Partial Multi-Label Learning via Multi-Subspace Representation

Ziwei Li* , **Gengyu Lyu*** and **Songhe Feng†**
Beijing Key Laboratory of Traffic Data Analysis and Mining
School of Computer and Information Technology, Beijing Jiaotong University
{18120390, lvgengyu, shfeng}@bjtu.edu.cn

Abstract

Partial Multi-Label Learning (PML) aims to learn from the training data where each instance is associated with a set of candidate labels, among which only a part of them are relevant. Existing PML methods mainly focus on label disambiguation, while they lack the consideration of noise in feature space. To tackle the problem, we propose a novel framework named *partial multi-label learning via **MU**lti-**S**ubspac**E** **R**epresentation* (**MUSER**), where the redundant labels together with noisy features are jointly taken into consideration during the training process. Specifically, we first decompose the original label space into a latent label subspace and a label correlation matrix to reduce the negative effects of redundant labels, then we utilize the correlations among features to map the original noisy feature space to a feature subspace to resist the noisy feature information. Afterwards, we introduce a graph Laplacian regularization to constrain the label subspace to keep intrinsic structure among features and impose an orthogonality constraint on the correlations among features to guarantee discriminability of the feature subspace. Extensive experiments conducted on various datasets demonstrate the superiority of our proposed method.

1 Introduction

Partial Multi-Label Learning (PML) is a weakly supervised multi-label learning framework, where each instance is associated with a set of labels contained redundant information. The task of PML is to learn a precise predictor for unseen instances from the training data with redundant label information. A straightforward way to solve the problem is applying off-the-shelf MLL methods to train the model [Gibaja and Ventura, 2015]. However, the redundant noise labels mixed in training data will degenerate the performance.

To overcome the problem, [Xie and Huang, 2018] proposed the first PML framework, which provided an effective solution to cope with the redundant candidate labels. Existing PML methods can be roughly classified into two categories: unified strategy and two-stage strategy. For unified strategy-based methods, the whole training process is unified, the prediction model is learned with optimizing candidate labels simultaneously. PML-fp and PML-lc [Xie and Huang, 2018] optimized label confidence values and trained the model by minimizing the ranking loss and exploiting data structure information. fPML [Yu *et al.*, 2018] adopted a feature and label coherent matrix to factorize the original matrix for prediction model training. PML-LRS [Sun *et al.*, 2019] utilized the idea of low-rank and sparse decomposition to get the ground-truth labels and trained the model simultaneously. For two-stage strategy-based methods, the whole training process is divided into two stages, including reliable label selection by disambiguating strategy and model training by using the reliable labels. PARTICLE [Fang and Zhang, 2019] developed iterative label propagation to extract credible labels with high-confidence values, then used the credible labels to train the prediction model. DRAMA [Wang *et al.*, 2019] performed the feature manifold to get the reliable labels with high-confidence values and introduced a gradient boost model for training.

Obviously, existing PML methods mainly focus on the noise in label space while the noise concealed in feature space is regrettably ignored, such as shadow and blurry in multi-label image recognition field. If we directly learn the PML model from such ambiguous features, the performance

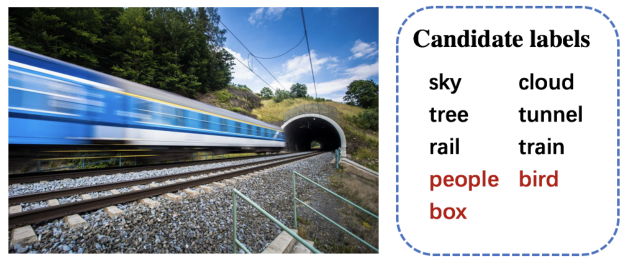


Figure 1: An example of partial multi-label learning with noisy features. Among nine candidate labels of the example, six in black font are ground-truth labels while three in red font are noisy labels. Obviously, the noisy features derived from the high speed motion.

*Indicates equal contribution.
†Corresponding author.

of the learned model would degenerate inevitably. For example, in Figure 1, due to high speed motion, the blue train's feature information is blurred. If the blurred feature information is utilized in the training process directly, the performance of the prediction model will be affected. To get a robust PML model for feature noise, we propose a novel method named *partial multi-label learning via **MU**lti-Subspac**E** **R**epresentation*(**MUSER**), which simultaneously utilizes the feature mapping and label decomposition to train the desired model. Specifically, we firstly decompose the original label matrix into a low-dimensional label subspace matrix and a corresponding label correlation matrix. Secondly, we introduce a graph Laplacian regularization to constrain the latent label subspace matrix to keep the intrinsic structure information among feature information. Thirdly, to resist the feature noise information during training process, we employ a low-dimensional feature subspace matrix mapped by feature correlation matrix to train the model, which can reduce the negative effects in feature space and boost performance by offering a more accurate feature matrix. Meanwhile, to ensure the feature subspace space more discriminative, an orthogonality constraint is imposed on the feature correlation matrix. Finally, the unified prediction model is optimized in an alternative manner by minimizing the least square loss. Extensive experiments have demonstrated that our proposed method can achieve superior performance against state-of-the-art methods.

2 Related Work

As a weakly supervised multi-label learning framework, partial multi-label learning aims to learn a precise multi-label predictor from training data with redundant labels. Actually, PML can be seen as a fusion of two popular learning frameworks: multi-label learning and partial label learning.

Multi-Label Learning (MLL) aims to predict the ground-truth labels for unseen instances, where each instance is associated with a set of accurate labels [Liu and Tsang, 2017; Liu *et al.*, 2018; Feng *et al.*, 2019]. Existing MLL methods can be roughly divided into two categories: 1) Problem transformation methods tackle MLL problem by processing the multi-label training samples to other learning problems [Boutell *et al.*, 2004]. 2) Algorithm adaptation methods tackle MLL problem by adopting the improvment of the commonly used supervised algorithms [Elisseeff and Weston, 2001; Zhang and Zhou, 2007]. Recently, some weakly supervised MLL frameworks are proposed, but most of them are designed to solve missing labels, such as [Sun *et al.*, 2010; Chen *et al.*, 2015].

