.0070	0.7600 ± 0.00151	0.7096 ± 0.0098	**0.7648 ± 0.0056**
	OE	0.2536 ± 0.0010	0.3538 ± 0.0173	0.2702 ± 0.0066	0.2518 ± 0.0206	**0.2253 ± 0.0129**
	RL	0.2109 ± 0.0092	0.3512 ± 0.0123	0.1675 ± 0.0066	0.2120 ± 0.0055	**0.1662 ± 0.0041**
	CV	0.4941 ± 0.0164	0.6531 ± 0.0091	**0.4434 ± 0.0069**	0.4946 ± 0.0157	0.4600 ± 0.0050
	AUC	0.7786 ± 0.0089	0.6342 ± 0.0117	0.8175 ± 0.0069	0.7788 ± 0.0058	**0.8200 ± 0.0028**
Pascal07	HL	0.1166 ± 0.0008	0.0583 ± 0.0000	0.0918 ± 0.0001	0.0875 ± 0.0026	**0.0468 ± 0.0006**
	SA	0.0309 ± 0.0003	0.2662 ± 0.0111	0.0871 ± 0.0013	0.0955 ± 0.0135	**0.3965 ± 0.0086**
	AP	0.4610 ± 0.0018	0.7061 ± 0.0045	0.6359 ± 0.0000	0.6478 ± 0.0163	**0.7756 ± 0.0071**
	OE	0.5871 ± 0.0008	0.3641 ± 0.0107	0.4724 ± 0.0030	0.4049 ± 0.0194	**0.2658 ± 0.0099**
	RL	0.2363 ± 0.0020	0.0876 ± 0.0022	0.1158 ± 0.0035	0.1438 ± 0.0090	**0.0756 ± 0.0032**
	CV	0.2989 ± 0.0033	0.1346 ± 0.0030	0.1562 ± 0.0055	0.1942 ± 0.0115	**0.1184 ± 0.0024**
	AUC	0.7652 ± 0.0039	0.8945 ± 0.0016	0.8739 ± 0.0048	0.8470 ± 0.0098	**0.9103 ± 0.0025**
Corel5k	HL	0.0243 ± 0.0020	0.0127 ± 0.0000	0.0214 ± 0.0000	0.0216 ± 0.0000	**0.0117 ± 0.0001**
	SA	0.0000 ± 0.0000	0.0056 ± 0.0043	0.0135 ± 0.0025	0.0012 ± 0.0012	**0.0490 ± 0.0052**
	AP	0.1307 ± 0.1134	0.3964 ± 0.0078	0.2739 ± 0.0053	0.2772 ± 0.0054	**0.4999 ± 0.0090**
	OE	0.8742 ± 0.1258	0.5279 ± 0.0074	0.6976 ± 0.0135	0.6754 ± 0.0056	**0.4104 ± 0.0136**
	RL	**0.0754 ± 0.0754**	0.0875 ± 0.0030	0.1434 ± 0.0000	0.1291 ± 0.0028	0.0835 ± 0.0056
	CV	0.4113 ± 0.0881	**0.2121 ± 0.0056**	0.3151 ± 0.0020	0.2875 ± 0.0045	0.2144 ± 0.0125
	AUC	0.7896 ± 0.0606	0.9123 ± 0.0043	0.8582 ± 0.0000	0.8714 ± 0.0036	**0.9166 ± 0.0056**
Espgame	HL	0.0290 ± 0.0002	0.0173 ± 0.0001		0.0285 ± 0.0001	**0.0172 ± 0.0002**
	SA	0.0000 ± 0.0000	0.0044 ± 0.0005		0.0000 ± 0.0000	**0.0106 ± 0.0008**
	AP	0.2045 ± 0.0024	0.3095 ± 0.0017		0.2324 ± 0.0054	**0.3611 ± 0.0023**
	OE	0.7558 ± 0.0035	0.5630 ± 0.0064	OM	0.6787 ± 0.0182	**0.4944 ± 0.0046**
	RL	0.2036 ± 0.0030	0.1519 ± 0.0011		0.1909 ± 0.0010	**0.1518 ± 0.0029**
	CV	0.4763 ± 0.0064	**0.3890 ± 0.0032**		0.4519 ± 0.0028	0.4056 ± 0.0055
	AUC	0.8000 ± 0.0027	**0.8509 ± 0.0014**		0.8120 ± 0.0009	0.8493 ± 0.0016
Iaprtc12	HL	0.0322 ± 0.0000	0.0194 ± 0.0000		0.0314 ± 0.0000	**0.0190 ± 0.0001**
	SA	0.0000 ± 0.0000	0.0012 ± 0.0000		0.0003 ± 0.0003	**0.0056 ± 0.0006**
	AP	0.2093 ± 0.0034	0.3208 ± 0.0027		0.2362 ± 0.0015	**0.3921 ± 0.0028**
	OE	0.6850 ± 0.0131	0.5572 ± 0.0054	OM	0.6448 ± 0.0074	**0.4652 ± 0.0043**
	RL	0.1847 ± 0.0020	0.1144 ± 0.0012		0.1643 ± 0.0002	**0.1044 ± 0.0013**
	CV	0.4886 ± 0.0032	**0.3297 ± 0.0045**		0.4339 ± 0.0021	0.3305 ± 0.0033
	AUC	0.8156 ± 0.0022	0.8859 ± 0.0011		0.8344 ± 0.0003	**0.8938 ± 0.0008**
Mirflickr	HL	0.0133 ± 0.0001	0.0058 ± 0.0001		0.0133 ± 0.0000	**0.0058 ± 0.0001**
	SA	0.0000 ± 0.0000	0.2508 ± 0.0046		0.0000 ± 0.0000	**0.2530 ± 0.0078**
	AP	0.0921 ± 0.0045	0.0885 ± 0.0013		0.0940 ± 0.0024	**0.0954 ± 0.0018**
	OE	0.8988 ± 0.0082	0.8868 ± 0.0039	OM	0.8888 ± 0.0100	**0.8800 ± 0.0031**
	RL	0.2875 ± 0.0025	0.2043 ± 0.0023		0.2863 ± 0.0001	**0.1840 ± 0.0031**
	CV	0.4981 ± 0.0023	0.3638 ± 0.0035		0.4903 ± 0.0005	**0.3325 ± 0.0042**
	AUC	**0.7276 ± 0.0034**	0.5435 ± 0.0034		0.7270 ± 0.0028	0.5609 ± 0.0042

