



Label Embedding for Multi-label Classification Via Dependence Maximization

Yachong Li¹ · Youlong Yang¹

Published online: 17 August 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Multi-label classification has aroused extensive attention in various fields. With the emergence of high-dimensional label space, academia has devoted to performing label embedding in recent years. Whereas current embedding approaches do not take feature space correlation sufficiently into consideration or require an encoding function while learning embedded space. Besides, few of them can be spread to track the missing labels. In this paper, we propose a Label Embedding method via Dependence Maximization (LEDM), which obtains the latent space on which the label and feature information can be embedded simultaneously. To end this, the low-rank factorization model on the label matrix is applied to exploit label correlations instead of the encoding process. The dependence between feature space and label space is increased by the Hilbert–Schmidt independence criterion to facilitate the predictability. The proposed LEDM can be easily extended the missing labels in learning embedded space at the same time. Comprehensive experimental results on data sets validate the effectiveness of our approach over the state-of-art methods on both complete-label and missing-label cases.

Keywords Multi-label learning · Label embedding · Low-rank factorization · Hilbert–Schmidt independence criterion · Missing labels

1 Introduction

In machine learning, multi-label classification refers to the situation where an instance is associated with a set of labels simultaneously. It has such widespread applications, including text classification [1], categorization of genes [2], image annotation [3] and so on. Therefore, multi-label classification has caused more and more attention of academic circles.

Currently, there are two principal approaches with respect to multi-label learning. One is called problem transformation, which is to switch multi-label classification tasks into multiple

✉ Youlong Yang
ylyang@mail.xidian.edu.cn
Yachong Li
Licg1415@163.com

¹ School of Mathematics and Statistics, Xidian University, Xi'an 710071, People's Republic of China

single label classification tasks, such as Binary Relevance (BR) [4,5], Label Power-set [6] and Classifier Chain [7]. Another is algorithm adaptation. It extends available classification techniques directly, for instance, Multi-label K-Nearest Neighbor [8] and Adaboost.MH [9]. Nevertheless, with the exponential increase of the number of labels, it is computationally impractical for many conventional multi-label classification algorithms to work in the initial label space. Under such conditions, a great number of label embedding methods are designed to alleviate the problem, which not only improves the classification performance, but also reduces the cost of training and predicting.

Label embedding is a popular paradigm by viewing all possible label sets as a high-dimensional label vector, which concentrates on transforming original label space into an lower-dimensional embedded space in different manners. In the meantime, it takes full advantage of the correlation among all labels, identifying the hidden structure of the original space. All kinds of methods have been proposed to study this. Compressed Sensing (CS) [10] was the first attempt to compress label space to a low dimensional space using a random projection based on the label sparsity. As a result of huge time consumption in the prediction step, Principle Label Space Transformation (PLST) [11] was designed, similar to PCA technique. It mainly obtains the projection matrix and decoder by performing the SVD on the label matrix efficiently. The Conditional Principle Label Space Transformation (CPLST) [12] was proposed to introduce the feature space information into PLST. To optimize the different criteria, the Cost-sensitive Label Embedding with Multidimensional Scaling (CLEMS) [13] takes the evaluation criteria into consideration, which calculates the embedded vectors by multidimensional scaling. More recently, the End to End Feature-aware method [14] was presented on the basis of the canonical correlation analysis theory. The only drawback is that the linear correlation between the instance space and label space is described.

Furthermore, it is extremely hard to get all appropriate labels. As a result, a partial set of labels are only observed [15,16]. Therefore, a great many approaches [17–21] proposed attempt to handle the missing labels. A Semi-supervised algorithm for Multi-label learning by solving a Sylvester Equation (SMSE2) [19] treats the missing labels as negative labels. This is due to the hypothesis that a great proportion of the available labels are negative for each instance. In order to facilitate the classification performance, Multi-label Learning with Missing Labels (MLML) [21] and MLML Using mixed dependency graphs (MLMG) [17] are formulated to distinguish the missing and negative labels explicitly. Positive, negative and missing labels are respectively denoted as $+1$, -1 and 0 or 0 , 1 , and $\frac{1}{2}$. Related many methods [22,23] take advantage of low rank assumption of the entire label matrix to recover missing labels. In general, minimizing the rank of a matrix is converted to a minimized nuclear norm [20,24,25].

Label embedding containing missing labels is a huge challenge than missing labels or label embedding separately. It is worth mentioning that the missing labels have significant effects on the performance of multi-label classification algorithms. Based on this, Zhu et al. [26] developed a new approach GLOCAL to obtain a lower-dimensional latent space and restore the missing labels. Besides, the Low Rank multi-label classification with Missing Labels (LRML) [27] is also designed to deal with the missing labels by using the low-rank mapping. However, neither of them utilize the correlation between feature space and embedding space. In this paper, we propose a novel method called Label Embedding via Dependence Maximization (LEDM) making the utmost of the global and local label relationship. On the one hand, the proposed LEDM derives the embedded space by low-rank factorization on the label matrix. Further more, to measure the dependence between instance space and label space, the Hilbert-Schmidt independence criterion (HSIC) is adopted to capture the nonlinear correlation. On the other hand, the missing labels are also recovered through applying

low-rank factorization model and Laplacian manifold regularization based on instance-level and class-level. It is well known that low-rank factorization model provides a theoretical basis for well recovering the missing labels. Therefore, the proposed method can be used for handling with both the complete labels and missing labels. In the end, all mentioned above are integrated into one optimization model and an effective alternating algorithm is presented to solve it.

The rest of the paper is organized as follows. Section 2 introduces the related works of multi-label classification, especially focusing on the low rank factorization model and HSIC. In Sect. 3, the proposed LEDM is described in detail. Section 4 gives the optimization methods to deal with complete labels and missing labels. Then we present the experimental results and analyses in Sect. 5. Finally, Sect. 6 gives several concluding remarks and issues for future work.

2 Related Work

Let $X = [x_1, \dots, x_n]^T \in R^{n \times m}$ denote a data matrix consisting of n training examples, and $Y = [y_1, \dots, y_n]^T \in R^{n \times c}$ be a label matrix, where $y_i \in \{0, \frac{1}{2}, 1\}^c$ is the corresponding label vector of x_i . That is, each example x_i can take one or more labels from the c different classes. If $Y_{i,j} = 1$, the instance x_i is associated with the j -th label, and if $Y_{i,j} = 0$, x_i does not have the j -th label. $Y_{i,j} = \frac{1}{2}$ indicates that the j -th label is considered unknown for the data point x_i .

2.1 Low-Rank Factorization Model in Multi-Label Learning

Given the presence of correlations among different labels, the whole label matrix is viewed as row-rank. Therefore its rank is smaller than its size [28,29]. For instance, when labels “white cloud” and “blue sky” are present simultaneously, it is completely possible to appear label “sunny”. There are a great number of ways to utilize the low-rank structure of labels [30–32]. In general, these algorithms for minimizing the rank of a matrix are based on nuclear norm and learn a regression model W from feature space X to label space Y directly. Therefore, we obtain the following common optimization problem:

$$\min_W \|XW - Y\|_F^2 + \|W\|_* \quad (1)$$

where $\|\cdot\|_*$ is the nuclear norm of a matrix. However, with the increasing of label dimensions, they are also faced with the expensive computational cost. To track this issue, the matrix factorization [33,34] is adopted to recover the low-rank structure, which represents the hidden structure of the labels. Meanwhile, it also captures the complex correlations among multiple labels.

Specifically, inspired by [33], label matrix $Y \in R^{n \times c}$ can be written as the product of two low-rank matrices.

$$Y = Z \times D \quad (2)$$

where $Z \in R^{n \times d}$ represents the lower-dimensional embedded space that jointly extracts the information pertaining to entire labels, while $D \in R^{d \times c}$ reflects how the initial label matrix is related to the embedded vectors.

In fact, multi-label classification usually encounters the problem of missing labels, because it is very likely for labelers to ignore the unknown labels. To end this, low-rank factorization

of matrix [34] can also provide a theoretical basis for the recovery of missing labels. At the same time, it also makes use of the global label correlations implicitly.

2.2 Hilbert–Schmidt Independence Criterion

Previous work is based on canonical correlation analysis (CCA) to describe the linear projection between feature space and label space. The kernel-based approaches consider the nonlinear correlation between two variables, which have found abroad applications, including feature selection [35], dimension reduction [36], gene selection [37] and independent component analysis [38,39]. The key to their success is that covariance and cross-covariance operators can be defined in Reproducing Kernel Hilbert Spaces (RKHS). And we may obtain an approximate statistics suitable for measuring the dependence between variables.

Specially, let \mathcal{F} be an RKHS of functions from \mathcal{X} to \mathcal{R} . To each point $x \in \mathcal{X}$, there corresponds a mapping $\phi(x) \in \mathcal{F}$, such that $\langle \phi(x), \phi(x') \rangle = k(x, x')$, where $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathcal{R}$ is a unique positive definite kernel. Likewise, let \mathcal{G} be a second RKHS on \mathcal{Y} with kernel $l(\cdot, \cdot)$ and map $\varphi(y)$. Following [40], the cross-covariance operator $C_{xy} : \mathcal{G} \rightarrow \mathcal{F}$ is defined as

$$C_{xy} = E_{xy}[(\phi(x) - \mu_x) \otimes (\varphi(y) - \mu_y)] \tag{3}$$

where $\mu_x = E_x\phi(x)$, $\mu_y = E_y\varphi(y)$, and \otimes is the tensor product. The Hilbert–Schmidt independence criterion (HSIC) [40,41] is proposed to test independence. The detailed description is shown below.