Partial Label Learning (PLL) is a weakly supervised multi-class learning framework, where each instance is associated with a set of candidate labels and only one label is correct [Feng and An, 2019a; Feng and An, 2019b; Lyu *et al.*, 2019; Lyu *et al.*, 2020]. Existing PLL methods can be roughly divided into three categories: 1) Averaging disambiguation strategy-based methods predict the ground-truth label by the average outputs from the whole candidate label set [Zhang and Yu, 2015]. 2) Identification disambiguation strategy-based methods predict the ground-truth label by refining the model latent parameters [Jin and Ghahramani, 2003]. 3) Disambiguation-free strategy-based methods learn the PLL model by adapting off-the-shelf learning techniques directly without disambiguation process [Zhang *et al.*, 2017; Wu and Zhang, 2018].

Partial Multi-Label Learning (PML) combines the characteristics of MLL and PLL, where each instance is associated with a set of candidate labels and only a part of the them are relevant. Existing PML methods can be roughly divided into two categories: 1) Unified strategy-based methods tackle the PML problem in a unified framework, where the prediction model is trained with optimizing candidate labels simultaneously [Xie and Huang, 2018; Yu *et al.*, 2018; Sun *et al.*, 2019]. 2) Two-stage strategy-based methods decompose PML problem into two subproblems, refining the candidate labels and training the predictor with the refined candidate labels [Wang *et al.*, 2019; Fang and Zhang, 2019].

3 The Proposed Method

Formally speaking, we denote $\mathbf{X}=[\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ as the instance-feature matrix for n instances, $\mathbf{Y} = [\mathbf{y}_1; \mathbf{y}_2; ...; \mathbf{y}_n] \in \{0,1\}^{n\times q}$ as the candidate label matrix where $\mathbf{y}_i$ corresponds to i-th instance's label vector, $y_{ij} = 1$ means the j-th label is included in the candidate label set of instance $\mathbf{x}_i$, $y_{ij} = 0$, otherwise. PML aims to learn a multi-label model from the feature matrix together with candidate label matrix and assign the predictive labels for unseen instances.

3.1 Formulation

MUSER is a novel PML framework based on multi-subspace representation, which can reduce the negative effects caused by redundant labels and noisy features during training process.

Label Subspace We suppose $\widetilde{\mathbf{Y}} \in \{0,1\}^{n\times q}$ is the ground-truth label matrix, and it is not accessible to PML algorithm during the training process. Inspired by the low-rank label matrix in multi-label learning that labels are correlated [Yu *et al.*, 2018], the ground-truth label matrix $\widetilde{\mathbf{Y}}$ can also be assumed to be low-rank in PML. Thus, $\widetilde{\mathbf{Y}}$ can be reduced to a lower-dimensional label subspace $\mathbf{U}$, which is approximated as the product of two matrices:

$$\widetilde{\mathbf{Y}} \simeq \mathbf{UP}, \tag{1}$$

where $\mathbf{U} \in \mathbb{R}^{n\times c}$ denotes the instances representation in c-dimensional latent label subspace and $\mathbf{P} \in \mathbb{R}^{c\times q}$ encodes the label correlation between q labels and c latent labels. Each original label may be affected by all c latent labels, which implies high-order one-to-all label correlation.

To learn $\widetilde{\mathbf{Y}}$ effectively, we minimize the reconstruction error between the candidate label matrix $\mathbf{Y}$ and the product of $\mathbf{U}$ and $\mathbf{P}$ as follows:

$$\min_{\mathbf{U},\mathbf{P}} \ \frac{1}{2}\|\mathbf{Y} - \mathbf{UP}\|_F^2 + \mathcal{R}(\mathbf{U}, \mathbf{P}), \tag{2}$$

where $\mathcal{R}(\mathbf{U}, \mathbf{P})$ denotes the regularization to control the model complexity.

Usually, the ideal latent label subspace is expected to be consistent with intrinsic structural among features [Zhu *et al.*, 2017]. In our model, a graph Laplacian regularization is introduced to ensure such consistency between features and latent labels. Specifically, we define $\mathbf{S} \in \mathbb{R}^{n\times n}$ as a pairwise similarity matrix, where $\mathbf{S}_{ij} = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/\sigma^2)$ if instance i and instance j are the mutually k-nearest neighbors, $\mathbf{S}_{ij} = 0$, otherwise. Then we can get the following regularization term:

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{S}_{ij}\|\frac{\mathbf{u}_i}{\sqrt{\mathbf{E}_{ii}}} - \frac{\mathbf{u}_j}{\sqrt{\mathbf{E}_{jj}}}\|_2^2 = Tr(\mathbf{U}^\top\mathbf{L}\mathbf{U}), \quad (3)$$

where $\mathbf{L} = \mathbf{E}^{-\frac{1}{2}}(\mathbf{E}-\mathbf{S})\mathbf{E}^{-\frac{1}{2}}$ is a graph Laplacian matrix and $\mathbf{E}$ is a diagonal matrix with $\mathbf{E}_{ii} = \sum_{j=1}^{n}\mathbf{S}_{ij}$. By combining the regularization Eq. (3), the formulation can be updated as follows:

$$\min_{\mathbf{U},\mathbf{P}} \frac{\alpha}{2}\|\mathbf{Y} - \mathbf{UP}\|_F^2 + \frac{\beta}{2}Tr(\mathbf{U}^\top\mathbf{L}\mathbf{U}) + \mathcal{R}(\mathbf{U},\mathbf{P}), \quad (4)$$

here α, β denote the trade-off parameters.

Feature Subspace As mentioned before, most existing PML methods just focus on the noisy information in label space and lack the consideration of noise in feature space. Actually, in the real-world application, feature information can be often corrupted by outliers and noise, just like label space. Therefore, we introduce the second subspace representation, latent feature subspace, in our prediction model. A feature correlation matrix $\mathbf{Q} \in \mathbb{R}^{d\times m}$ is learned to map the original feature space to a low-dimensional feature subspace, which can provide compact and discriminative feature information for reducing the negative effects caused by noisy feature information. Here m is the dimension of feature subspace. The latent feature representation in m-dimensional subspace can be formulated as $\mathbf{X}^\top\mathbf{Q}$.