4.6. Parameter sensitivity analysis

TM3L has four trade-off parameters C, α, β and λ. We tested the sensitivity of TM3L to four parameters on the *Yeast*, *Corel5k* and *Pascal07* data sets, and we fixed the values of three parameters to known values (for example $C = 10^{-1}$, $\alpha = 10^{5}$, $\beta = 10^{3}$, and $\lambda = 10^{-5}$), and then change the value of one of the parameters in the range $\{10^{-5}, 10^{-4}, \ldots, 10^{5}\}$. Fig. 4 shows the change of the *HL* and *AUC* values of the algorithm when the label missing rate of $w\% = 40\%$.

The parameter C controls the scaling ratio of the shared space coefficient. For the parameter α, it controls the contribution of all views. The parameter β controls the magnitude of the effect of local label correlation. The parameter λ controls the dependency of different private subspaces. It can be observed that the parameter C is effective in taking the intermediate value. Similar to parameter C, parameter β tends to choose an intermediate value. Besides, the β value should not be too large, which will cause TM3L to be unable to obtain valid label correlation information. From Fig. 4, the values of the other two regularization parameters are relatively insensitive to changes in TM3L performance. Similar results and similar results can be obtained on the evaluation metrics of other data sets.

Further, we design experiments to study the sensitivity of the shared space dimension r change to the results of the TM3L algorithm. Fig. 5 reports the *HL* and *AUC* values of TM3L on the *Yeast* data set, with r varying from $0.1d_{min}$ to $0.9d_{min}$. It can be seen that the performance of TM3L keeps increasing with the increase of r. So in the experiment, we set $r = 0.9d_{min}$. Intuitive understanding is that because of the kernel extreme learning machine classifier used in the second step of the solution process, increasing the feature amount can effectively improve the performance of the neural network algorithm.

4.7. TM3L algorithm iteration efficiency

We report in this section the convergence trend of TM3L's first step for all methods at the label missing rate of 40%(The main optimization process of TM3L is the optimization process of the first step. In this section, we only consider the convergence trend of the first step). Fig. 6 shows the convergence curves on the Pascal07 and Core15k data sets on the first step. It can be seen from Fig. 6 that TM3L tends to iterate 25 times on both data sets, confirming that our algorithm can converge faster and iterate. Convergence results are similar to other data sets.