Definition 1 (HSIC) Given separable RKHSs \mathcal{F} \mathcal{G} and a joint measure P_{xy} over $\mathcal{X} \times \mathcal{Y}$, we define the Hilbert–Schmidt Independence Criterion (HSIC) as the squared HS-norm of the associated cross-covariance operator C_{xy} :

$$\begin{aligned} HSIC(P_{xy}, \mathcal{F}, \mathcal{G}) &= \|C_{xy}\|_{HS}^2 \\ &= E_{xx'yy'}[k(x, x')l(y, y')] + E_{xx'}[k(x, x')]E_{yy'}[l(y, y')] \\ &\quad - 2E_{xy}[E_{x'}[k(x, x')]E_{y'}[l(y, y')]] \end{aligned} \tag{4}$$

Here $E_{xx'yy'}$ denotes the expectation over independent pairs (x, y) and (x', y') drawn from P_{xy} .

More importantly, it is sufficient evident to suggest that HSIC is indeed a dependence criterion under all circumstances. HSIC is zero if and only if the random variables are independent. For convenience, we often utilize an empirical estimate of HSIC. The definition is as follows [40,42]:

Definition 2 (Empirical HSIC) Let $Z = \{(x_1, y_1), \dots, (x_n, y_n)\} \subseteq \mathcal{X} \times \mathcal{Y}$ be a series of n independent observations drawn from P_{xy} . The specific form is given:

$$HSIC(Z, \mathcal{F}, \mathcal{G}) = (n - 1)^{-2}tr(KHLH) \tag{5}$$

where $H, K, L \in R^{n \times n}$, $K_{i,j} = k(x_i, x_j)$, $L_{i,j} = l(x_i, x_j)$, and $H_{i,j} = \delta_{i,j} - n^{-1}$ is the centering matrix.

Obviously, the larger the value, the greater the dependence of the two variables. Empirical HSIC has advantages over existing kernel independence measurement. Unlike kernel generalised variance [38] or the canonical correlation, it only uses the trace of the product of Gram matrices without additional regularization terms for simple and good properties. In addition, HSIC has the characteristics of high speed of convergence [43].

3 The Proposed Method

In this section, we propose a feature-aware label embedding model for multi-label classification which learns a lower-dimensional embedded space connecting the feature space with the label space in Sect. 3.1. It can be easily extended to missing labels in Sect. 3.2.

3.1 Learning Embedded Space for Joint Feature and Label Embedding

Label embedding aims to seek a predictive lower-dimensional embedded space. Learning the mapping from feature space to embedded space is also much easier than learning the one to the original label space. The model applying low-rank decomposition on the label matrix can identify the hidden structure behind the labels by exploiting label correlations. Furthermore, to improve the prediction ability of embedded space, we utilize the HSIC to increase the dependence between feature space and label space. With the aid of HSIC, we can take full advantage of the information of instance features to support the label space embedding process.

Label space embedding To capture the label correlations and deal with the high dimensional labels, we employ the low-rank structure of label matrix to find a more compact and abstract latent vectors representation. Specifically, we use Eq. (2) to decompose the label matrix Y to two low-rank matrices Z and D , following PLST [11], CPLST [12] and MLC-BMaD [44]. Then the sub-objective is given via minimizing the reconstruction error of the embedded space returned from original space:

$$\varphi(Z, D) = \min_{Z, D} \|Y - ZD\|_F^2 \quad (6)$$

where $\|\cdot\|_F$ is the *Frobenius norm* of a matrix, Z represents embedded space, also known as code matrix and D is the decoding matrix reflecting the correlation between original labels and embedded vectors.

Feature space embedding A number of label embedding methods choose to learn a linear mapping function from the instance matrix X to the code matrix Z . Intuitively, this seems reasonable since we really need to train a regression model. However, this may over-fit the training data set, thus degrading the performance of classification.

In order to exploit instance information efficiently, and improve the prediction ability of the embedded space, it is necessary to retain the dependence between the instance matrix X and the code matrix Z rather than simply learning a linear model between them. There are many criteria to measure their relationship. Due to its simplicity and neat theoretical properties, the HSIC is adopted. The expression is formulated as follows:

$$\phi(Z, X) = \max_Z (n-1)^{-2} \text{Tr}(KHLH) \quad (7)$$

where $H, K, L \in R^{n \times n}$, K and L are kernel matrices with the instance kernel $K_{i,j} = k(x_i, x_j)$ and the label kernel $L_{i,j} = l(x_i, x_j)$, and $H_{i,j} = \delta_{i,j} - \frac{1}{n}$ is the centering matrix.

For the sake of convenience, we utilize the linear kernels for $L: L = ZZ^T$. With $(n-1)^{-2}$ a constant, thus Eq. (7) is equivalent to:

$$\phi(Z, X) = \max_Z \text{Tr}(KHZZ^TH) \quad (8)$$

Obviously, it can be observed that the HSIC criteria makes full use of instances information through the instance kernel K . In general, the most efficient way to set K chooses the RBF

kernel, which is widely used and of great competence [45]. A RBF kernel is denoted by $K(x, x')$ on instances.

$$K(x, x') = \exp(-\gamma \|x - x'\|^2) \tag{9}$$

where $\gamma > 0$ is a kernel parameter.

The model on embedded space is built by minimizing the reconstruction error and maximizing the dependence between feature space and embedded space. Combining Eqs. (6) and (8), we gain the following optimization problem:

$$L(Z, D) = \min_{Z, D} \|Y - ZD\|_F^2 - \alpha \text{Tr}(KHZZ^T H) \tag{10}$$

where α is a trade-off parameter for controlling the impact between feature embedding and label embedding. The Z obtained not only has strong recovery ability, but also has good prediction ability.

3.2 Learning Embedded Space with Missing Labels

The label matrix is generally incomplete in the real word applications. It is unreliable to learn the classifier directly using the missing label matrix. What's more, different labels are typically not independent but inherently correlated. Hence, it is difficult to exploit the label correlations with the missing labels. In this section, the proposed method is extended to deal with missing labels.

Recovering the missing labels As is mentioned in Sect. 2.1, the low-rank decomposition on the label matrix plays a significant role in recovering missing labels. Specifically, we denote the original labels matrix by $Y = \{y_1, \dots, y_n\}^T \in \{0, \frac{1}{2}, 1\}^{n \times c}$, where $y_{ij} = 1$ or 0 indicates the j -th label is observed for the data point x_i while $y_{ij} = \frac{1}{2}$ indicates the label is missing. The ground-truth label matrix is denoted by $\tilde{Y} = \{\tilde{y}_1, \dots, \tilde{y}_n\}^T$. Assume that $\Omega \subset \{(i, j) : 1 \leq i \leq n, 1 \leq j \leq c\}$ is the set of observed label indicators, excluding missing labels. Ω^c is the set of missing labels. We propose to solve the following low-rank factorization model based on Eq. (6).

$$\begin{aligned} \min_{\tilde{Y}, Z, D} & \|\tilde{Y} - ZD\|_F^2 \\ \text{s.t.} & \tilde{Y}_{i,j} = Y_{i,j}, (i, j) \in \Omega \end{aligned} \tag{11}$$

This formula can guarantee that the observed labels remain unchanged except for the missing labels, which is obvious difference from the previous method. Building off the label correlation from global aspect, we can go one step further and preserve the label structure from local way. A smoothness assumption is usually adopted that the distance of two instances in their feature space can measure the similarities of their corresponding labels. In other words, if two samples x_i and x_j are more closer in the intrinsic geometry of the feature distribution, the recovered labels of them are also more closer to each other in the label space, and vice versa. The manifold regularizer can be defined as

$$\sum_{i,j} \frac{1}{2} \omega_{i,j} \|\tilde{y}_i - \tilde{y}_j\|^2 = \text{tr}(\tilde{Y}^T L_0 \tilde{Y}) \tag{12}$$

where $L_0 = D - W$ is the Laplacian matrix, and D is a diagonal matrix with $D_{ii} = \sum_{j=1}^n \omega_{i,j}$. W denotes the sample similarity matrix by the heat kernel function, which is

given:

$$\omega_{i,j} = \begin{cases} \exp\left(-\frac{\|x_i-x_j\|^2}{\sigma^2}\right), & x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i) \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where σ is the Gaussian function variance, $N_k(x_i)$ is the k nearest neighbors set of x_i .

Assume that the label correlation is available, we can also incorporate the information by adding another Laplacian regularizer. Here, cosine similarity is used for defining the weight matrix V among class, as follows:

$$v_{i,j} = \frac{\langle y_i, y_j \rangle}{\|y_i\| \|y_j\|} \tag{14}$$

Similar to the Eq. (12), the label manifold regularizer is formulated as:

$$\sum_{i,j} \frac{1}{2} v_{i,j} \|\tilde{y}_i^T - \tilde{y}_j^T\|^2 = \text{tr}(\tilde{Y}L_1\tilde{Y}^T) \tag{15}$$

where $L_1 = G - V$ is the Laplacian matrix, and G is a diagonal matrix with $G_{ii} = \sum_{j=1}^n v_{i,j}$.

