Recovering Accurate Labeling Information from Partially Valid Data for Effective Multi-Label Learning

Ximing Li[1,2] and **Yang Wang**[3,4,*]

[1] College of Computer Science and Technology, Jilin University, China

[2] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, China

[3] Key Laboratory of Knowledge Engineering with Big Data, Ministry of Education, Hefei University of Technology, China

[4] School of Computer Sci & Information Engineering, Hefei University of Technology, China
liximing86@gmail.com, yangwang@hfut.edu.cn

Abstract

Partial Multi-label Learning (PML) aims to induce the multi-label predictor from datasets with noisy supervision, where each training instance is associated with several candidate labels but only partially valid. To address the noisy issue, the existing PML methods basically recover the ground-truth labels by leveraging the ground-truth confidence of the candidate label, *i.e.,* the likelihood of a candidate label being a ground-truth one. However, they neglect the information from non-candidate labels, which potentially contributes to the ground-truth label recovery. In this paper, we propose to recover the ground-truth labels, *i.e.,* estimating the ground-truth confidences, from the label enrichment, composed of the relevance degrees of candidate labels and irrelevance degrees of non-candidate labels. Upon this observation, we further develop a novel two-stage PML method, namely *Partial Multi-Label Learning with Label Enrichment-Recovery* (PML^{3}ER), where in the first stage, it estimates the label enrichment with unconstrained label propagation, then jointly learns the ground-truth confidence and multi-label predictor given the label enrichment. Experimental results validate that PML^{3}ER outperforms the state-of-the-art PML methods.

1 Introduction

Partial **M**ulti-label **L**earning (**PML**), a novel learning paradigm with noisy supervision, draws increasing attention from the machine learning community [Fang and Zhang, 2019; Sun *et al.*, 2019]. It refers to induce the multi-label predictor from PML datasets, with each training instance associated with multiple candidate labels that are only partially valid. The PML datasets are available in many real-world applications, where collecting the accurate supervision is quite expensive for many scenarios, *e.g.,* crowdsourcing annotations. To visualize this, we illustrate a PML image instance in Fig.1(a): an annotator may roughly select more candidate labels, so as to cover all ground-truth labels but ineluctably with several irrelevant ones, imposing big challenges for learning with such noisy PML training instances.

Formally speaking, given a PML dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{i=n}$ with n instances and l labels, where $\mathbf{x}_i \in \mathbb{R}^d$ denotes the feature vector and $\mathbf{y}_i \in \{0,1\}^l$ the candidate label set of $\mathbf{x}_i$. For $\mathbf{y}_i$, the value of 0 or 1 represents the corresponding label to be a non-candidate or candidate label. Let $\mathbf{y}^* \in \{0,1\}^l$ denote the (unknown) ground-truth label sets of instances. Specifically, for each instance $\mathbf{x}_i$, its corresponding $\mathbf{y}_i^*$ is covered by the candidate label set $\mathbf{y}_i$, *i.e.,* $\mathbf{y}_i^* \subseteq \mathbf{y}_i$. Accordingly, the task of PML aims to induce a multi-label predictor $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \{0,1\}^l$ from $\mathcal{D}$.

To solve the problem, several typical attempts have been made [Xie and Huang, 2018; Yu *et al.*, 2018; Fang and Zhang, 2019; Sun *et al.*, 2019], where the basic idea is to recover the ground-truth labels by leveraging the *ground-truth confidence* of the candidate label, *i.e.,* the likelihood of a candidate label being a ground-truth one, and learning with it, instead of the candidate label. For example, an early framework of PML [Fang and Zhang, 2019] estimates the ground-truth confidences via label propagation over an instance neighbor graph, following the intuition that neighboring instances tend to share the same labels; another work [Sun *et al.*, 2019] recovered the ground-truth confidences by decomposing the candidate labels under a low-rank scheme.

Revisiting the existing PML methods, we find that they basically estimate the ground-truth confidence around candidate label annotations, but neglect the information from non-candidate labels, which potentially contributes to the ground-truth label recovery. The irrelevance degree of the non-candidate label, *i.e.,* the degree of a non-candidate label being irrelevant with the instance, is contributive to distinguish the irrelevant noisy labels within candidate label sets, since any candidate label tends to be an irrelevant noisy one if its highly correlated labels are given higher irrelevance degrees. For example, in Fig.1(b) and (c), the candidate label "airplane" is more likely to be an irrelevant noisy label, since its highly correlated labels, *e.g.,* "airport" and "runway", are with higher irrelevance degrees for the example image instance.