5. Conclusion

In this paper, we propose a TM3L method to solve the multi-label learning problem of multi-view with missing labels. We

Table A.5
Experimental results of 5 algorithms on 6 data sets with the missing label rate of 60%.

Data set	Metric	ICM2L	LSML	McWL	iMvWL	TM3L
Yeast	HL	0.2726 ± 0.0106	0.2689 ± 0.0086	0.2269 ± 0.0028	0.2661 ± 0.0065	**0.1981 ± 0.0077**
	SA	0.0062 ± 0.0041	0.0372 ± 0.0123	0.0186 ± 0.0062	0.0095 ± 0.0088	**0.1582 ± 0.0166**
	AP	0.6873 ± 0.0119	0.5864 ± 0.0109	**0.7617 ± 0.0095**	0.7018 ± 0.0049	0.7589 ± 0.0206
	OE	0.2899 ± 0.0083	0.3661 ± 0.0250	0.2557 ± 0.0135	0.2658 ± 0.0074	**0.2331 ± 0.0269**
	RL	0.2241 ± 0.0117	0.3711 ± 0.0105	**0.1645 ± 0.0058**	0.2113 ± 0.0056	0.1690 ± 0.0140
	CV	0.5148 ± 0.0197	0.6905 ± 0.0095	**0.4348 ± 0.0025**	0.4890 ± 0.0100	0.4568 ± 0.0093
	AUC	0.7655 ± 0.0112	0.6161 ± 0.0083	**0.8191 ± 0.0035**	0.7791 ± 0.0064	0.8164 ± 0.0126
Pascal07	HL	0.1183 ± 0.0000	0.0581 ± 0.0000	0.0959 ± 0.0002	0.0899 ± 0.0024	**0.0522 ± 0.0010**
	SA	0.0266 ± 0.0005	0.2623 ± 0.0083	0.0806 ± 0.0043	0.0898 ± 0.0069	**0.3216 ± 0.0087**
	AP	0.4260 ± 0.0047	0.6978 ± 0.0065	0.6009 ± 0.0023	0.6377 ± 0.0153	**0.7411 ± 0.0040**
	OE	0.5989 ± 0.0000	0.3661 ± 0.0076	0.5010 ± 0.0065	0.4149 ± 0.0116	**0.3232 ± 0.0081**
	RL	0.2940 ± 0.0111	0.0950 ± 0.0027	0.1350 ± 0.0005	0.1503 ± 0.0136	**0.0774 ± 0.0011**
	CV	0.3673 ± 0.0153	0.1448 ± 0.0036	0.1797 ± 0.0016	0.2020 ± 0.0149	**0.1196 ± 0.0029**
	AUC	0.7085 ± 0.0092	0.8848 ± 0.0033	0.8586 ± 0.0001	0.8392 ± 0.0135	**0.9090 ± 0.0018**
Corel5k	HL	0.0224 ± 0.0001	0.0127 ± 0.0000	0.0175 ± 0.0000	0.0216 ± 0.0000	**0.0119 ± 0.0002**
	SA	0.0000 ± 0.0000	0.0054 ± 0.0029	0.0350 ± 0.0069	0.0000 ± 0.0000	**0.0468 ± 0.0076**
	AP	0.2581 ± 0.0126	0.4002 ± 0.0052	0.4377 ± 0.0123	0.2781 ± 0.0053	**0.4883 ± 0.0105**
	OE	0.6882 ± 0.0185	0.5377 ± 0.0111	0.4780 ± 0.0185	0.6773 ± 0.0179	**0.4228 ± 0.0177**
	RL	0.1484 ± 0.0065	0.0921 ± 0.0025	0.1148 ± 0.0040	0.1340 ± 0.0033	**0.0912 ± 0.0046**
	CV	0.3319 ± 0.0123	**0.2251 ± 0.0078**	0.2713 ± 0.0057	0.2962 ± 0.0066	0.2319 ± 0.0082
	AUC	0.8515 ± 0.0056	0.9081 ± 0.0031	0.8863 ± 0.0045	0.8663 ± 0.0032	**0.9082 ± 0.0042**
Espgame	HL	0.0286 ± 0.0004	0.0173 ± 0.0001		0.0286 ± 0.0001	**0.0172 ± 0.0001**
	SA	0.0000 ± 0.0000	0.0038 ± 0.0006		0.0000 ± 0.0000	**0.0099 ± 0.0018**
	AP	0.2129 ± 0.0103	0.3140 ± 0.0040		0.2265 ± 0.0047	**0.3469 ± 0.0044**
	OE	0.7375 ± 0.0193	0.5639 ± 0.0111	OM	0.7019 ± 0.0155	**0.5039 ± 0.0069**
	RL	0.2013 ± 0.0041	**0.1433 ± 0.0012**		0.1912 ± 0.0027	0.1686 ± 0.0016
	CV	0.4702 ± 0.0071	**0.3662 ± 0.0019**		0.4501 ± 0.0043	0.4442 ± 0.0030
	AUC	0.8043 ± 0.0043	**0.8592 ± 0.0016**		0.8114 ± 0.0022	0.8308 ± 0.0013
Iaprtc12	HL	0.0325 ± 0.0001	0.0194 ± 0.0000		0.0312 ± 0.0002	**0.0189 ± 0.0002**
	SA	0.0000 ± 0.0000	0.0000 ± 0.0000		0.0000 ± 0.0000	**0.0045 ± 0.0007**
	AP	0.2041 ± 0.0017	0.3205 ± 0.0027		0.2406 ± 0.0000	**0.3799 ± 0.0032**
	OE	0.7118 ± 0.0112	0.5508 ± 0.0089	OM	0.6358 ± 0.0019	**0.4745 ± 0.0049**
	RL	0.1871 ± 0.0038	0.1176 ± 0.0011		0.1660 ± 0.0021	**0.1169 ± 0.0023**
	CV	0.4935 ± 0.0088	**0.3446 ± 0.0028**		0.4398 ± 0.0015	0.3676 ± 0.0055
	AUC	0.8120 ± 0.0035	**0.8830 ± 0.0000**		0.8333 ± 0.0009	0.8821 ± 0.0021
Mirflickr	HL	0.0132 ± 0.0001	**0.0058 ± 0.0001**		0.0132 ± 0.0000	0.0059 ± 0.0000
	SA	0.0000 ± 0.0000	0.2505 ± 0.0051		0.0000 ± 0.0000	**0.2515 ± 0.0059**
	AP	0.0951 ± 0.0008	0.0920 ± 0.0014		0.0945 ± 0.0002	**0.0981 ± 0.0023**
	OE	0.8981 ± 0.0043	0.8830 ± 0.0029	OM	0.8901 ± 0.0010	**0.8760 ± 0.0041**
	RL	0.2869 ± 0.0018	0.1939 ± 0.0015		0.4975 ± 0.0027	**0.1791 ± 0.0014**
	CV	0.4989 ± 0.0001	0.3464 ± 0.0036		0.4975 ± 0.0027	**0.3243 ± 0.0022**
	AUC	**0.7266 ± 0.0005**	0.5558 ± 0.0036		0.7252 ± 0.0010	0.5714 ± 0.0062