Considering all the above discussions, the optimization objective function to recover missing labels becomes:

$$\begin{aligned} \min_{\tilde{Y}, Z, D} & \|\tilde{Y} - ZD\|_F^2 + \beta \text{Tr}(\tilde{Y}^T L_0 \tilde{Y}) + \eta \text{Tr}(\tilde{Y}L_1\tilde{Y}^T) \\ \text{s.t.} & \tilde{Y}_{i,j} = Y_{i,j}, (i, j) \in \Omega \end{aligned} \tag{16}$$

where β, η are regularization parameters, which control the influence between the sample-level correlation and class-level correlation.

Label embedding with missing labels Our goal is to find the low dimensional embedded space and recover the missing labels at the same time. It can be seen that $\|\tilde{Y} - ZD\|_F^2$ represents the label space embedding from Sect. 3.1. Therefore, we should take feature space embedding into consideration in obtaining embedded space with the missing labels. Combining the Eqs. (16) and (8), we obtain the optimization problem:

$$\begin{aligned} \min_{\tilde{Y}, Z, D} & \|\tilde{Y} - ZD\|_F^2 + \beta \text{Tr}(\tilde{Y}^T L_0 \tilde{Y}) + \eta \text{Tr}(\tilde{Y}L_1\tilde{Y}^T) - \alpha \text{Tr}(KHZZ^T H) \\ \text{s.t.} & \tilde{Y}_{i,j} = Y_{i,j}, (i, j) \in \Omega \end{aligned} \tag{17}$$

For the sake of calculation, we define $P_\Omega(X)$ as the orthogonal projection operator on set Ω of the matrix X :

$$(P_\Omega(X))_{i,j} = \begin{cases} X_{i,j} & (i, j) \in \Omega \\ 0 & (i, j) \in \Omega^c \end{cases} \tag{18}$$

The objective function can be rewritten:

$$\begin{aligned} \min_{\tilde{Y}, Z, D} & \|\tilde{Y} - ZD\|_F^2 + \beta \text{Tr}(\tilde{Y}^T L_0 \tilde{Y}) + \eta \text{Tr}(\tilde{Y}L_1\tilde{Y}^T) - \alpha \text{Tr}(KHZZ^T H) \\ \text{s.t.} & P_\Omega(\tilde{Y}) = P_\Omega(Y) \end{aligned} \tag{19}$$

The objective function has two main clear interpretations as follows:

- (1) The first three terms are used to recover the missing labels.

- (2) The first term and the fourth term aim at learning the embedded space, which can efficiently deal with the curse of dimensionality problems.

In conclusion, our proposed method joints the embedded space learning and the missing matrix recovery. To recover the missing labels, we propagates the semantic information from feature space to label space via Laplace manifold regularization. By utilizing low-rank decomposition, the label correlations can be efficiently captured.

4 Optimization

4.1 Optimization Algorithm for Complete Labels

An efficient alternating algorithm is proposed to optimize the objective function (10). To calculate D first, problem reduces to:

$$L(D) = \min_D \|Y - ZD\|_F^2 \tag{20}$$

The optimal D can be obtained by taking the derivative of $L(D)$ with respect to D and setting it to 0:

$$\frac{\partial L(D)}{\partial D} = Z^T(ZD - Y) = 0 \tag{21}$$

We obtain the closed-form expression:

$$D = (Z^T Z)^{-1} Z^T Y \tag{22}$$

To eliminate redundant information in the embedded space and then make the decode process more compactly, we assume that dimensions of Z are uncorrelated and thus orthonormal. That is, $Z^T Z = I$. Thus Eq.(22) can be reformulated as:

$$D = Z^T Y \tag{23}$$

With D derived, the final optimization objective function is transformed as:

$$\begin{aligned} L(Z) &= \max_Z Tr(Z^T Y Y^T Z) + \alpha Tr(Z^T H K H Z) \\ &= \max_Z Tr(Z^T (Y Y^T + \alpha H K H) Z) \\ s.t. & Z Z^T = I \end{aligned} \tag{24}$$

The optimization problem (24) can be easily solved by eigenvalue decomposition. In the end, Z is obtained by concatenating the normalized eigenvectors corresponding to the top k largest eigenvalues $\lambda_i (i = 1, \dots, k)$ of $A = Y Y^T + \alpha(H K H)$. The optimal value of (24) is $\sum_{i=1}^k \lambda_i$.

4.2 Optimization Algorithm for Missing Labels

The whole optimization problem (19) is reduced to several simpler subproblems that are easier to solve.

- Updating D with others fixed

By ignoring the irrelevant terms with respect to D , the problem turns into:

$$\min_D \|\tilde{Y} - ZD\|_F^2 \tag{25}$$

It is obvious to see that the form is the same as Eq. (20). Therefore the solution to D is $D = Z^T \tilde{Y}$.

- Updating Z with others fixed

Similarly, Z is the only variable and the problem can then be rewritten as:

$$\min_Z \left\| \tilde{Y} - ZD \right\|_F^2 - \alpha \text{Tr}(KHZZ^T H) \tag{26}$$

As analysed in Sect. 4.1, Z can be obtained by Eq. (24).

- Updating \tilde{Y} with others fixed

Given D and Z , the problem reduces to:

$$\begin{aligned} \min_{\tilde{Y}, Z, D} & \left\| \tilde{Y} - ZD \right\|_F^2 + \beta \text{Tr}(\tilde{Y}^T L_0 \tilde{Y}) + \eta \text{Tr}(\tilde{Y} L_1 \tilde{Y}^T) \\ \text{s.t.} & P_\Omega(\tilde{Y}) = P_\Omega(Y) \end{aligned} \tag{27}$$

To track the problem, a Lagrangian multiplier $\Lambda \in \mathcal{R}^{n \times c}$ is introduced. The Lagrangian function of Eq. (27) is defined as

$$\left\| \tilde{Y} - ZD \right\|_F^2 + \beta \text{Tr}(\tilde{Y}^T L_0 Y) + \eta \text{Tr}(\tilde{Y} L_1 \tilde{Y}^T) + \Lambda \cdot P_\Omega(\tilde{Y} - Y) \tag{28}$$

In order to get the optimal solution to \tilde{Y} , the corresponding subproblems of $P_\Omega(\tilde{Y})$ and $P_{\Omega^c}(\tilde{Y})$ need to be determined respectively. The equation with respect to $P_\Omega(\tilde{Y})$ is as follows:

$$\left\| P_\Omega(\tilde{Y} - ZD) \right\|_F^2 + \beta \text{Tr}(P_\Omega(\tilde{Y}^T L_0 Y)) + \eta \text{Tr}(P_\Omega(\tilde{Y} L_1 \tilde{Y}^T)) + \Lambda \cdot P_\Omega(\tilde{Y} - Y) \tag{29}$$

Taking the derivative with respect to Λ and \tilde{Y} , and setting those to zero, we obtain:

$$P_\Omega(\tilde{Y} - Y) = 0 \tag{30}$$

$$P_\Omega((\tilde{Y} - ZD) + \beta L_0 \tilde{Y} + \eta \tilde{Y} L_1) = \Lambda \tag{31}$$

The subproblem with respect to $P_{\Omega^c}(\tilde{Y})$ is formulated as follows:

$$\left\| P_{\Omega^c}(\tilde{Y} - ZD) \right\|_F^2 + \beta \text{Tr}(P_{\Omega^c}(\tilde{Y}^T L_0 Y)) + \eta \text{Tr}(P_{\Omega^c}(\tilde{Y} L_1 \tilde{Y}^T)) \tag{32}$$

Similarly, we obtain:

$$P_{\Omega^c}((\tilde{Y} - ZD) + \beta L_0 \tilde{Y} + \eta \tilde{Y} L_1) = 0 \tag{33}$$

Thus, according to the Eqs. (30) and (33), we gain the solution to \tilde{Y} :

$$P_\Omega(\tilde{Y}) = P_\Omega(Y) \tag{34}$$

$$P_{\Omega^c}(\tilde{Y}) = P_{\Omega^c}((I + \beta L_0)^{-1} ZD(I + \eta L_1)^{-1}) \tag{35}$$

Please refer to the method [46] to solve Eq. (33). With the updates about D , Z and \tilde{Y} being in closed form, the proposed LEDM is given in Algorithm 1, which contains the full labels and missing labels situations. Learning the mapping for the classifiers from feature space X to lower-dimensional embedded space Z is much easier than learning the one to the original high-dimensional label space. In the end, the forecasting results are decoded to the initial label space. When we reconstruct the predicted label vectors through step 17, the results may contain non-binary values. As a consequence, a threshold requires to be selected to determine whether the values belong to the class. The fixed value of 0.5 is a simple and direct approach

Algorithm 1 Label Embedding via Dependence Maximization.**Input:**

Data matrix X , Label matrix Y , Embedded dimensions K .
Parameters α, β, η .