We further introduce a model coefficient matrix $\mathbf{W}\in\mathbb{R}^{m\times c}$, which can map the instance from latent feature subspace to latent label subspace. Accordingly, we can obtain the final objective function for the proposed partial multi-label learning method MUSER:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{Q},\mathbf{U},\mathbf{P}} \ & \frac{1}{2}\|\mathbf{U} - \mathbf{X}^\top\mathbf{QW}\|_F^2 + \frac{\alpha}{2}\|\mathbf{Y} - \mathbf{UP}\|_F^2 \\ & + \frac{\beta}{2}Tr(\mathbf{U}^\top\mathbf{L}\mathbf{U}) + \mathcal{R}(\mathbf{W},\mathbf{U},\mathbf{P}) \\ & s.t. \quad \mathbf{Q}^\top\mathbf{Q} = \mathbf{I},\end{aligned} \quad (5)$$

where $\mathcal{R}(\mathbf{W},\mathbf{U},\mathbf{P}) = \frac{\gamma}{2}(\|\mathbf{W}\|_F^2 + \|\mathbf{U}\|_F^2 + \|\mathbf{P}\|_F^2)$, and the orthogonality constraint for $\mathbf{Q}$ is to ensure the latent feature subspace be more compact after mapping.

In summary, MUSER utilizes both label and feature subspace representations to train the desired model. For the label subspace representation, it can reduce the negative effects caused by redundant labels. For the feature subspace representation, it can reduce the feature noise and generate a discriminative feature information. Combining above two subspaces, the trained PML model is desired to be more effective and robust to both feature and label noises.

Prediction In the predict stage, we firstly adopt the obtained feature correlation matrix $\mathbf{Q}$ to map the unseen instances matrix $\mathbf{X}^*$ to a latent feature subspace, then we utilize the coefficient matrix $\mathbf{W}$ to predict the latent semantics in label subspace, finally we use the label correlation matrix $\mathbf{P}$ to recover the ground-truth labels from the label subspace.

$$\widehat{\mathbf{Y}} = \mathbf{X}^{*\top}\mathbf{QWP}, \quad (6)$$

here $\widehat{\mathbf{Y}}$ is the prediction label matrix corresponding to the $\mathbf{X}^*$.

3.2 Optimization

Our proposed method is convex and it can be solved effectively by an alternating optimization scheme.

Step 1: Calculate P. With $\mathbf{U}, \mathbf{Q}, \mathbf{W}$ fixed, Eq. (5) can be reduced to:

$$\min_{\mathbf{P}} \frac{\alpha}{2}\|\mathbf{Y} - \mathbf{UP}\|_F^2 + \frac{\gamma}{2}\|\mathbf{P}\|_F^2, \quad (7)$$

and we can get the closed form solution:

$$\mathbf{P} = (\alpha\mathbf{U}^\top\mathbf{U} + \gamma\mathbf{I})^{-1}\alpha\mathbf{U}^\top\mathbf{Y}. \quad (8)$$

Step 2: Calculate U. With $\mathbf{P}, \mathbf{Q}, \mathbf{W}$ fixed, we can calculate $\mathbf{U}$ by minimizing the following objective function:

$$\begin{aligned}\min_{\mathbf{U}} \ & \frac{1}{2}\|\mathbf{U} - \mathbf{X}^\top\mathbf{QW}\|_F^2 + \frac{\alpha}{2}\|\mathbf{Y} - \mathbf{UP}\|_F^2 \\ & + \frac{\beta}{2}Tr(\mathbf{U}^\top\mathbf{L}\mathbf{U}) + \frac{\gamma}{2}\|\mathbf{U}\|_F^2.\end{aligned} \quad (9)$$

The objective function is differentiable, thus $\mathbf{U}$ can be optimized via the standard gradient descent algorithm:

$$\nabla_{\mathbf{U}} = (\mathbf{1}+\gamma)\mathbf{U} + \beta\mathbf{LU} + \alpha\mathbf{UPP}^\top - \alpha\mathbf{YP}^\top - \mathbf{X}^\top\mathbf{QW} \quad (10)$$

$$\mathbf{U} := \mathbf{U} - \lambda_{\mathbf{U}}\nabla_{\mathbf{U}}$$

$\nabla_{\mathbf{U}}$ is the gradient of Eq (9), $\lambda_{\mathbf{U}}$ is the stepsize of gradient descent which is obtained by *armijo rule* [Bertsekas, 1999].

Step 3: Calculate Q. With $\mathbf{U}, \mathbf{P}, \mathbf{W}$ fixed, the subproblem to variable $\mathbf{Q}$ is simplified as:

$$\begin{aligned}\min_{\mathbf{Q}} \ & \frac{1}{2}\|\mathbf{U} - \mathbf{X}^\top\mathbf{QW}\|_F^2 \\ & s.t. \quad \mathbf{Q}^\top\mathbf{Q} = \mathbf{I}.\end{aligned} \quad (11)$$

Similarity to **Step 2**, we can get $\mathbf{Q}$ as follows:

$$\mathbf{Q} := \mathbf{Q} - \lambda_{\mathbf{Q}}(-\mathbf{XUW}^\top + \mathbf{XX}^\top\mathbf{QWW}^\top). \quad (12)$$

To satisfy the constraint $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$, we map each row of $\mathbf{Q}$ onto the unit norm ball after each iteration:

$$\mathbf{Q}_{i,:} \leftarrow \frac{\mathbf{Q}_{i,:}}{\|\mathbf{Q}_{i,:}\|}, \quad (13)$$

where $\mathbf{Q}_{i,:}$ is the i-th row of $\mathbf{Q}$.

Step 4: Calculate W. With $\mathbf{P}, \mathbf{U}, \mathbf{Q}$ fixed, Eq. (5) can be reformulated as follows:

$$\min_{\mathbf{W}} \frac{1}{2}\|\mathbf{U} - \mathbf{X}^\top\mathbf{QW}\|_F^2 + \frac{\gamma}{2}\|\mathbf{W}\|_F^2, \quad (14)$$

and we can get the closed form solution:

$$\mathbf{W} = (\mathbf{Q}^\top\mathbf{XX}^\top\mathbf{Q} + \gamma\mathbf{I})^{-1}\mathbf{Q}^\top\mathbf{XU}. \quad (15)$$

During the entire process of optimization, we first initialize the required variables, then repeat the above steps until the function converges or reach the maximum iterations.