adopt a two-step learning strategy. The first step is to learn the shared subspace of the entire views data set. We learn to obtain shared subspace information under the unified framework by using the dependency of the private spatial information and the difference in the contribution of each view. In the second step, a multi-label classification learning method of extreme learning machine combined with the correlation between labels is proposed to solve the multi-label classification problem under incomplete label sets. The biggest difference from previous learning methods that focus on multi-view and multi-label joint learning by capturing shared and individual information between multiple views is that TM3L has the advantages of both multi-view learning and multi-label learning algorithms. A large number of experiments on 6 benchmark data sets show that our proposed method has a certain competitive performance, and effectively solves the problem of the missing label through shared subspace learning and effective utilize of label correlations.

Although this two-step learning method has a high learning rate, it also faces such problems. (1) There is no communication between subspace learning and label space, which leads to the process of model solving may be a suboptimal model. (2) The matrix factorization may ignore the non-linear characteristics of the data. In the future, we will consider solving the above problems.

CRediT authorship contribution statement

Dawei Zhao: Conceptualization, Methodology, Software, Investigation, Data curation, Writing - original draft, Writing - review & editing. **Qingwei Gao:** Validation, Supervision, Project administration, Funding acquisition. **Yixiang Lu:** Visualization, Investigation. **Dong Sun:** Formal analysis, Validation.

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.

Appendix. Experimental results

See Tables A.3–A.6.