Output:

The ground-truth label matrix \tilde{Y} , Predicted label of test data x_t .

```

1: if  $Y$  does not contain missing labels then
2:   calculate embedded matrix  $Z$  according to Eq. (24);
3:   calculate decoding matrix  $D$  according to Eq. (23);
4: else
5:   Initialize:  $\tilde{Y} = Y$ ;
6:   repeat
7:     fix  $\tilde{Y}$ ,  $D$  and update  $Z$  by step 2;
8:     fix  $\tilde{Y}$ ,  $Z$  and update  $D$  by step 3;
9:     fix  $Z$ ,  $D$  and update  $\tilde{Y}$  according to Eqs. (34) and (35);
10:  until convergence;
11: end if
12: Train:
13: Learning a multi-dimension regression model  $f(x)$  from  $\{(x_i, z_i)\}_{i=1}^n: f(x_i) \rightarrow z_i$ ;
14: Prediction:
15: Given a test data  $x_t$ 
16: Obtain the embedded vector  $z_t$  of test data  $x_t: z_t = f(x_t)$ 
17: Predict the label of test data  $x_t: h(x_t) = \text{round}(f(x_t)D)$ .

```

[11]. To boost the classification performance, an adaptive threshold is adopted via maximizing evaluation criteria from training data in this paper, similar to [47,48]. Specifically, sort the prediction values in a descending manner and find the best split point (threshold) to achieve high performance. If more than the threshold, the value is assigned with 1, otherwise 0.

4.3 Computational Complexity Analysis

In this subsection, we make a computational complexity analysis of optimization given in Eq. (19). For each iteration in Algorithm 1, when updating Z in the step 7, we define $A = \tilde{Y}\tilde{Y}^T + \alpha(HKH)$. The optimal solution to Eq. (24) is the eigenvectors of A corresponding to the top k largest eigenvalues. We just need to directly calculate A and then perform an eigenvalue decomposition on it. The computational complexity of deriving A is $O(cn^2)$. Considering that $d < n$, and A is a real symmetric matrix, the eigenvalue problem *w.r.t* A can also be solved efficiently using iterative methods like Arnoldi iteration [49], which can achieve an optimal computational complexity of $O(nd^2)$. Thus updating Z requires $O(cn^2 + nd^2)$. According to Eq. (23), the update of D takes $O(ndc)$. To calculate \tilde{Y} , we need to solve the matrix equation Eq. (33). A efficient algorithm is proposed to solve the pairwise constraint propagation problem [46]. The computation of updating \tilde{Y} requires $O(n^2)$. Finally, the overall computational complexity required by LEDM is $O(cn^2 + nd^2 + ndc)$.

5 Experiments

To validate the proposed method, a large number of experiments have been conducted on datasets. Performance on both the complete-label case and the missing-label case are discussed in this section.

Table 1 The basic characteristics of the data sets

Data set	#Instances n	#Features m	#Labels c	Type	LC
Emotions	593	72	6	Music	1.869
Yeast	2417	103	14	Biology	4.237
Medical	978	1449	45	Text	1.245
Scene	2407	294	6	Images	1.074
Enron	1702	1001	53	Text	3.378
CAL500	502	68	174	Music	26.044
Langlog	1460	1004	75	Images	1.18
Corel5k	5000	499	374	Images	3.522

5.1 Learning with Complete Labels

In the first experiment, we focus on the situation that there are no missing elements in the label matrix.

5.1.1 Datasets

We choose eight public-available multi-label datasets to assess the effectiveness of our method from diverse fields. The characteristics of the datasets are shown in detail by Table 1. For each dataset, let #Instances n , #Features m , and #Labels c denote the number of instances, features and possible labels respectively. LC represents label cardinality reflecting the average number of label per instance.

5.1.2 Evaluation Metrics

The performance evaluation is a pivotal factor in comparison of various models. Given a test dataset $D = \{(x_i, y_i) | 1 \leq i \leq n\}$, where $y_i \in \{0, 1\}^c$ denotes the ground truth labels with the i -th test sample, and \hat{y}_i represents its predicted set of labels. We have adopted three evaluation criteria extensively used in multi-label classification, namely *Rank Loss*, *Average Precision* and *Macro F_1* [50,51].

- *Rank Loss*:

$$Rloss = \frac{1}{n} \sum_{i=1}^n \frac{|Q_i|}{|y_i^+| |y_i^-|} \tag{36}$$

where $Q_i = \{(y', y'') | f(x_i, y') \leq f(x_i, y''), (y', y'') \in y_i^+ \times y_i^-\}$. Let y_i^+ and y_i^- be respectively the sets of positive and negative labels associated with the i -th instances. *Rank Loss* evaluates the proportion of misordered label pairs.

- *Average Precision*:

$$AP = \frac{1}{n} \sum_{i=1}^n \frac{1}{|y_i^+|} \sum_{y \in y_i^+} \frac{|\{y' | rank_f(x_i, y') \leq rank_f(x_i, y), y' \in y_i^+\}|}{rank_f(x_i, y)} \tag{37}$$

where $rank_f(x_i, y')$ stands for the ranking of label y' in the label set of x' predicted by the multi-label classifier f . This criteria evaluates the average proportion of relevant labels ranked higher than a particular label $y \in y_i^+$.

- *Macro F₁*:

$$\begin{aligned} \text{Macro } F_1 &= \frac{1}{c} \sum_{i=1}^c \frac{2p_i r_i}{p_i + r_i} \\ \text{s.t. } &= \frac{TP}{TP+FP}, \quad r_i = \frac{TP}{TP+FN} \end{aligned} \quad (38)$$

where TP , FP and FN represent respectively the number of true positive, false positive and false negative with respect to a specific label. For label-based criteria, *Macro F₁* is the harmonic mean of recall r and precision p .

It is evident that their values vary from 0 to 1. For *rank loss*, the smaller the value, the better the generalization performance. Whereas for *average precision* and *macro F₁*, the larger the value, the better the performance.

5.1.3 Baselines

In this experiment, the proposed method is compared with five representative label reduction algorithms, including Principle Label Space Transformation (PLST) [11], Conditional Principal Label Space Transformation (CPLST) [12], Label Compression Coding through Maximizing Dependence (LCCMD) [52], Cost-sensitive Label Embedding with Multidimensional Scaling (CLEMS) [13] and End-to-End Feature-aware label space Encoding (E²FE) [14].

For all the algorithms, the Random Forest is used to learn the predictive model between feature space and embedded space. The model parameters such as the maximum depth of the trees and the number of the estimators are selected respectively from 5, 10, ..., 35 to 2, 4, ..., 40 via grid search. In our experiment, each dataset is conducted 5-fold cross validation, taking four part for train and the rest for test. Following the preprocessing steps of baselines, we convert the mean value of feature vectors to be zero and the variance to be one. In general, α in the proposed method, as well as α in E²FE require to be determined in advance. We adjust these parameters via cross-validation to achieve the best results. The remaining parameters in other methods are set to the values suggested in the original papers.

5.1.4 Results

All comparable algorithms are performed with respect to the evaluation metrics versus the different values of K/M on eight datasets, where K and M are the dimensions of the embedded space and initial label space respectively. The specific experimental results are shown in Figs. 1, 2 and 3 in which the abscissa axis represents the ratio of the embedded space dimensions (K/M). The parameters in the following algorithms are set to the values suggested in the original papers. As the ratio increases, all label embedding methods achieve better performance on account of the better preservation of label information. As can be seen, LEDM consistently and remarkably outperforms the existing approaches in most cases, which confirms its effectiveness.

There are some interesting phenomenon observed from the Figs. 1, 2 and 3 as follows. (1) To reflect the importance of label space embedding in the first component, our model degenerates into LCCMD. LCCMD is similar to the second part of our method, which only considers the dependence maximization. We can clearly observe that the curves of LEDM are higher than that of LCCMD in terms of three criteria in almost all cases. (2) Compared to the E²FE method based on CCA, which measures the linear relationship between feature space and latent space, HSIC adopted to capture the nonlinear correlations in our method

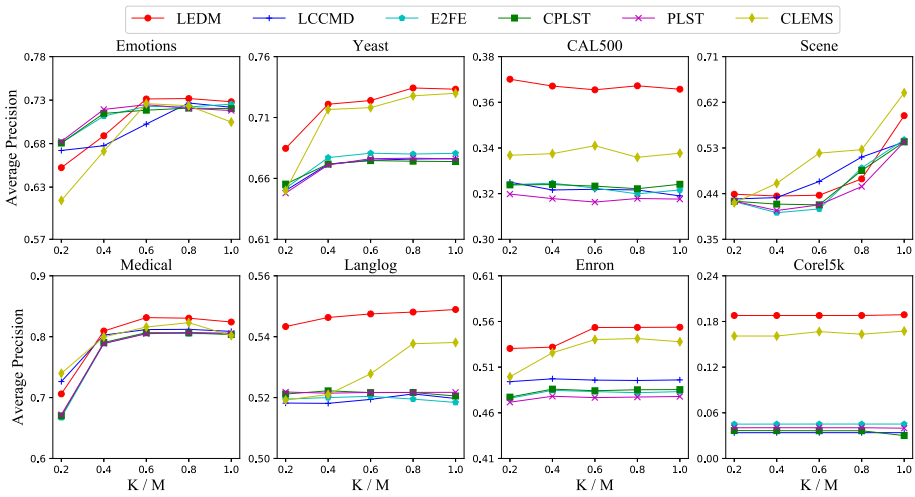


Fig. 1 Average precision with different embedded space dimension ratios (K/M)