[*] Yang Wang is the Corresponding Author, denoted by *

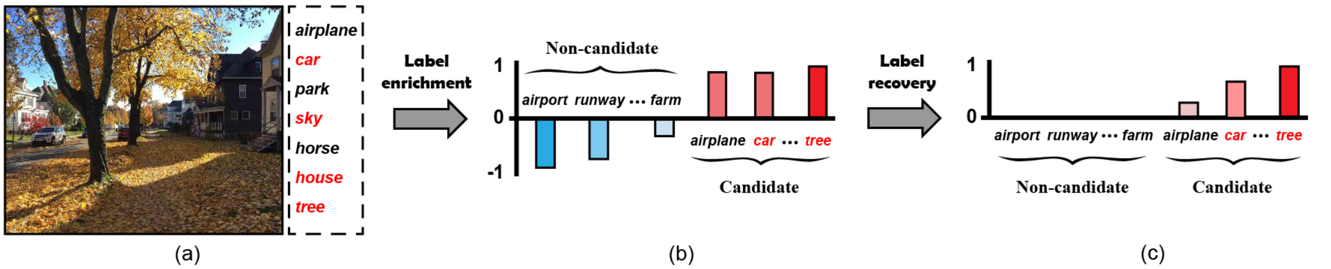

Figure 1: The basic idea of PML^{3}ER. (a) An example of PML image instance annotated with 7 candidate labels, with only 4 of them are ground-truth labels, *i.e.,* the ones in red (best view in color). (b) indicates the label enrichment, *i.e.,* estimating both the relevance degrees of the candidate labels within (0,1) range, and irrelevance degrees of the non-candidate labels within (-1,0) range. (c) indicates the label recovery, *i.e.,* estimating the ground-truth confidences of candidate labels from the label enrichment. The candidate label "airplane" is more likely to be an irrelevant noisy label, since its highly correlated labels, *e.g.,* "airport" and "runway", are with higher irrelevance degrees, to filter out the noisy candidate label.

Based on the above analysis, we propose to estimate the ground-truth confidences over both candidate and non-candidate labels. In particular, we develop a novel two-stage PML method, namely **P**artial **M**ulti-**L**abel **L**earning with **L**abel **E**nrichment-**R**ecovery (**PML^{3}ER**). On the first stage, we estimate the *label enrichment*, composed of the relevance degrees of candidate labels (*i.e.,* the complementary definition of irrelevance degree) and irrelevance degrees of non-candidate ones, using an unconstrained label propagation procedure. In the second stage, we jointly train the ground-truth confidence and multi-label predictor given the label enrichment learned from the first stage. We conduct extensive experiments to validate the effectiveness of PML^{3}ER.

The contributions of this paper are summarized below:

- We propose PML^{3}ER by leveraging both the information from candidate and non-candidate labels.
- The PML^{3}ER estimates the label enrichment using unconstrained label propagation, and then trains the multi-label predictor with label recovery simultaneously.
- We conduct extensive experiments to validate the effectiveness of PML^{3}ER.

2 Related Work

2.1 Partial Multi-label Learning

Abundant researches towards Partial Label Learning (PLL) have been made, where each training instance is annotated with a candidate label set but only one is valid [Cour *et al.*, 2011; Liu and Dietterich, 2012; Chen *et al.*, 2014; Zhang *et al.*, 2017; Yu and Zhang, 2017; Wu and Zhang, 2018; Gong *et al.*, 2018; Chen *et al.*, 2018; Feng and An, 2018; Feng and An, 2019b; Feng and An, 2019a; Wang *et al.*, 2019]. The core idea of PLL follows the spirit of disambiguation, *i.e.,* identifying the ground-truth label from the candidate label set for each instance. In some cases, PLL can be deemed as a special case of PML, where the ground-truth label number is fixed to one. Naturally, PML is more challenging than PLL, even the number of ground-truth labels is unknown.