Table A.6
Experimental results of 5 algorithms on 6 data sets with the missing label rate of 90%.

Data set	Metric	ICM2L	LSML	McWL	iMvWL	TM3L
Yeast	HL	0.2673 ± 0.0075	0.2802 ± 0.0044	0.2792 ± 0.0021	0.2677 ± 0.0044	**0.2149 ± 0.0023**
	SA	0.0098 ± 0.0009	0.0087 ± 0.0030	0.0176 ± 0.0031	0.0066 ± 0.0038	**0.0791 ± 0.0173**
	AP	0.7041 ± 0.0062	0.5629 ± 0.0104	0.7025 ± 0.0020	0.7025 ± 0.0125	**0.7434 ± 0.0123**
	OE	0.2679 ± 0.0024	0.3993 ± 0.0129	0.2464 ± 0.0083	0.2658 ± 0.0348	**0.2398 ± 0.0218**
	RL	0.2100 ± 0.0067	0.4002 ± 0.0118	0.2123 ± 0.0002	0.2095 ± 0.0048	**0.1836 ± 0.0053**
	CV	0.4900 ± 0.0110	0.7332 ± 0.0085	0.4834 ± 0.0017	0.4867 ± 0.0098	**0.4822 ± 0.0060**
	AUC	0.7798 ± 0.0053	0.5866 ± 0.0116	0.7790 ± 0.0002	0.7792 ± 0.0047	**0.8031 ± 0.0041**
Pascal07	HL	0.1172 ± 0.0010	0.0595 ± 0.0000	0.1179 ± 0.0000	0.1064 ± 0.0051	**0.0549 ± 0.0003**
	SA	0.0319 ± 0.0038	0.2410 ± 0.0069	0.0339 ± 0.0008	0.0420 ± 0.0222	**0.2897 ± 0.0107**
	AP	0.4374 ± 0.0039	0.6678 ± 0.0071	0.4270 ± 0.0043	0.5273 ± 0.0322	**0.6908 ± 0.0072**
	OE	0.5863 ± 0.0035	0.3882 ± 0.0093	0.5919 ± 0.0010	0.5789 ± 0.0660	**0.3611 ± 0.0111**
	RL	0.2772 ± 0.0075	**0.1196 ± 0.0058**	0.3071 ± 0.0100	0.1973 ± 0.0151	0.1203 ± 0.0047
	CV	0.3541 ± 0.0102	0.1763 ± 0.0077	0.3802 ± 0.0084	0.2524 ± 0.0159	**0.1735 ± 0.0059**
	AUC	0.7227 ± 0.0081	0.8572 ± 0.0062	0.7020 ± 0.0071	0.7892 ± 0.0155	**0.8646 ± 0.0047**
Corel5k	HL	0.0224 ± 0.0001	0.0130 ± 0.0000	0.0213 ± 0.0000	0.0218 ± 0.0000	**0.0126 ± 0.0002**
	SA	0.0005 ± 0.0005	0.0048 ± 0.0023	0.0060 ± 0.0046	0.0018 ± 0.0026	**0.0144 ± 0.0054**
	AP	0.2435 ± 0.0156	0.3147 ± 0.0049	0.2792 ± 0.0093	0.2661 ± 0.0024	**0.3565 ± 0.0080**
	OE	0.7086 ± 0.0466	0.6081 ± 0.0107	0.6742 ± 0.0183	0.6969 ± 0.0165	**0.5459 ± 0.0155**
	RL	0.1648 ± 0.0120	0.1835 ± 0.0053	**0.1468 ± 0.0017**	0.1541 ± 0.0033	0.1794 ± 0.0033
	CV	0.3572 ± 0.0235	0.4017 ± 0.0073	**0.3239 ± 0.0019**	0.3467 ± 0.0107	0.4201 ± 0.0074
	AUC	0.8354 ± 0.0111	0.8243 ± 0.0034	**0.8537 ± 0.0019**	0.8462 ± 0.0036	0.8203 ± 0.0028
Espgame	HL	0.0283 ± 0.0001	**0.0174 ± 0.0001**		0.0284 ± 0.0001	0.0175 ± 0.0000
	SA	0.0000 ± 0.0000	0.0060 ± 0.0009		0.0000 ± 0.0000	**0.0089 ± 0.0015**
	AP	0.2215 ± 0.0025	0.2781 ± 0.0023		0.2308 ± 0.0029	**0.3446 ± 0.0030**
	OE	0.7207 ± 0.0053	0.5838 ± 0.0032	OM	0.6856 ± 0.0051	**0.5042 ± 0.0050**
	RL	0.1967 ± 0.0020	0.1992 ± 0.0013		0.1992 ± 0.0034	**0.1766 ± 0.0022**
	CV	0.4677 ± 0.0030	0.5048 ± 0.0048		0.4711 ± 0.0074	**0.4642 ± 0.0061**
	AUC	0.8072 ± 0.0025	0.7961 ± 0.0014		0.8039 ± 0.0029	**0.8232 ± 0.0025**
Iaprtc12	HL	0.0323 ± 0.0003	0.0195 ± 0.0000		0.0313 ± 0.0000	**0.0193 ± 0.0001**
	SA	0.0000 ± 0.0000	**0.0029 ± 0.0000**		0.0000 ± 0.0000	0.0002 ± 0.0001
	AP	0.2076 ± 0.0101	0.2933 ± 0.0026		0.2420 ± 0.0001	**0.2972 ± 0.0044**
	OE	0.6944 ± 0.0084	0.5711 ± 0.0034	OM	0.6197 ± 0.0019	**0.5710 ± 0.0100**
	RL	0.1891 ± 0.0073	0.1518 ± 0.0000		0.1684 ± 0.0010	**0.1425 ± 0.0027**
	CV	0.5000 ± 0.0148	0.4495 ± 0.0032		0.4483 ± 0.0016	**0.4187 ± 0.0072**
	AUC	0.8117 ± 0.0070	0.8410 ± 0.0011		0.8303 ± 0.0003	**0.8547 ± 0.0028**
Mirflickr	HL	0.0131 ± 0.0000	0.0059 ± 0.0001		0.0132 ± 0.0000	**0.0058 ± 0.0001**
	SA	0.0000 ± 0.0000	0.2433 ± 0.0036		0.0000 ± 0.0000	**0.2583 ± 0.0037**
	AP	**0.0988 ± 0.0034**	0.0679 ± 0.0008		0.0942 ± 0.0024	0.0809 ± 0.0010
	OE	**0.8855 ± 0.0100**	0.9170 ± 0.0029	OM	0.8892 ± 0.0048	0.8936 ± 0.0010
	RL	0.2920 ± 0.0027	0.2496 ± 0.0043		0.3066 ± 0.0003	**0.2249 ± 0.0030**
	CV	0.5081 ± 0.0067	0.4261 ± 0.0067		0.5266 ± 0.0009	**0.3930 ± 0.0060**
	AUC	**0.7194 ± 0.0036**	0.4896 ± 0.0022		0.7022 ± 0.0002	0.5193 ± 0.0020