Datasets	#n	#d	#q	#Max	#Cardinality
Emotions	593	72	6	3	1.87
Genbase	662	1185	27	6	1.25
Medical	978	1449	45	3	1.25
Corel5k	5000	499	374	5	3.52
Bibtex	7395	1836	159	28	2.4
Eurlex-dc	19348	5000	412	7	1.29
Eurlex-sm	19348	5000	201	12	2.21

Table 1: Characteristics of the employed experimental datasets. For each dataset, the number of examples (#n), the dimension of features (#d), and the number of class labels (#q), the maximum (#Max) and average (#Cardinality) number of ground-truth labels are recorded.

4 Experiments

4.1 Experimental Setup

We conduct experiments on seven PML datasets, which are synthesized from widely-used MLL datasets including *Emotions* [Trohidis *et al.*, 2008], *Genbase* [Diplaris *et al.*, 2005], *Medical* [Pestian *et al.*, 2007], *Corel5k* [Duygulu *et al.*, 2002], *Bibtext* [Katakis *et al.*, 2008], *Eurlex-dc* and *Eurlex-sm* [Mencía and Fürnkranz, 2008]. These datasets are added with redundant noise labels by the controlling parameter r. Here, $r \in \{1, 2, 3\}$ represents the average number of false positive labels for training examples. Table 1 shows the characteristics of the experimental datasets.

To highligt the strengths of MUSER method, we compare it with six state-of-the-art methods, including MLL methods **ML-KNN** [Zhang and Zhou, 2007], **RankSVM** [Elisseeff and Weston, 2001], PML methods **PML-fp** [Xie and Huang, 2018], **fPML** [Yu *et al.*, 2018], **PARTICLE** [Fang and Zhang, 2019], **DRAMA** [Wang *et al.*, 2019]. We also set the trade-off parameters according to the suggestions in respective literatures. Parameters in MUSER method including α, β, γ are chosen from $\{10^{-3}, 10^{-2}, ..., 10^{2}, 10^{3}\}$ with a grid search manner. Five widely-used multi-label metrics are employed to evaluate each comparing method, including *Hamming Loss, Ranking Loss, One-Error, Coverage* and *Average Precision*. Meanwhile, we use 10-fold cross-validation to train the model.

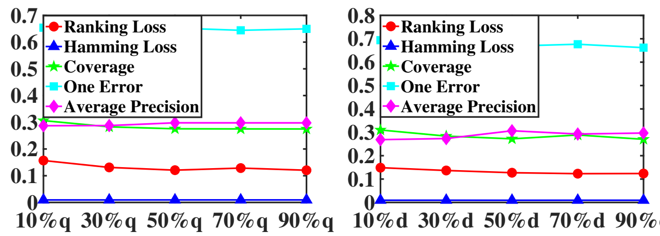
4.2 Experimental Results

Table 3 and Table 4 illustrate the experimental comparisons between our MUSER and other six methods. Due to page limited, we just report part of results, the extra results are reported in the supplementary materials. Out of 735 (7 datasets $\times$ 3 configurations $\times$ 5 metrics $\times$ 7 methods) statistical comparisons, the following observations can be made:

- On twenty-one datasets (7 datasets $\times$ 3 configurations) across all evaluation metrics, MUSER ranks $1st$ in 74.29% cases and ranks $2nd$ in 17.14% cases.
- For each comparing method, MUSER obviously outperforms the counterpart PML methods including PML-fp, fPML, PARTICLE and DRAMA in 99.05%, 87.62%, 91.43% and 89.52% cases, and significantly outperforms the tailored MLL methods including ML-KNN and RankSVM in 90.48% and 99.05% cases.

Evaluation	F_F	Critical value
Ranking Loss	15.8407	
Hamming Loss	14.2050	
One Error	15.6569	2.1750
Coverage	15.8641	
Average Precision	19.8548	

Table 2: Friedman statistics F_F in terms of each evaluation metric and the critical value at 0.05 significance level(#comparing methods $k = 7$, #datasets $N = 21$)



(a) label (b) feature

Figure 2: The performance of MUSER changes as the dimension of subspaces proportion changes. Here q and d are the dimensions of original label and feature space.

- For each evaluation metric, MUSER achieves almost optimal in terms of all evaluation metrics. MUSER is superior to other comparing methods in 96.03% cases on Ranking Loss, 89.68% cases on Hamming Loss, 92.86% cases on One Error, 91.27% cases on Coverage, 96.83% cases on Average Precision.

Furthermore, Friedman test [Demšar, 2006] is utilized as the statistical test to analyze the relative performance among the comparing methods in this paper. Table 2 reports the Friedman statistics F_F and the corresponding critical value. Then the post-hoc Bonferroni-Dunn test [Demšar, 2006] is also utilized to show the relative performance among the comparing methods. Here, MUSER is used as the control method whose average rank difference against the comparing algorithm is calibrated with the *critical difference* (CD). Accordingly, MUSER is deemed to have a significantly different performance to one comparing method if their average ranks differ by at least one CD (CD = 1.759 in this paper: # comparing methods $k = 7$, # datasets $N = 7 \times 3 = 21$). Figure 3 illustrates the CD diagrams on each evaluation metric, where the average rank of each comparing algorithm is marked along the axis (lower ranks to the right). In each subfigure, any comparing methods whose average rank is within one CD to that of MUSER is interconnected to each other with a thick line. Obviously, MUSER performs significant superiority against other comparing methods.

4.3 Further Analysis

Robust Analysis: In order to learn the influence of the varying subspace dimensions, we choose the dimension of label subspace c from $\{10\%q, 30\%q, ..., 90\%q\}$ and feature subspace m from $\{10\%d, 30\%d, ..., 90\%d\}$. Figure 2 shows the results of MUSER with different values of m and c over Corel5k. According to the experimental results, it is noted