The existing PML methods mainly recover the ground-truth labels by estimating the ground-truth confidences [Xie and Huang, 2018; Yu *et al.*, 2018; Fang and Zhang, 2019; Sun *et al.*, 2019]. Two PML methods are proposed [Xie and Huang, 2018], *i.e.,* Partial Multi-label Learning with *l*abel *c*orrelation (PML-*lc*) and *f*eature *p*rototype (PML-*fp*), which are upon the ranking loss objective weighted by ground-truth confidences. The other one [Sun *et al.*, 2019], namely Partial Multi-label Learning by Low-Rank and Sparse decomposition (PML-LRS), trains the predictor with ground-truth confidences under the low-rank assumption. Besides them, the two-stage PML framework [Fang and Zhang, 2019], *i.e.,* PARTIal multi-label learning via Credible Label Elicitation (PARTICLE), estimates the ground-truth confidences by label propagation, and then trains the predictor over candidate labels with high confidences only. Two traditional methods of victual Label Splitting (VLS) and Maximum A Posteriori (MAP) are used in its second stage, leading to two versions, *i.e.,* PARTICLE-VLS and PARTICLE-MAP.

Orthogonal to those methods, our PML^{3}ER estimates the ground-truth confidence using the label enrichment involving both candidate and non-candidate labels to recover more accurate supervision.

2.2 Learning with Label Enrichment

Learning with label enrichment, also known as label enhancement, explores richer label information, *e.g.,* the label membership degrees of instances, to enhance the performance [Gayar *et al.*, 2006; Jiang *et al.*, 2006; Li *et al.*, 2015; Hou *et al.*, 2016; Xu *et al.*, 2018]. The existing methods achieve the label enrichment using the similarity knowledge of instances by various schemes, such as fuzzy clustering [Gayar *et al.*, 2006], label propagation [Li *et al.*, 2015], manifold learning [Hou *et al.*, 2016] *etc.* They have been applied to the paradigms of multi-label learning and label distribution learning with accurate annotations.

Our PML^{3}ER also refers to the label enrichment, but it works on the scenario of PML under noisy supervision.

3 The PML^{3}ER Algorithm

Following the notations in Section 1, we denote $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]^\top \in \mathbb{R}^{n\times d}$ and $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_n]^\top \in \{0,1\}^{n\times l}$ the instance matrix and candidate label matrix, respectively. For each instance $\mathbf{x}_i$, $y_{ij} = 1$ means that the j-th label is a candidate label, otherwise it is a non-candidate one.

Given a PML dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$, our PML^{3}ER first estimates the label enrichment, which describes the relevance degrees of candidate labels and irrelevance degrees of non-candidate ones. Specifically, for each instance $\mathbf{x}_i$, we denote the corresponding label enrichment $\widehat{\mathbf{y}}_i = [\widehat{y}_{i1}, \cdots, \widehat{y}_{il}]^\top$ as follows:

$$\widehat{y}_{ij} \in \begin{cases} [0,1], & \textbf{if } y_{ij} = 1 \\ [-1,0], & \textbf{otherwise} \end{cases} \qquad \forall j \in [l], \tag{1}$$

referring to the example image instance in Fig.1(b). Further, we denote by $\widehat{\mathbf{Y}} = [\widehat{\mathbf{y}}_1, \cdots, \widehat{\mathbf{y}}_n]^\top$ the label enrichment matrix. Then, PML^{3}ER induces the multi-label predictor from the enrichment version of PML dataset $\widehat{\mathcal{D}} = \{\mathbf{X}, \widehat{\mathbf{Y}}\}$.

We now describe the two stages of PML^{3}ER, *i.e., label enrichment by unconstrained propagation* and *jointly learning the ground-truth confidence and multi-label predictor*.

3.1 Label Enrichment by Unconstrained Propagation

In the first stage, PML^{3}ER estimates the label enrichment matrix $\widehat{\mathbf{Y}}$. Inspired by [Li *et al.*, 2015; Fang and Zhang, 2019], we develop an unconstrained label propagation procedure, which estimates the labeling degrees by progressively propagating annotation information over a weighted *k*-Nearest Neighbor (*k*NN) graph over the instances. The intuition is that the candidate and non-candidate labels with more accumulations during the *k*NN propagation tend to be with higher relevance degrees and lower irrelevance degrees, respectively.

We describe the detailed steps of the unconstrained label propagation procedure below.