References

[1] Zhe Xue, Guorong Li, Qingming Huang, Joint multi-view representation and image annotation via optimal predictive subspace learning, Inform. Sci. 451 (2018) 180–194.

[2] Xuan Wu, Qing-Guo Chen, Yao Hu, Dengbao Wang, Xiaodong Chang, Xiaobo Wang, Min-Ling Zhang, Multi-view multi-label learning with view-specific information extraction, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press, 2019, pp. 3884–3890.

[3] Ze-Sen Chen, Xuan Wu, Qing-Guo Chen, Yao Hu, Min-Ling Zhang, Multi-view partial multi-label learning with graph-based disambiguation., in: AAAI, 2020, pp. 3553–3560.

[4] Yifeng Li, Fang-Xiang Wu, Alioune Ngom, A review on machine learning principles for multi-view biological data integration, Brief. Bioinform. 19 (2) (2016) 325–340.

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

[6] Fei Wu, Xiao-Yuan Jing, Xinge You, Dong Yue, Ruimin Hu, Jing-Yu Yang, Multi-view low-rank dictionary learning for image classification, Pattern Recognit. 50 (2016) 143–154.

[7] Sebastian Ruder, An overview of multi-task learning in deep neural networks, 2017, arXiv preprint arXiv:1706.05098.

[8] Min-Ling Zhang, Yu-Kun Li, Xu-Ying Liu, Xin Geng, Binary relevance for multi-label learning: an overview, Front. Comput. Sci. 12 (2) (2018) 191–202.

[9] Min-Ling Zhang, Zhi-Hua Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowl. Data Eng. 26 (8) (2013) 1819–1837.