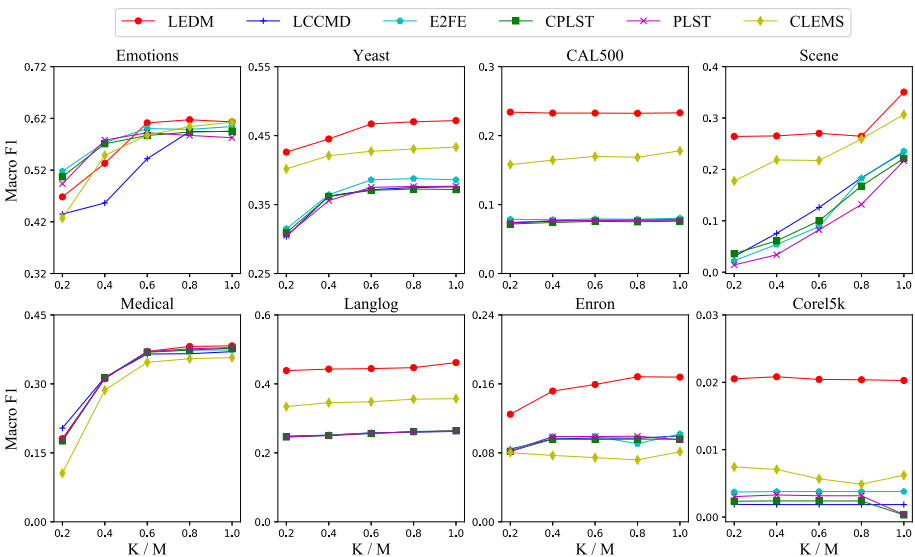


Fig. 2 Macro F1 with different embedded space dimension ratios (K/M)

is significantly better. This also verifies the benefits of choosing HSIC. (3) CLEMS does not take into account the feature space when learning latent space, which may result in poor predictive ability. (4) In spite of introducing feature information for CPLST, PLST seems to be slightly superior on dataset Yeast. The reason might be that the linear mapping from feature space to latent space overfit the training data. (5) We can utilize the much lower dimensional embedded space to preserve the structure information of the original label space by applying low-rank decomposition on the label matrix. Similar to E²FE, as a result of the orthonormality constraint on embedded space, the performance varies a little with increasing K. However, CLEMS may decrease dramatically sometimes. For instance, the

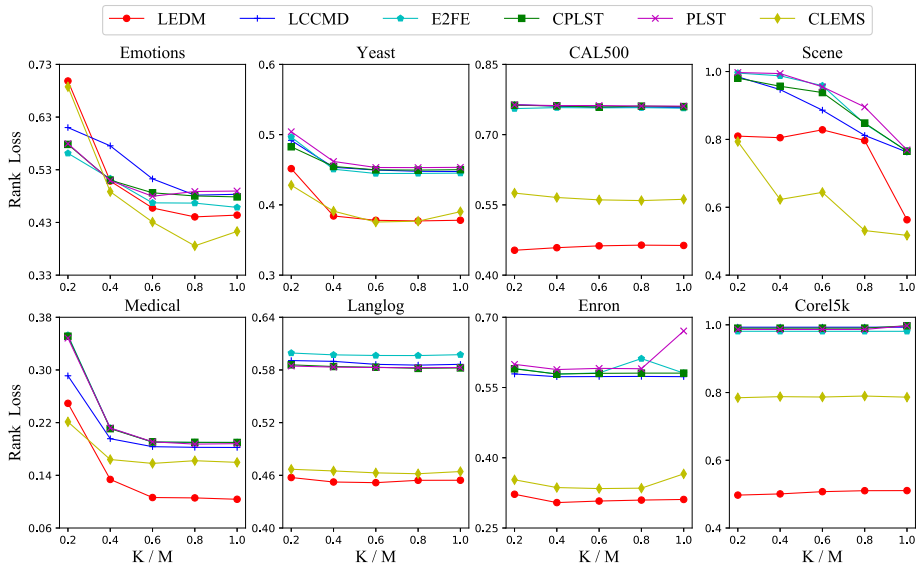


Fig. 3 Rank loss with different embedded space dimension ratios (K/M)

average precision of CLEMS drops from 0.6387 to 0.4217 when the ratio of the embedded space dimensions varies from 1 to 0.2 on Scene dataset in Fig. 1.

The experimental results confirm that utilizing the low-rank decomposition on the label matrix for reconstruction error of labels and using HSIC to maximize the dependence between feature space and embedded space are an effective paradigm. Furthermore, the model employs the implicit encoding in learning embedded space representation, instead of requiring an explicit encoding function like PLST, CPLST and LCCMD.

To study the influence of α , the experiments on Enron and Langlog datasets in terms of three different criteria are conducted. The trade-off parameter α controls the impact between feature embedding and label embedding. The variation of LEDM with the parameter α is showed in Fig. 4. On the whole, the α is not very sensitive to the proposed approach within limits.

5.2 Learning with Missing Labels

In this section, the experiment with full labels will be extended to deal with missing labels. The datasets and evaluation metrics are identical with those described in Sect. 5.1. The recovery of missing labels and the prediction of unknown data have been discussed in detail.

5.2.1 Experiments Setting

To demonstrate the effectiveness of our method in handling with missing labels, we randomly select a certain proportion of elements as missing labels removed from the original label matrix. Several existing multi-label classification algorithms which can address the missing labels problems will be contrasted with proposed method, including Binary Relevance (BR) [5], Multi-label Learning with Missing Labels (MLML) [21] and Low Rank multi-label classification with Missing Labels (LRML) [27]. Due to the presence of the missing labels,

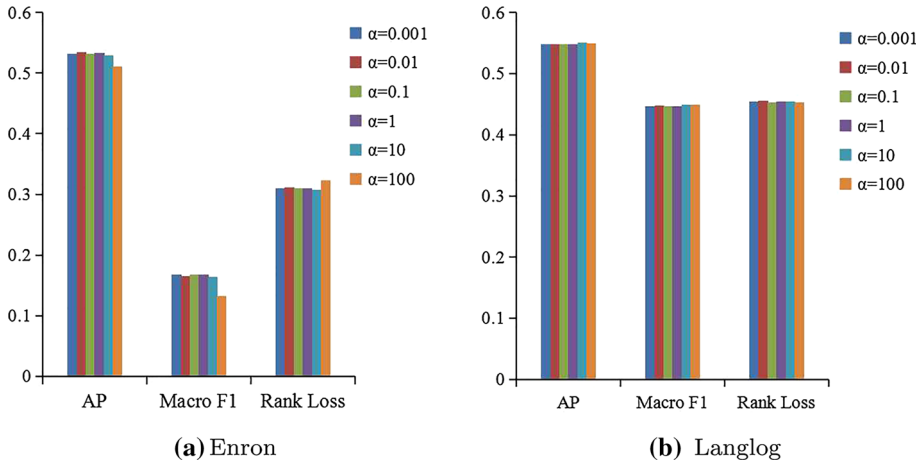


Fig. 4 Varying α on Enron and Langlog datasets respectively

BR can not directly learn the classifier. We regard missing labels as negative labels to train in BR.

For BR, MLML and proposed LEDM, the Random Forest is used for training model. In general, the trade-off parameters β and η in the proposed method, the parameters λ_x and λ_c for MLML, as well as α , β and γ in LRML are tuned by cross-validation to obtain the better results as much as possible.

5.2.2 Results

The results on both missing label recovery and test data prediction with respect to different evaluation criteria are respectively shown in Tables 2, 3, 4, 5, 6 and 7, in which ρ denotes missing label ratio. The best performance among all the algorithms being compared is highlighted in boldface. On the whole, the more the labels are observed, the better the performance is. This is in accordance with the fact that more label information is utilized sufficiently in the process of classification.

Whether to recover missing labels of training data or to predict unknown data, it is quite obvious that LEDM yields the best performance in most cases in terms of three criteria. The reason for its success is the simultaneous optimization of the low-rank decomposition on the label matrix, manifold regularization based on instance-level and class-level and feature embedding via HSIC. The proposed LEDM not only recovers the missing labels, but also learns the embedded space by utilizing label correlation and feature space correlation. However, it is slightly worse on the datasets Enron and Scene. It is possible that the non-convex optimization in the low-rank decomposition model may get stuck in local minimum.

As is shown in Tables 2, 3 and 4, LEDM, MLML and LRML that are used for dealing with missing labels in different manners outperform BR in most cases, which demonstrates the necessity to handle missing labels. Moreover, BR treats missing labels as negative labels in multi-label classification. This is mainly due to the presence of a large enough number of negative labels for each sample. The low rank decomposition model adopted can make sure that except for the missing labels, the observed labels keep invariability in true label matrix \tilde{Y} , leading to significant performance improvements.