Datasets	ML-KNN	RankSVM	PML-fp	fPML	PARTICLE	DRAMA	MUSER
Ranking Loss (the smaller, the better)							
Emotions	0.170 ±0.022	0.244 ±0.092	0.285 ±0.060	0.423 ±0.043	0.241 ±0.022	0.264 ±0.031	**0.168 ±0.031**
Genbase	0.006 ±0.006	0.005 ±0.005	0.079 ±0.033	0.009 ±0.008	0.022 ±0.015	0.006 ±0.009	**0.002 ±0.004**
Medical	0.072 ±0.009	0.086 ±0.001	0.050 ±0.010	0.043 ±0.011	0.099 ±0.025	0.049 ±0.026	**0.024 ±0.006**
Corel5k	0.133 ±0.006	0.112 ±0.006	0.152 ±0.016	0.138 ±0.006	0.354 ±0.055	0.185 ±0.003	**0.110 ±0.007**
Bibtex	0.243 ±0.007	0.224 ±0.001	0.336 ±0.011	**0.092 ±0.007**	0.307 ±0.009	0.192 ±0.012	0.113 ±0.004
Eurlex_dc	0.064 ±0.003	0.120 ±0.005	0.075 ±0.018	0.072 ±0.002	0.058 ±0.004	0.062 ±0.009	**0.045 ±0.003**
Eurlex_sm	0.050 ±0.005	0.079 ±0.006	0.076 ±0.009	0.065 ±0.008	0.049 ±0.002	0.062 ±0.004	**0.047 ±0.003**
Hamming Loss (the smaller, the better)							
Emotions	0.203 ±0.017	0.276 ±0.054	0.300 ±0.035	0.394 ±0.021	0.224 ±0.024	0.258 ±0.020	**0.202 ±0.021**
Genbase	0.005 ±0.003	0.014 ±0.005	0.054 ±0.010	0.005 ±0.002	0.012 ±0.006	0.003 ±0.000	**0.003 ±0.001**
Medical	0.021 ±0.001	0.053 ±0.002	0.055 ±0.005	**0.012 ±0.006**	0.020 ±0.003	0.016 ±0.002	0.014 ±0.002
Corel5k	0.009 ±0.000	0.010 ±0.002	0.012 ±0.001	0.009 ±0.000	0.010 ±0.001	0.013 ±0.002	**0.009 ±0.000**
Bibtex	0.015 ±0.000	0.021 ±0.001	0.018 ±0.000	0.013 ±0.000	0.016 ±0.001	**0.010 ±0.000**	0.014 ±0.009
Eurlex_dc	0.002 ±0.000	0.003 ±0.001	0.010 ±0.003	0.006 ±0.002	0.003 ±0.000	0.004 ±0.001	**0.002 ±0.002**
Eurlex_sm	0.008 ±0.001	0.009 ±0.005	0.013 ±0.002	0.010 ±0.003	0.006 ±0.000	0.008 ±0.001	**0.006 ±0.000**
One Error (the smaller, the better)							
Emotions	0.327 ±0.060	0.387 ±0.134	0.349 ±0.046	0.561 ±0.052	0.290 ±0.049	0.383 ±0.059	**0.270 ±0.080**
Genbase	0.021 ±0.026	0.056 ±0.023	0.174 ±0.053	0.003 ±0.006	0.015 ±0.012	0.009 ±0.015	**0.002 ±0.005**
Medical	0.383 ±0.035	0.532 ±0.043	0.282 ±0.053	0.196 ±0.036	0.245 ±0.045	0.249 ±0.012	**0.159 ±0.032**
Corel5k	0.715 ±0.017	0.758 ±0.013	0.732 ±0.025	**0.649 ±0.024**	0.812 ±0.075	0.679 ±0.026	0.663 ±0.020
Bibtex	0.723 ±0.009	0.518 ±0.003	0.465 ±0.010	0.406 ±0.015	0.575 ±0.013	0.402 ±0.012	**0.368 ±0.019**
Eurlex_dc	0.413 ±0.010	0.581 ±0.021	0.412 ±0.009	0.432 ±0.014	0.376 ±0.009	0.392 ±0.012	**0.277 ±0.006**
Eurlex_sm	0.230 ±0.016	0.241 ±0.009	0.283 ±0.015	0.252 ±0.016	0.230 ±0.027	0.243 ±0.012	**0.226 ±0.012**
Coverage (the smaller, the better)							
Emotions	0.304 ±0.026	0.372 ±0.079	0.425 ±0.056	0.511 ±0.028	0.362 ±0.040	0.381 ±0.043	**0.300 ±0.032**
Genbase	0.021 ±0.011	0.026 ±0.005	0.132 ±0.028	0.030 ±0.017	0.042 ±0.025	0.025 ±0.016	**0.013 ±0.007**
Medical	0.097 ±0.014	0.105 ±0.016	0.050 ±0.033	0.063 ±0.016	0.115 ±0.028	0.063 ±0.012	**0.038 ±0.011**
Corel5k	0.305 ±0.011	0.435 ±0.012	0.532 ±0.021	0.321 ±0.010	0.558 ±0.059	0.465 ±0.015	**0.273 ±0.015**
Bibtex	0.382 ±0.012	0.276 ±0.013	0.325 ±0.013	**0.163 ±0.011**	0.469 ±0.012	0.198 ±0.015	0.211 ±0.009
Eurlex_dc	0.081 ±0.004	0.149 ±0.031	0.109 ±0.013	0.108 ±0.012	0.094 ±0.004	0.075 ±0.016	**0.058 ±0.013**
Eurlex_sm	**0.088 ±0.006**	0.356 ±0.012	0.153 ±0.006	0.108 ±0.006	0.110 ±0.004	0.092 ±0.005	0.095 ±0.006
Average Precision (the larger, the better)							
Emotions	0.793 ±0.020	0.724 ±0.087	0.710 ±0.064	0.577 ±0.032	0.758 ±0.023	0.705 ±0.028	**0.797 ±0.034**
Genbase	0.980 ±0.018	0.965 ±0.014	0.815 ±0.073	0.985 ±0.012	0.972 ±0.020	0.986 ±0.027	**0.994 ±0.007**
Medical	0.701 ±0.021	0.599 ±0.024	0.706 ±0.016	0.852 ±0.031	0.756 ±0.041	0.811 ±0.026	**0.880 ±0.022**
Corel5k	0.255 ±0.007	0.265 ±0.008	0.260 ±0.009	0.276 ±0.009	0.167 ±0.046	0.234 ±0.006	**0.290 ±0.011**
Bibtex	0.260 ±0.007	0.325 ±0.008	0.325 ±0.011	0.542 ±0.012	0.291 ±0.010	0.534 ±0.012	**0.568 ±0.010**
Eurlex_dc	0.635 ±0.008	0.449 ±0.015	0.637 ±0.012	0.663 ±0.022	0.674 ±0.083	0.682 ±0.021	**0.762 ±0.004**
Eurlex_sm	**0.794 ±0.005**	0.532 ±0.012	0.609 ±0.016	0.735 ±0.012	0.678 ±0.014	0.749 ±0.012	0.770 ±0.012