[10] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, Christopher M. Brown, Learning multi-label scene classification, Pattern Recognit. 37 (9) (2004) 1757–1771.

[11] Min-Ling Zhang, Zhi-Hua Zhou, ML-KNN: A lazy learning approach to multi-label learning, Pattern Recognit. 40 (7) (2007) 2038–2048.

[12] Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencía, Klaus Brinker, Multilabel classification via calibrated label ranking, Mach. Learn. 73 (2) (2008) 133–153.

[13] Yu Zhang, Dit-Yan Yeung, Multilabel relationship learning, ACM Trans. Knowl. Discov. Data (TKDD) 7 (2) (2013) 7.

[14] Lu Sun, Mineichi Kudo, Keigo Kimura, Multi-label classification with meta-label-specific features, in: 2016 23rd International Conference on Pattern Recognition, ICPR, IEEE, 2016, pp. 1612–1617.

[15] Zhi-Fen He, Ming Yang, Yang Gao, Hui-Dong Liu, Yilong Yin, Joint multi-label classification and label correlations with missing labels and feature selection, Knowl.-Based Syst. 163 (2019) 145–158.

[16] Jun Huang, Xiwen Qu, Guorong Li, Feng Qin, Xiao Zheng, Qingming Huang, Multi-view multi-label learning with view-label-specific features, IEEE Access 7 (2019) 100979–100992.

[17] Weijieying Ren, Lei Zhang, Bo Jiang, Zhefeng Wang, Guangming Guo, Guiquan Liu, Robust mapping learning for multi-view multi-label classification with missing labels, in: International Conference on Knowledge Science, Engineering and Management, Springer, 2017, pp. 543–551.

[18] Xuran Zhao, Nicholas Evans, Jean-Luc Dugelay, A subspace co-training framework for multi-view clustering, Pattern Recognit. Lett. 41 (2014) 73–82.

[19] Riikka Huusari, Cécile Capponi, Paul Villoutreix, Hachem Kadri, Kernel transfer over multiple views for missing data completion, 2019, arXiv preprint arXiv:1910.05964.

[20] Meng Liu, Yong Luo, Dacheng Tao, Chao Xu, Yonggang Wen, Low-rank multi-view learning in matrix completion for multi-label image classification, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.