Table 2 Recovery results for missing label on average precision

Data set	ρ	LEDM	MLML	LRML	BR
<i>Average precision</i>					
Emotion	0.3	0.921	0.920	0.843	0.886
	0.7	0.838	0.846	0.772	0.781
Yeast	0.3	0.893	0.891	0.792	0.869
	0.7	0.811	0.805	0.716	0.738
CAL500	0.3	0.829	0.791	0.784	0.816
	0.7	0.653	0.633	0.652	0.623
Scene	0.3	0.883	0.926	0.884	0.853
	0.7	0.764	0.845	0.785	0.711
Medical	0.3	0.788	0.776	0.772	0.766
	0.7	0.601	0.588	0.531	0.542
Langlog	0.3	0.855	0.811	0.852	0.827
	0.7	0.707	0.667	0.704	0.643
Enron	0.3	0.777	0.823	0.764	0.795
	0.7	0.613	0.667	0.543	0.585
Corel5k	0.3	0.750	0.735	0.742	0.733
	0.7	0.515	0.510	0.513	0.511

Table 3 Recovery results for missing label on Macro F1

Data set	ρ	LEDM	MLML	LRML	BR
<i>Macro F1</i>					
Emotion	0.3	0.881	0.905	0.857	0.847
	0.7	0.781	0.801	0.814	0.666
Yeast	0.3	0.885	0.874	0.805	0.85
	0.7	0.746	0.738	0.717	0.66
CAL500	0.3	0.845	0.776	0.755	0.849
	0.7	0.663	0.633	0.643	0.657
Scene	0.3	0.882	0.898	0.825	0.852
	0.7	0.737	0.778	0.756	0.663
Medical	0.3	0.803	0.783	0.755	0.796
	0.7	0.612	0.593	0.565	0.557
Langlog	0.3	0.858	0.856	0.855	0.85
	0.7	0.685	0.683	0.682	0.663
Enron	0.3	0.825	0.822	0.754	0.822
	0.7	0.643	0.648	0.63	0.651
Corel5k	0.3	0.818	0.785	0.801	0.813
	0.7	0.632	0.605	0.614	0.612

Our method has advantages over MLML, LRML and BR remarkably in test data prediction according to the experimental results shown in Tables 5, 6 and 7. This is due to the fact that LEDM learns the embedded space via dependence maximization at the same time while recovering the missing labels. More importantly, it is much easier and faster for a base

Table 4 Recovery results for missing label on rank loss

Data set	ρ	LEDM	MLML	LRML	BR
<i>Rank loss</i>					
Emotion	0.3	0.131	0.133	0.229	0.262
	0.7	0.268	0.271	0.324	0.498
Yeast	0.3	0.115	0.138	0.209	0.262
	0.7	0.224	0.258	0.288	0.502
CAL500	0.3	0.128	0.239	0.249	0.259
	0.7	0.251	0.394	0.339	0.503
Scene	0.3	0.137	0.132	0.18	0.258
	0.7	0.262	0.272	0.245	0.507
Medical	0.3	0.107	0.254	0.155	0.257
	0.7	0.208	0.444	0.205	0.505
Langlog	0.3	0.199	0.233	0.204	0.251
	0.7	0.334	0.339	0.342	0.493
Enron	0.3	0.116	0.191	0.213	0.257
	0.7	0.22	0.363	0.291	0.513
Corel5k	0.3	0.103	0.271	0.178	0.257
	0.7	0.233	0.496	0.356	0.503

Table 5 Performance of algorithms on average precision

Data set	ρ	LEDM	MLML	LRML	BR
<i>Average precision</i>					
Emotion	0.3	0.731	0.716	0.729	0.645
	0.7	0.708	0.687	0.682	0.568
Yeast	0.3	0.71	0.654	0.441	0.573
	0.7	0.704	0.648	0.44	0.45
CAL500	0.3	0.358	0.31	0.202	0.248
	0.7	0.344	0.307	0.197	0.175
Scene	0.3	0.48	0.542	0.539	0.499
	0.7	0.467	0.537	0.528	0.451
Medical	0.3	0.809	0.734	0.775	0.289
	0.7	0.733	0.685	0.721	0.112
Langlog	0.3	0.53	0.419	0.41	0.469
	0.7	0.51	0.53	0.399	0.309
Enron	0.3	0.441	0.395	0.348	0.296
	0.7	0.442	0.346	0.252	0.197
Corel5k	0.3	0.057	0.027	0.027	0.027
	0.7	0.055	0.025	0.026	0.023

Table 6 Performance of algorithms on Macro F1

Data set	ρ	LEDM	MLML	LRML	BR
<i>Macro F1</i>					
Emotion	0.3	0.614	0.576	0.616	0.389
	0.7	0.585	0.523	0.61	0.16
Yeast	0.3	0.462	0.334	0.367	0.234
	0.7	0.456	0.313	0.366	0.062
CAL500	0.3	0.232	0.073	0.212	0.046
	0.7	0.231	0.066	0.205	0.015
Scene	0.3	0.292	0.213	0.355	0.149
	0.7	0.299	0.198	0.346	0.052
Medical	0.3	0.357	0.311	0.325	0.098
	0.7	0.303	0.258	0.287	0.01
Langlog	0.3	0.422	0.164	0.389	0.185
	0.7	0.421	0.232	0.374	0.059
Enron	0.3	0.126	0.065	0.12	0.042
	0.7	0.122	0.051	0.18	0.013
Corel5k	0.3	0.025	0.018	0.020	0.009
	0.7	0.02	0.015	0.016	0.006

classifier to yield a high performance from feature space to the dense, real-valued, lower-dimensional embedded space than that to the sparse, binary-valued, higher-dimensional original label space. There is a great deal of imbalanced data with a disproportionate number of instances in the classes. For example, there is over 300 labels with less than 10 positive labels for each sample on dataset Corel5k. As a result, it is difficult for BR to obtain a better classification performances without utilizing the label relationship. MLML focuses on recovering the missing labels rather than classification. Consequently, the prediction results are not competitive with LRML and LEDM. LRML directly learns the latent semantics of the label space by taking advantage of low-rank mapping. However, it's worth noting that feature space correlation is incorporated via HSIC to obtain embedded space in proposed LEDM, which further improve the predictive ability.

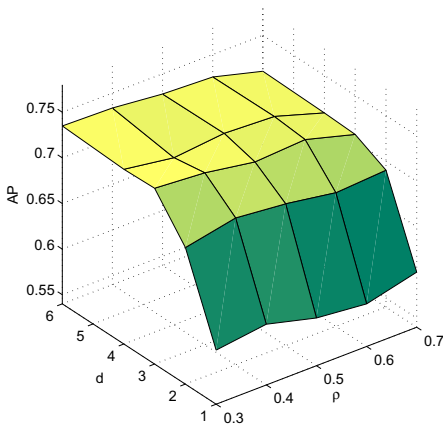
5.3 Impact of Missing Labels

To explore the impact of missing label ratio on proposed method, the experiments on Emotions and Medical datasets in terms of AP and Macro F1 are conducted. The missing label ratio ρ varies from 0.3 to 0.7. The results under different missing label ratio ρ and different embedded dimensions d are showed in Figs. 5 and 6, where axis ρ and d denote missing label ratio and the dimensions of the embedded space, respectively.

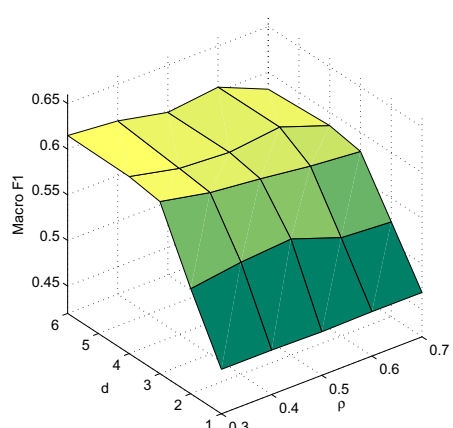
From the Figs. 5 and 6, it can be clearly seen that the classification performance of LEDM is relatively stable for different ρ on Emotions and Medical. This also indicate that the proposed model is robust to the multi-label data. Whereas, with the reduction of embedded dimensions d , the performance gets worse. This is possibly due to the fact that the embedded space can not capture enough label information using lower dimensions to some extent. Hence, it is necessary for each dataset to choose appropriate embedded dimensions.

Table 7 Performance of algorithms on rank loss

Data set	ρ	LEDM	MLML	LRML	BR
<i>Rank loss</i>					
Emotion	0.3	0.445	0.509	0.441	0.731
	0.7	0.463	0.561	0.453	0.915
Yeast	0.3	0.382	0.484	0.709	0.679
	0.7	0.391	0.496	0.722	0.933
CAL500	0.3	0.471	0.577	0.518	0.885
	0.7	0.484	0.584	0.528	0.969
Scene	0.3	0.735	0.792	0.514	0.885
	0.7	0.716	0.812	0.531	0.965
Medical	0.3	0.257	0.476	0.302	0.785
	0.7	0.303	0.529	0.347	0.977
Langlog	0.3	0.497	0.716	0.629	0.689
	0.7	0.518	0.561	0.625	0.887
Enron	0.3	0.357	0.5	0.548	0.835
	0.7	0.382	0.569	0.771	0.967
Corel5k	0.3	0.537	0.745	0.697	0.889
	0.7	0.544	0.783	0.721	0.903



(a) Average Precision on Emotions



(b) Macro F1 on Emotions

Fig. 5 Performance on emotions with different missing label ratio ρ

5.4 Sensitivity to Parameters

To study the influence of manifold regularization parameters β and η , the experiments on Yeast dataset in terms of AP and Macro F1 are conducted. The trade-off parameters β and η control the impact between the sample-level correlation and class-level correlation. The larger β is, the more important the sample-level correlation is. It is also similar to η . Figures 7 and 8 report respectively the sensitivity analysis of LEDM with respect to β and η . When $\beta = 0$, class-level correlation is only considered on recovering missing labels.