Table 3: Comparison of MUSER with state-of-the-art MLL and PML methods on five evaluation metrics, where the best performances are shown in bold face. ($r = 1$, pairwise t-test at 0.05 significance level)

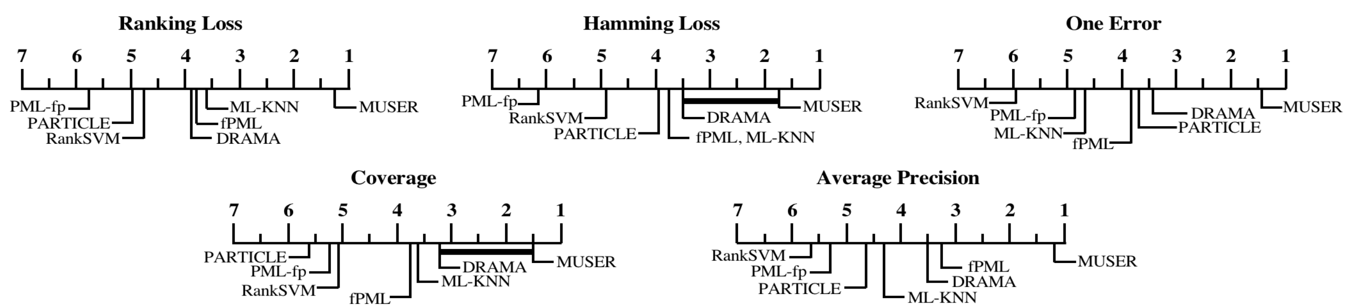


Figure 3: Comparison of MUSER against six comparing methods with the Bonferroni-Dunn test. Methods not connected with MUSER in the CD diagram are considered to have a significantly different performance from MUSER (CD = 1.759 at 0.05 significance level)

that the performance of MUSER is less sensitive to both m and c, and thus in our experiment, m and c are set to 50% of original feature and label space.

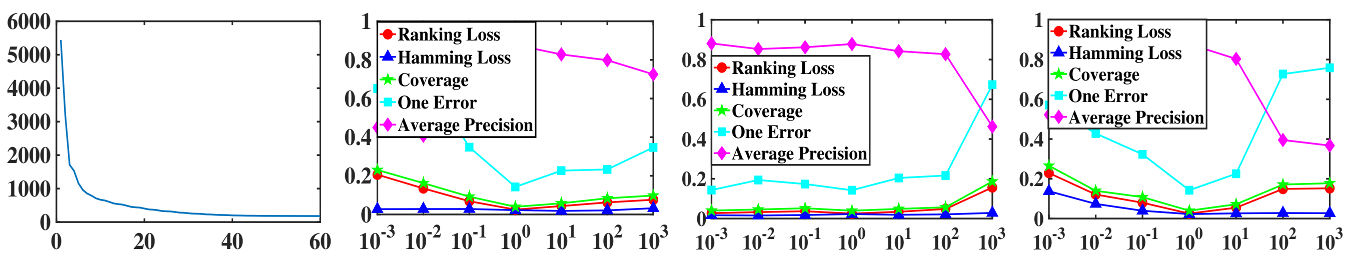
Complexity Analysis: For our proposed model, at each iteration of the method, the main computational complexity includes matrix inversion and multiplication operations. The cost complexity of matrix inversion is $O(c^3 + m^3)$, and generally, $m < d$ and $c < q$, the overall complexity of MUSER is $O(ndq + nq^2 + n^2q + nd^2 + c^3 + m^3)$.

Convergence Analysis: We conduct the convergence analysis of MUSER on *Medical* dataset, where the convergence curve is shown in the left sub-figure of Figure 4. We can observe that the objective function value gradually decreases to a stationary state as the number of iteration increases. Therefore, the convergence of MUSER is demonstrated.

Parameter Analysis: There are three trade-off parameters in MUSER, including α, β, γ. We chose them from $\{10^{-3}, 10^{-2}, ..., 10^2, 10^3\}$. To learn the influence of parameters, we show the experimental results of the three parameters under different configurations on *Medical* dataset. The right three sub-figures of Figure 4 show the performance of MUSER changes as each parameter increases with other parameter fixed. According to the experimental results, the parameters usually follow the optimal configurations ($\alpha =$