[21] Qiaoyu Tan, Guoxian Yu, Jun Wang, Carlotta Domeniconi, Xiangliang Zhang, Individuality-and commonality-based multiview multilabel learning, IEEE Trans. Cybern. (2019).
[22] Changqing Zhang, Ziwei Yu, Qinghua Hu, Pengfei Zhu, Xinwang Liu, Xiaobo Wang, Latent semantic aware multi-view multi-label classification, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[23] Qiaoyu Tan, Guoxian Yu, Carlotta Domeniconi, Jun Wang, Zili Zhang, Incomplete multi-view weak-label learning, in: IJCAI, 2018, pp. 2703–2709.
[24] Qiaoyu Tan, Guoxian Yu, Carlotta Domeniconi, Jun Wang, Zili Zhang, Multi-view weak-label learning based on matrix completion, in: Proceedings of the 2018 SIAM International Conference on Data Mining, SIAM, 2018, pp. 450–458.
[25] Cheng Liang, Shengpeng Yu, Jiawei Luo, Adaptive multi-view multi-label learning for identifying disease-associated candidate mirnas, PLoS Comput. Biol. 15 (4) (2019) e1006931.
[26] Xuesong Niu, Hu Han, Shiguang Shan, Xilin Chen, Multi-label co-regularization for semi-supervised facial action unit recognition, in: Advances in Neural Information Processing Systems, 2019, pp. 907–917.
[27] Sheng Li, Yaliang Li, Yun Fu, Multi-view time series classification: A discriminative bilinear projection approach, in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, 2016, pp. 989–998.
[28] Sheng Li, Ming Shao, Yun Fu, Multi-view low-rank analysis for outlier detection, in: Proceedings of the 2015 SIAM International Conference on Data Mining, SIAM, 2015, pp. 748–756.
[29] Yuying Xing, Guoxian Yu, Carlotta Domeniconi, Jun Wang, Zili Zhang, Multi-label co-training, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, AAAI Press, 2018, pp. 2882–2888.
[30] Shirui Luo, Changqing Zhang, Wei Zhang, Xiaochun Cao, Consistent and specific multi-view subspace clustering, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[31] Xiaofeng Zhu, Xuelong Li, Shichao Zhang, Block-row sparse multiview multilabel learning for image classification, IEEE Trans. Cybern. 46 (2) (2015) 450–461.
[32] Pengfei Zhu, Qi Hu, Qinghua Hu, Changqing Zhang, Zhizhao Feng, Multi-view label embedding, Pattern Recognit. 84 (2018) 126–135.
[33] Yongshan Zhang, Jia Wu, Zhihua Cai, S.Y.u Philip, Multi-view multi-label learning with sparse feature selection for image annotation, IEEE Trans. Multimed. (2020).
[34] Jianghong Ma, Tommy W.S. Chow, Robust non-negative sparse graph for semi-supervised multi-label learning with missing labels, Inform. Sci. 422 (2018) 336–351.
[35] Yusheng Cheng, Kun Qian, Yibin Wang, Dawei Zhao, Missing multi-label learning with non-equilibrium based on classification margin, Appl. Soft Comput. (2019) 105924.
[36] Miao Xu, Rong Jin, Zhi-Hua Zhou, Speedup matrix completion with side information: Application to multi-label learning, in: Advances in Neural Information Processing Systems, 2013, pp. 2301–2309.
[37] Jianghong Ma, Zhaoyang Tian, Haijun Zhang, Tommy W.S. Chow, Multi-label low-dimensional embedding with missing labels, Knowl.-Based Syst. 137 (2017) 65–82.
[38] Yue Zhu, James T. Kwok, Zhi-Hua Zhou, Multi-label learning with global and local label correlation, IEEE Trans. Knowl. Data Eng. 30 (6) (2017) 1081–1094.
[39] Serhat Selcuk Bucak, Rong Jin, Anil K. Jain, Multi-label learning with incomplete class assignments, in: CVPR 2011, IEEE, 2011, pp. 2801–2808.
[40] Sheng Li, Yun Fu, Robust multi-label semi-supervised classification, in: 2017 IEEE International Conference on Big Data, Big Data, IEEE, 2017, pp. 27–36.
[41] Arthur Gretton, Olivier Bousquet, Alex Smola, Bernhard Schölkopf, Measuring statistical dependence with Hilbert–Schmidt norms, in: International Conference on Algorithmic Learning Theory, Springer, 2005, pp. 63–77.
[42] Xi-Zhu Wu, Zhi-Hua Zhou, A unified view of multi-label performance measures, in: Proceedings of the 34th International Conference on Machine Learning, Vol. 70, JMLR. org, 2017, pp. 3780–3788.
[43] Sheng-Jun Huang, Zhi-Hua Zhou, Multi-label learning by exploiting label correlations locally, in: Twenty-Sixth AAAI Conference on Artificial Intelligence, 2012.
[44] Changming Zhu, Duoqian Miao, Zhe Wang, Rigui Zhou, Lai Wei, Xiafen Zhang, Global and local multi-view multi-label learning, Neurocomputing 371 (2020) 67–77.
[45] Stephen Boyd, Stephen P. Boyd, Lieven Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[46] Gao Huang, Guang-Bin Huang, Shiji Song, Keyou You, Trends in extreme learning machines: A review, Neural Netw. 61 (2015) 32–48.
[47] Yusheng Cheng, Dawei Zhao, Yibin Wang, Gensheng Pei, Multi-label learning with kernel extreme learning machine autoencoder, Knowl.-Based Syst. 178 (2019) 1–10.
[48] Matthieu Guillaumin, Jakob Verbeek, Cordelia Schmid, Multimodal semi-supervised learning for image classification, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 902–909.
[49] Eva Gibaja, Sebastián Ventura, A tutorial on multilabel learning, ACM Comput. Surv. 47 (3) (2015) 52.
[50] Jia Zhang, Candong Li, Donglin Cao, Yaojin Lin, Songzhi Su, Liang Dai, Shaozi Li, Multi-label learning with label-specific features by resolving label correlations, Knowl.-Based Syst. 159 (2018) 148–157.
[51] Jun Huang, Feng Qin, Xiao Zheng, Zekai Cheng, Zhixiang Yuan, Weigang Zhang, Qingming Huang, Improving multi-label classification with missing labels by learning label-specific features, Inform. Sci. 492 (2019) 124–146.
[52] Janez Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (Jan) (2006) 1–30.