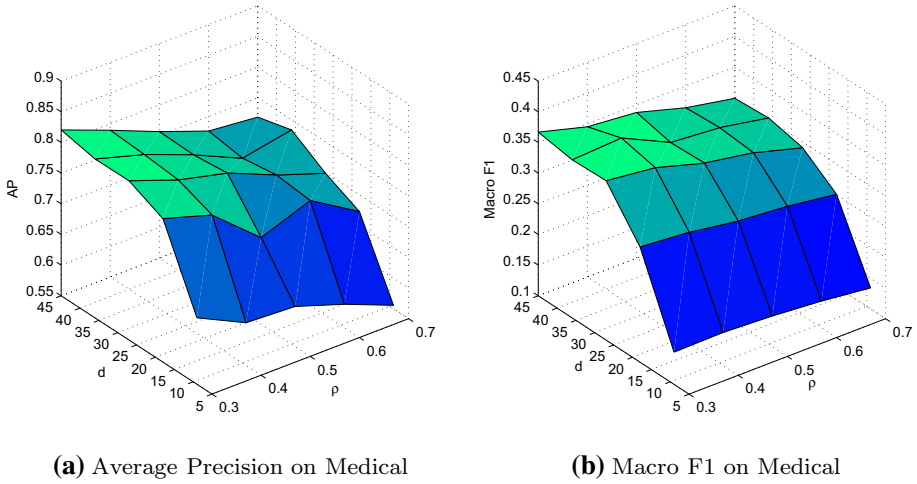


Fig. 6 Performance on medical with different missing label ratio ρ

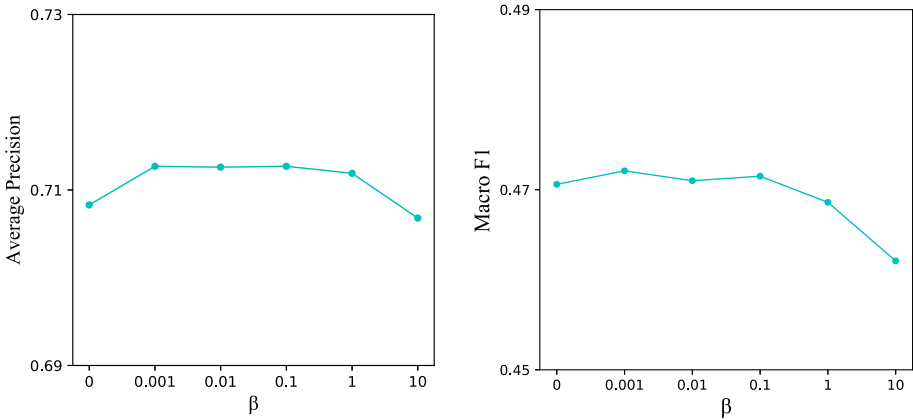


Fig. 7 Effects of parameter β on yeast dataset

Consequently, the performance becomes worse. However, larger values such as $\beta \geq 10$ can also result in performance degradation. The similar circumstance is also observed for η . This further demonstrates that keeping an optimal trade-off between both obtains better results.

5.5 Convergence Analysis

Convergence analysis of proposed method is given in this section. In each iteration, we update the variables with gradient descent. As seen in Algorithm 1, in the t iteration, we should solve the following problem:

$$\begin{aligned}
 \tilde{Y}_{t+1}, Z_{t+1}, D_{t+1} = \arg \min_{\tilde{Y}, Z, D} & \left\| \tilde{Y} - ZD \right\|_F^2 + \beta \operatorname{Tr}(\tilde{Y}^T L_0 \tilde{Y}) \\
 & + \eta \operatorname{Tr}(\tilde{Y} L_1 \tilde{Y}^T) - \alpha \operatorname{Tr}(KHZZ^T H)
 \end{aligned} \tag{39}$$

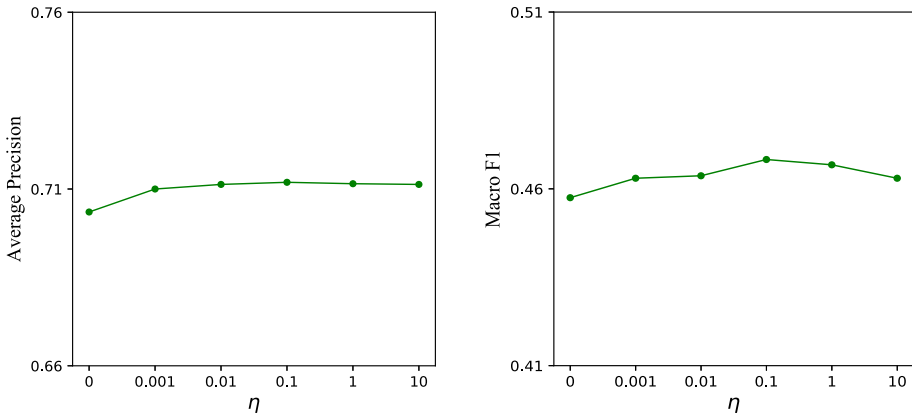


Fig. 8 Effects of parameter η on Yeast dataset

Then we get

$$\begin{aligned} & \left\| \tilde{Y}_{t+1} - Z_{t+1} D_{t+1} \right\|_F^2 + \beta Tr(\tilde{Y}_{t+1}^T L_0 \tilde{Y}_{t+1}) + \eta Tr(\tilde{Y}_{t+1} L_1 \tilde{Y}_{t+1}^T) \\ & - \alpha Tr(KH Z_{t+1} Z_{t+1}^T H) \leq \left\| \tilde{Y}_t - Z_t D_t \right\|_F^2 + \beta Tr(\tilde{Y}_t^T L_0 \tilde{Y}_t) \\ & + \eta Tr(\tilde{Y}_t L_1 \tilde{Y}_t^T) - \alpha Tr(KH Z_t Z_t^T H) \end{aligned} \tag{40}$$

This inequality indicates the algorithm will monotonically decrease the value of the objective function Eq. (19) in each iteration. Besides, the objective function has lower bounds.

Since

$$\begin{aligned} & \beta Tr(\tilde{Y}_t^T L_0 \tilde{Y}_t) + \eta Tr(\tilde{Y}_t L_1 \tilde{Y}_t^T) \\ & = \beta \sum_{i,j} \frac{1}{2} \omega_{i,j} \|\tilde{y}_i - \tilde{y}_j\|^2 + \eta \sum_{i,j} \frac{1}{2} v_{i,j} \|\tilde{y}_i^T - \tilde{y}_j^T\|^2 \geq 0 \end{aligned} \tag{41}$$

As analyzed in Eq. (24), the optimal value of Eq. (24) is $\sum_{i=1}^k \lambda_i$, where $\lambda_i (i = 1, \dots, k)$ is the top k largest eigenvalues of $A = YY^T + \alpha(HKH)$. Thus we can get

$$\left\| \tilde{Y} - ZD \right\|_F^2 - \alpha Tr(KHZZ^T H) \geq - \sum_{i=1}^k \lambda_i \tag{42}$$

Finally we obtain

$$\left\| \tilde{Y} - ZD \right\|_F^2 - \alpha Tr(KHZZ^T H) + \beta Tr(\tilde{Y}_t^T L_0 \tilde{Y}_t) + \eta Tr(\tilde{Y}_t L_1 \tilde{Y}_t^T) \geq - \sum_{i=1}^k \lambda_i \tag{43}$$

Based on the above analysis, the algorithm will converge to the global or local optimal solution. Then the curves of the objective value with the increasing iterations on Emotions and Yeast are drawn in Fig. 9. As can be seen, the algorithm features high speed of convergence in a few iterations. The similar circumstance is also presented on other datasets.

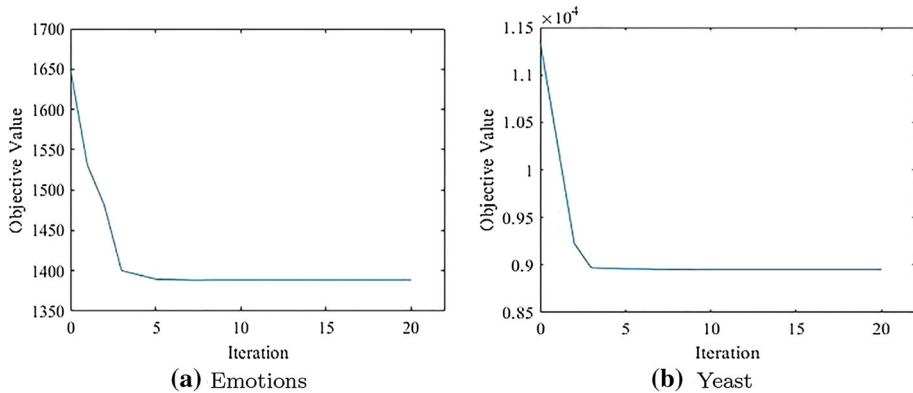


Fig. 9 Convergence of LEDM on Emotions and Yeast datasets

6 Conclusion

In this paper, we present a novel algorithm for label embedding, called LEDM, which embeds the initial label space to a low dimensional latent space by applying low-rank factorization on the label matrix. To achieve the embedding of feature space, the Hilbert-Schmidt independence criterion (HSIC) is utilized to increase the dependence between feature space and label space. Furthermore, low-rank factorization model plays an important role in recovering missing labels. Therefore, the missing labels are also restored through low-rank factorization model and Laplacian manifold regularization based on instance-level and class-level. We integrate above mentioned into an optimization model, which is the first to recover missing labels at the same time while learning embedded space by considering side information from feature space. Extensive experimental results validate the effectiveness of our approach over the state-of-art methods on both full-label and missing-label cases.