Datasets	ML-KNN	RankSVM	PML-fp	fPML	PARTICLE	DRAMA	MUSER
Ranking Loss (the smaller, the better)							
Emotions	0.204 ±0.018	0.225 ±0.065	0.401 ±0.035	0.377 ±0.078	0.252 ±0.028	0.265 ±0.031	**0.192 ±0.030**
Genbase	0.008 ±0.003	0.006 ±0.004	0.009 ±0.001	0.007 ±0.005	0.025 ±0.016	0.008 ±0.003	**0.002 ±0.002**
Medical	0.072 ±0.009	0.086 ±0.001	0.050 ±0.010	0.043 ±0.011	0.099 ±0.025	0.049 ±0.026	**0.030 ±0.011**
Corel5k	0.135 ±0.007	0.165 ±0.008	0.162 ±0.013	0.137 ±0.004	0.349 ±0.070	0.192 ±0.012	**0.120 ±0.005**
Bibtex	0.223 ±0.007	0.243 ±0.008	0.341 ±0.009	**0.087 ±0.004**	0.287 ±0.011	0.203 ±0.011	0.123 ±0.004
Eurlex_dc	0.072 ±0.005	0.132 ±0.006	0.079 ±0.016	0.078 ±0.001	0.062 ±0.004	0.068 ±0.008	**0.049 ±0.002**
Eurlex_sm	**0.032 ±0.001**	0.082 ±0.003	0.082 ±0.012	0.068 ±0.008	0.053 ±0.001	0.051 ±0.006	0.043 ±0.003
Hamming Loss (the smaller, the better)							
Emotions	0.258 ±0.011	0.363 ±0.074	0.392 ±0.023	0.430 ±0.037	0.229 ±0.017	0.289 ±0.028	**0.226 ±0.019**
Genbase	0.005 ±0.002	0.021 ±0.005	0.036 ±0.002	0.003 ±0.001	0.010 ±0.005	0.002 ±0.004	**0.002 ±0.002**
Medical	0.021 ±0.001	0.053 ±0.002	0.055 ±0.005	**0.012 ±0.006**	0.020 ±0.003	0.016 ±0.002	0.014 ±0.002
Corel5k	0.009 ±0.000	0.012 ±0.001	0.012 ±0.001	0.009 ±0.000	0.009 ±0.000	0.015 ±0.003	**0.009 ±0.000**
Bibtex	0.013 ±0.001	0.025 ±0.002	0.017 ±0.002	0.013 ±0.000	0.016 ±0.001	0.012 ±0.001	**0.009 ±0.000**
Eurlex_dc	0.010 ±0.002	0.005 ±0.002	0.007 ±0.005	0.008 ±0.005	0.003 ±0.001	0.006 ±0.002	**0.002 ±0.002**
Eurlex_sm	**0.008 ±0.002**	0.011 ±0.004	0.015 ±0.004	0.012 ±0.004	0.009 ±0.001	0.010 ±0.002	0.013 ±0.001
One Error (the smaller, the better)							
Emotions	0.319 ±0.075	0.371 ±0.097	0.476 ±0.067	0.558 ±0.061	**0.293 ±0.065**	0.383 ±0.089	0.295 ±0.04
Genbase	0.024 ±0.022	0.053 ±0.038	**0.000 ±0.000**	0.002 ±0.005	0.003 ±0.006	0.000 ±0.015	0.003 ±0.005
Medical	0.383 ±0.035	0.532 ±0.043	0.282 ±0.053	0.196 ±0.036	0.245 ±0.045	0.249 ±0.012	**0.176 ±0.035**
Corel5k	0.724 ±0.021	0.768 ±0.012	0.746 ±0.021	0.672 ±0.030	0.823 ±0.091	0.680 ±0.012	**0.665 ±0.016**
Bibtex	0.620 ±0.024	0.529 ±0.016	0.435 ±0.012	0.406 ±0.021	0.549 ±0.015	0.413 ±0.012	**0.377 ±0.017**
Eurlex_dc	0.469 ±0.013	0.589 ±0.012	0.405 ±0.003	0.441 ±0.012	0.357 ±0.007	0.401 ±0.008	**0.283 ±0.012**
Eurlex_sm	**0.186 ±0.004**	0.246 ±0.010	0.286 ±0.016	0.249 ±0.013	0.244 ±0.006	0.236 ±0.009	0.226 ±0.010
Coverage (the smaller, the better)							
Emotions	0.346 ±0.030	0.352 ±0.053	0.487 ±0.052	0.471 ±0.054	0.366 ±0.046	0.378 ±0.046	**0.326 ±0.033**
Genbase	0.024 ±0.009	0.017 ±0.007	0.028 ±0.008	0.012 ±0.011	0.046 ±0.030	0.026 ±0.015	**0.012 ±0.006**
Medical	0.097 ±0.014	0.105 ±0.016	0.050 ±0.033	0.063 ±0.016	0.115 ±0.028	0.063 ±0.012	**0.045 ±0.012**
Corel5k	0.310 ±0.014	0.445 ±0.035	0.436 ±0.016	0.318 ±0.008	0.561 ±0.063	0.468 ±0.012	**0.279 ±0.010**
Bibtex	0.358 ±0.013	0.285 ±0.011	0.333 ±0.015	**0.156 ±0.006**	0.450 ±0.014	0.205 ±0.011	0.231 ±0.008
Eurlex_dc	0.102 ±0.005	0.152 ±0.008	0.091 ±0.012	0.099 ±0.008	0.097 ±0.007	0.081 ±0.012	**0.063 ±0.002**
Eurlex_sm	0.086 ±0.006	0.359 ±0.011	0.155 ±0.009	0.100 ±0.007	0.113 ±0.003	0.096 ±0.008	**0.085 ±0.006**
Average Precision (the larger, the better)							
Emotions	0.765 ±0.023	0.735 ±0.062	0.618 ±0.240	0.605 ±0.049	0.749 ±0.029	0.716 ±0.046	**0.779 ±0.025**
Genbase	0.977 ±0.010	0.964 ±0.025	0.986 ±0.004	0.989 ±0.006	0.978 ±0.016	0.986 ±0.019	**0.994 ±0.006**
Medical	0.701 ±0.021	0.599 ±0.024	0.706 ±0.016	0.852 ±0.031	0.756 ±0.041	0.811 ±0.026	**0.861 ±0.025**
Corel5k	0.252 ±0.011	0.250 ±0.045	0.250 ±0.009	0.268 ±0.013	0.162 ±0.054	0.228 ±0.004	**0.289 ±0.007**
Bibtex	0.319 ±0.015	0.356 ±0.005	0.319 ±0.009	0.544 ±0.011	0.315 ±0.012	0.549 ±0.008	**0.550 ±0.014**
Eurlex_dc	0.625 ±0.009	0.432 ±0.009	0.618 ±0.006	0.658 ±0.019	0.631 ±0.009	0.675 ±0.010	**0.752 ±0.009**
Eurlex_sm	0.773 ±0.005	0.528 ±0.011	0.600 ±0.016	0.729 ±0.013	0.667 ±0.028	0.752 ±0.013	**0.774 ±0.009**

Table 4: Comparison of MUSER with state-of-the-art MLL and PML methods on five evaluation metrics, where the best performances are shown in bold face. ($r = 2$, pairwise t-test at 0.05 significance level)



(a) Convergence curve (b) Vary α (c) Vary β (d) Vary γ

Figure 4: The left subfigure shows the objective function value of MUSER changes with increasing number of iterations. The right three subfigures show performance of MUSER changes as each parameter increases with other parameters fixed.

$1, \beta = 1, \gamma = 1$) but vary with minor adjustments on different datasets.

5 Conclusion

In this paper, we propose a novel PML framework named MUSER, which trains a robust model by considering the noise in both feature space and label space. Specially, we use low-rank decomposition to reduce the negative effects of redundant labels and introduce graph Laplacian regularization to ensure the label subspace be in consistent with features, then we utilize feature subspace mapping and orthogonal subspace projection to provide a discriminative feature information. Empirical studies on various datasets demonstrate the superiority of MUSER.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Nos. 61872032), in part by the Beijing Natural Science Foundation (Nos. 4202058, 9192008), in part by National Key Research and Development Project (No. 2018AAA0100300), and in part by the Fundamental Research Funds for the Central universities (Nos. 2020YJS026, 2019JBM020).