[Step 1]: After constructing the *k*NN graph $\mathbf{\Omega}$, for each instance $\mathbf{x}_i$, we compute its reconstruction weight vector of *k*NNs, *i.e.,* $\mathbf{v}_i = [v_{i1}, \cdots, v_{in}]^\top \in \mathbb{R}^n$, by minimizing the following objective:

$$\begin{aligned} &\min_{\mathbf{v}_i} \|\mathbf{x}_i - \mathbf{X}^\top \mathbf{v}_i\|_2^2 \\ &\text{s.t. } v_{ij} \geq 0 \quad \forall j \in \mathbf{\Omega}(\mathbf{x}_i), \ v_{ij} = 0 \quad \forall j \notin \mathbf{\Omega}(\mathbf{x}_i), \end{aligned} \tag{2}$$

where $\mathbf{\Omega}(\mathbf{x}_i)$ denotes the *k*NNs of $\mathbf{x}_i$. This objective can be efficiently solved by any off-the-shelf Quadratic Programming (QP) solver[1]. Repeat to solve the problem of Eq.(2) for each instance, we can achieve the reconstruction weight matrix $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_n]^\top \in \mathbb{R}^{n \times n}$. Then, we normalize $\mathbf{V}$ by row, *i.e.,* $\mathbf{V} \leftarrow \mathbf{V}\mathbf{D}^{-1}$, where $\mathbf{D} = \text{diag}[d_1, \cdots, d_n]$, $d_i = \sum_{j=1}^n v_{ij}$.

[Step 2]: Following the idea that the relationships of instances can be translated to their associated labels, we can enrich the labeling information by propagating over $\mathbf{\Omega}$ with $\mathbf{V}$. Formally, we denote by $\mathbf{F} = [\mathbf{f}_1, \cdots, \mathbf{f}_n]^\top \in \mathbb{R}_+^{n \times l}$ the label propagation solution, which is initialized by the candidate label matrix $\mathbf{Y}$, *i.e.,* $\mathbf{F}^{(0)} = \mathbf{Y}$. Then, $\mathbf{F}$ can be iteratively updated by propagating over $\mathbf{\Omega}$ with $\mathbf{V}$ until convergence. At each iteration t, the update equation is given by:

$$\mathbf{F}^{(t)} = \alpha \cdot \mathbf{V}^\top \mathbf{F}^{(t-1)} + (1 - \alpha) \cdot \mathbf{F}^{(0)}, \tag{3}$$

where $\alpha \in [0,1]$ is the propagation rate. To avoid overestimate non-candidate labels, we normalize each row of $\mathbf{F}^{(t)}$:

$$f_{ij}^{(t)} = \mathbf{min}\left(1\,,\ \frac{f_{ij}^{(t)} - \mathbf{vmin}(\mathbf{f}_i^{(t)})}{\mathbf{cvmax}(\mathbf{f}_i^{(t)}) - \mathbf{vmin}(\mathbf{f}_i^{(t)})}\right), \qquad \forall i \in [n],\ \forall j \in [l], \tag{4}$$

where $\mathbf{vmin}(\mathbf{f}_i^{(t)})$ denotes the minimum of $\mathbf{f}_i^{(t)}$, and $\mathbf{cvmax}(\mathbf{f}_i^{(t)})$ the maximum over candidate labels in $\mathbf{f}_i^{(t)}$.

Algorithm 1 Unconstrained label propagation for $\widehat{\mathbf{Y}}$

```
Input: The PML dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$, nearest neighbor number $k$ and propagation rate $\alpha$;
Output: The label enrichment matrix $\widehat{\mathbf{Y}}$.
Construct the kNN graph of $\mathbf{X}$, and initialize $\mathbf{F}^{(0)}$ by $\mathbf{Y}$
Compute $\mathbf{V}$ by solving Eq.(2) for each instance, and then normalize it by row
While not convergence Do
    Update $\mathbf{F}$ using Eq.(3)
    Normalize $\mathbf{F}$ using Eq.(4)
End While
Compute $\widehat{\mathbf{Y}}$ using Eq.(5)
```