In our work, when the number of missing labels is large, the label correlations are not completely and accurately captured. Hence, it is desirable to research label correlation with weak labels and extend our work to semi-supervised multi-label setting in our future endeavors.

Acknowledgements This work was supported by the National Natural Science Foundation of China (61573266).

References

1. Katakis I, Tsoumakas G, Vlahavas I (2008) Multilabel text classification for automated tag suggestion. In: Proceedings of the ECML/PKDD'18
2. Elisseeff A, Weston J (2001) A kernel method for multi-labelled classification. *Adv Neural Inf Proces Syst* 14:681–687
3. Kong D, Ding CHQ, Huang H, Zhao H (2012) Multi-label relief and f-statistic feature selections for image annotation. In: 2012 IEEE conference on computer vision and pattern recognition, pp 2352–2359
4. Tsoumakas G, Katakis I (2007) Multi-label classification: an overview. *IJDWM* 3(3):1–13
5. Boutell MR, Luo J, Shen X, Brown CM (2004) Learning multi-label scene classification. *Pattern Recognit* 37(9):1757–1771
6. Tsoumakas G, Vlahavas IP (2007) Random k -labelsets: an ensemble method for multilabel classification. In: European conference on machine learning, pp 406–417
7. Read J, Pfahringer B, Holmes G, Frank E (2011) Classifier chains for multi-label classification. *Mach Learn* 85(3):333–359

8. Zhang M, Zhou Z (2007) ML-KNN: a lazy learning approach to multi-label learning. *Pattern Recognit* 40(7):2038–2048
9. Yoav F, Schapire R, Abe N (1999) A short introduction to boosting. *J Jpn Soc Artif Intell* 14(1612):771–780
10. Hsu DJ, Kakade S, Langford J, Zhang T (2009) Multi-label prediction via compressed sensing. In: *Advances in neural information processing systems*, pp 772–780
11. Tai F, Lin H (2012) Multilabel classification with principal label space transformation. *Neural Comput* 24(9):2508–2542
12. Chen Y, Lin H (2012) Feature-aware label space dimension reduction for multi-label classification. In: *Advances in neural information processing systems*, pp 1538–1546
13. Huang K, Lin H (2017) Cost-sensitive label embedding for multi-label classification. *Mach Learn* 106(9–10):1725–1746
14. Lin Z, Ding G, Han J, Shao L (2018) End-to-end feature-aware label space encoding for multilabel classification with many classes. *IEEE Trans Neural Netw Learn Syst* 29(6):2472–2487
15. Sun Y, Zhang Y, Zhou Z (2010) Multi-label learning with weak label. In: *Proceedings of the twenty-fourth AAAI conference on artificial intelligence*
16. Gao N, Huang S, Chen S (2016) Multi-label active learning by model guided distribution matching. *Front Comput Sci* 10(5):845–855
17. Wu B, Jia F, Liu W, Ghanem B, Lyu S (2018) Multi-label learning with missing labels using mixed dependency graphs. *Int J Comput Vis* 126(8):875–896
18. Bucak SS, Jin R, Jain AK (2011) Multi-label learning with incomplete class assignments. In: *The 24th IEEE conference on computer vision and pattern recognition*, pp 2801–2808
19. Chen G, Song Y, Wang F, Zhang C (2008) Semi-supervised multi-label learning by solving a Sylvester equation. In: *Proceedings of the SIAM international conference on data mining*, pp 410–419
20. Liu B, Li Y, Xu Z (2018) Manifold regularized matrix completion for multi-label learning with ADMM. *Neural Netw* 101:57–67
21. Wu B, Liu Z, Wang S, Hu B, Ji Q (2014) Multi-label learning with missing labels. In: *22nd international conference on pattern recognition*, pp 1964–1968
22. Yu H, Jain P, Kar P, Dhillon IS (2014) Large-scale multi-label learning with missing labels. In: *Proceedings of the 31th international conference on machine learning*, pp 593–601
23. Xu C, Tao D, Xu C (2016) Robust extreme multi-label learning. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp 1275–1284
24. Ji S, Ye J (2009) An accelerated gradient method for trace norm minimization. In: *Proceedings of the 26th annual international conference on machine learning*, pp 457–464
25. Cai J, Candès EJ, Shen Z (2010) A singular value thresholding algorithm for matrix completion. *SIAM J Optim* 20(4):1956–1982
26. Zhu Y, Kwok JT, Zhou Z (2018) Multi-label learning with global and local label correlation. *IEEE Trans Knowl Data Eng* 30(6):1081–1094
27. Guo B, Hou C, Shan J, Yi D (2018) Low rank multi-label classification with missing labels. In: *24th international conference on pattern recognition*, pp 417–422
28. Xu M, Jin R, Zhou Z (2013) Speedup matrix completion with side information: application to multi-label learning. In: *Advances in neural information processing systems*, pp 2301–2309
29. 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
30. Zhao F, Guo Y (2015) Semi-supervised multi-label learning with incomplete labels. In: *Proceedings of the twenty-fourth international joint conference on artificial intelligence*, pp 4062–4068
31. Yang H, Zhou JT, Cai J (2016) Improving multi-label learning with missing labels by structured semantic correlations. In: *14th European conference on computer vision—ECCV 2016*, pp 835–851
32. 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: *10th international conference on knowledge science, engineering and management*, pp 543–551
33. Koren Y, Bell RM, Volinsky C (2009) Matrix factorization techniques for recommender systems. *IEEE Comput* 42(8):30–37. <https://doi.org/10.1109/MC.2009.263>
34. Wen Z, Yin W, Zhang Y (2012) Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Math Program Comput* 4(4):333–361
35. Song L, Smola AJ, Gretton A, Borgwardt KM, Bedo J (2007) Supervised feature selection via dependence estimation. In: *Proceedings of the twenty-fourth international conference on machine learning*, pp 823–830
36. Fukumizu K, Bach FR, Jordan MI (2004) Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces. *J Mach Learn Res* 5:73–99

37. Yamanishi Y, Vert JP, Kanehisa M (2004) Heterogeneous data comparison and gene selection with kernel canonical correlation analysis. In: Kernel methods in computational biology, pp 209–229
38. Bach FR, Jordan MI (2002) Kernel independent component analysis. *J Mach Learn Res* 3:1–48
39. Gretton A, Herbrich R, Smola AJ (2003) The kernel mutual information. In: 2003 IEEE international conference on acoustics, pp 880–884
40. Gretton A, Bousquet O, Smola AJ, Schölkopf B (2005) Measuring statistical dependence with Hilbert–Schmidt norms. In: 16th international conference on algorithmic learning theory, pp 63–77
41. Gretton A, Fukumizu K, Teo CH, Song L, Schölkopf B, Smola AJ (2007) A kernel statistical test of independence. *Adv Neural Inf Process Syst* 20:585–592
42. Zhang X, Song L, Gretton A, Smola AJ (2008) Kernel measures of independence for non-iid data. In: Proceedings of the twenty-second annual conference on advances in neural information processing systems, Vancouver, British Columbia, Canada, 8–11 December 2008, vol 21, pp 1937–1944
43. Devroye L, Györfi L, Lugosi G (2013) A probabilistic theory of pattern recognition, vol 31. Springer, Berlin
44. Wicker J, Pfahringer B, Kramer S (2012) Multi-label classification using boolean matrix decomposition. In: Proceedings of the ACM symposium on applied computing, pp 179–186
45. Han S, Cao Q, Han M (2012) Parameter selection in SVM with RBF kernel function. *World Autom Congr* 2012:1–4
46. Lu Z, Ip HH, Peng Y (2011) Exhaustive and efficient constraint propagation: a semi-supervised learning perspective and its applications. *CoRR* [arXiv:1109.4684](https://arxiv.org/abs/1109.4684)
47. Pacharawongsakda E, Theeramunkong T (2012) Towards more efficient multi-label classification using dependent and independent dual space reduction. In: 16th Pacific-Asia conference on advances in knowledge discovery and data mining, pp 383–394
48. Han Y, Wu F, Jia J, Zhuang Y, Yu B (2010) Multi-task sparse discriminant analysis (NtSDA) with overlapping categories. In: Proceedings of the twenty-fourth AAAI conference on artificial intelligence
49. Lehoucq RB, Sorensen DC (1996) Deflation techniques for an implicitly restarted Arnoldi iteration. *SIAM J Matrix Anal Appl* 17(4):789–821. <https://doi.org/10.1137/S0895479895281484>
50. Zhang M, Zhou Z (2014) A review on multi-label learning algorithms. *IEEE Trans Knowl Data Eng* 26(8):1819–1837
51. Zhou Z, Zhang M (2017) Multi-label learning. Springer US, New York, pp 875–881
52. Cao L, Xu J (2015) A label compression coding approach through maximizing dependence between features and labels for multi-label classification. In: 2015 International joint conference on neural networks, pp 1–8

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