References

[Bertsekas, 1999] Dimitri P Bertsekas. Nonlinear programming. *Athena scientific*, 1999.

[Boutell *et al.*, 2004] Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

[Chen *et al.*, 2015] Zheng Chen, Minmin Chen, Kilian Q. Weinberger, and Weixiong Zhang. Marginalized denoising for link prediction and multi-label learning. In *AAAI*, pages 1707–1713, 2015.

[Demšar, 2006] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *JMLR*, 7(1):1–30, 2006.

[Diplaris *et al.*, 2005] Sotiris Diplaris, Grigorios Tsoumakas, Pericles A Mitkas, and Ioannis Vlahavas. Protein classification with multiple algorithms. In *PCI*, volume 3746, pages 448–456, 2005.

[Duygulu *et al.*, 2002] Pinar Duygulu, Kobus Barnard, Joao FG de Freitas, and David A Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*, pages 97–112, 2002.

[Elisseeff and Weston, 2001] André Elisseeff and Jason Weston. A kernel method for multi-labelled classification. In *NIPS*, pages 681–687, 2001.

[Fang and Zhang, 2019] Jun-Peng Fang and Min-Ling Zhang. Partial multi-label learning via credible label elicitation. In *AAAI*, pages 3518–3525, 2019.

[Feng and An, 2019a] Lei Feng and Bo An. Partial label learning by semantic difference maxization. In *IJCAI*, pages 2294–2300, 2019.

[Feng and An, 2019b] Lei Feng and Bo An. Partial label learning with self-guided retraining. In *AAAI*, pages 3542–3549, 2019.

[Feng *et al.*, 2019] Lei Feng, Bo An, and Shuo He. Collaboration based multi-label learning. In *AAAI*, pages 3550–3557, 2019.

[Gibaja and Ventura, 2015] Eva Gibaja and Sebastian Ventura. A tutorial on multi-label learning. *CSUR*, 47(3):1–38, 2015.

[Jin and Ghahramani, 2003] Rong Jin and Zoubin Ghahramani. Learning with multiple labels. In *NIPS*, pages 921–928, 2003.

[Katakis *et al.*, 2008] Ioannis Katakis, Grigorios Tsoumakas, and Ioannis Vlahavas. Multilabel text classification for automated tag suggestion. In *ECML/PKDD*, pages 5–13, 2008.

[Liu and Tsang, 2017] Weiwei Liu and Ivor Tsang. Making decision trees feasible in ultrahigh feature and label dimensions. *JMLR*, 18:1–36, 07 2017.

[Liu *et al.*, 2018] Weiwei Liu, Donna Xu, Ivor W Tsang, and Wenjie Zhang. Metric learning for multi-output tasks. *TPAMI*, 41(2):408–422, 2018.

[Lyu *et al.*, 2019] Gengyu Lyu, Songhe Feng, Tao Wang, Congyan Lang, and Yidong Li. Gm-pll: Graph matching based partial label learning. *TKDE*, pages 1–15, 2019.

[Lyu *et al.*, 2020] Gengyu Lyu, Songhe Feng, Yidong Li, Yi Jin, Guojun Dai, and Congyan Lang. Hera: Partial label learning by combining heterogeneous loss with sparse and low-rank regularization. *TIST*, 11(3):1–19, 2020.

[Mencía and Fürnkranz, 2008] Eneldo Mencía and Johannes Fürnkranz. Efficient pairwise multilabel classification for large-scale problems in the legal domain. In *ECML/PKDD*, pages 50–65, 2008.

[Pestian *et al.*, 2007] John Pestian, Chris Brew, Paweł Matykiewicz, D. Hovermale, Neil Johnson, Kevin Cohen, and Duch Wlodzislaw. A shared task involving multi-label classification of clinical free text. In *BioNLP '07*, pages 97–104, 2007.

[Sun *et al.*, 2010] Yu-Yin Sun, Yin Zhang, and Zhi-Hua Zhou. Multi-label learning with weak label. In *AAAI*, pages 593–598, 2010.

[Sun *et al.*, 2019] Lijuan Sun, Songhe Feng, Tao Wang, Congyan Lang, and Yi Jin. Partial multi-label learning by low-rank and sparse decomposition. In *AAAI*, pages 5016–5023, 2019.

[Trohidis *et al.*, 2008] Konstantinos Trohidis, Grigorios Tsoumakas, George Kalliris, and Ioannis P Vlahavas. Multi-label classification of music into emotions. In *ISMIR*, pages 325–330, 2008.

[Wang *et al.*, 2019] Haobo Wang, Weiwei Liu, Yang Zhao, Chen Zhang, Tianlei Hu, and Gang Chen. Discriminative and correlative partial multi-label learning. In *IJCAI*, pages 3691–3697, 2019.

[Wu and Zhang, 2018] Xuan Wu and Min-Ling Zhang. Towards enabling binary decomposition for partial label learning. In *IJCAI*, pages 2868–2874, 2018.

[Xie and Huang, 2018] Ming-Kun Xie and Sheng-Jun Huang. Partial multi-label learning. In *AAAI*, pages 4302–4309, 2018.

[Yu *et al.*, 2018] Guo-Xian Yu, Xia Chen, Carlotta Domeniconi, Jun Wang, Zhao Li, Zili Zhang, and Xindong Wu. Feature-induced partial multi-label learning. In *ICDM*, pages 1398–1403, 2018.

[Zhang and Yu, 2015] Min-Ling Zhang and Fei Yu. Solving the partial label learning problem: An instance-based approach. In *IJCAI*, pages 4048–4054, 2015.

[Zhang and Zhou, 2007] Min-Ling Zhang and Zhi-Hua Zhou. Ml-knn: A lazy learning approach to multi-label leaming. *Pattern Recognition*, 40(7):2038–2048, 2007.

[Zhang *et al.*, 2017] Min-Ling Zhang, Fei Yu, and Cai-Zhi Tang. Disambiguation-free partial label learning. *TKDE*, 29(10):2155–2167, 2017.

[Zhu *et al.*, 2017] Yue Zhu, James T Kwok, and Zhi-Hua Zhou. Multi-label learning with global and local label correlation. *TKDE*, 30(6):1081–1094, 2017.