Algorithm 2 Predictive model induction for PML^{3}ER

```
Input: The enriched PML dataset $\widehat{\mathcal{D}} = \{\mathbf{X}, \widehat{\mathbf{Y}}\}$, regularization parameters $\{\lambda_1, \lambda_2\}$;
Output: The predictive parameter matrix $\mathbf{W}$.
Initialize $\{\mathbf{C}, \mathbf{B}, \mathbf{W}, \widehat{\mathbf{B}}, \mathbf{\Theta}\}$
While not convergence Do
    Update $\mathbf{C}$ using Eq.(10)
    For $t = 1$ to $N_{iter}$
        Update $\widehat{\mathbf{B}}, \mathbf{B}, \mathbf{\Theta}$ using Eqs.(13), (14) and (15)
    End For
    Update $\mathbf{W}$ using Eq.(17)
End While
```

[Step 3]: After obtaining the optimal $\mathbf{F}$, denoted by $\mathbf{F}^* = [f_{ij}^*]_{n \times l}$, we compute the label enrichment matrix $\widehat{\mathbf{Y}}$ as follows:

$$\widehat{y}_{ij} = \begin{cases} f_{ij}^*\,, & \textbf{if } y_{ij} = 1 \\ f_{ij}^* - 1\,, & \textbf{otherwise} \end{cases} \qquad \forall i \in [n].\ \forall j \in [l] \tag{5}$$

For clarity, we summarize the unconstrained label propagation for $\widehat{\mathbf{Y}}$ in *Algorithm 1.*

3.2 Jointly Learning the Ground-truth Confidence and Multi-label Predictor

In the second stage, PML^{3}ER jointly trains the ground-truth confidence matrix $\mathbf{C} = [c_{ij}]_{n \times l} \in [0,1]^{n \times l}$ and multi-label predictor given the enrichment version of PML dataset $\widehat{\mathcal{D}} = \{\mathbf{X}, \widehat{\mathbf{Y}}\}$.

First, we aim to recover $\mathbf{C}$ from $\widehat{\mathbf{Y}}$ by leveraging the following minimization:

$$\begin{aligned} &\min_{\mathbf{C}, \mathbf{B}} \|\widehat{\mathbf{Y}} - \mathbf{C}\mathbf{B}\|_F^2 + \lambda_1 \|\mathbf{B}\|_* \\ &\textbf{s.t.} \quad \mathbf{0}_{n \times l} \preceq \mathbf{C} \preceq \mathbf{Y}, \end{aligned} \tag{6}$$

[1] We apply the public QP solver of *mosek* downloaded at https://www.mosek.com/

where $\mathbf{B} = [b_{ij}]_{l\times l}$ denotes the label correlation matrix, and $\mathbf{0}_{n\times l}$ the all zero matrix. Specifically for capturing local label correlations, we utilize the nuclear regularizer for $\mathbf{B}$, *i.e.,* $\|\mathbf{B}\|_*$, with the regularization parameter λ_1.

Second, we aim to train a linear multi-label predictor with $\mathbf{C}$ by leveraging a least square minimization with a squared Frobenius norm regularization:

$$\min_{\mathbf{W}} \|\mathbf{C}-\mathbf{XW}\|_F^2+\lambda_2\|\mathbf{W}\|_F^2 \tag{7}$$

where $\mathbf{W}\in\mathbb{R}^{d\times l}$ is the predictive parameter matrix, and λ_2 the regularization parameter.

By combining Eqs.(6) and (7), we achieve the overall objective as follows:

$$\begin{aligned}&\min_{\mathbf{C},\mathbf{B},\mathbf{W}} \|\widehat{\mathbf{Y}}-\mathbf{CB}\|_F^2+\|\mathbf{C}-\mathbf{XW}\|_F^2+\lambda_1\|\mathbf{B}\|_*+\lambda_2\|\mathbf{W}\|_F^2\\ &\textbf{s.t.}\quad \mathbf{0}_{n\times l}\preceq\mathbf{C}\preceq\mathbf{Y}\end{aligned} \tag{8}$$

Discussion on Recovering C from $\widehat{\mathbf{Y}}$. Referring to Eq.(6), we jointly learn the ground-truth confidence matrix $\mathbf{C}$ and label correlation matrix $\mathbf{B}$ by minimizing their reconstruction error of $\widehat{\mathbf{Y}}$. By omitting the regularizers, it can be roughly re-expressed below:

$$\begin{aligned}&\min_{\mathbf{C},\mathbf{B}} \sum\nolimits_{i=1}^{n}\sum\nolimits_{j=1}^{l}\left(\widehat{y}_{ij}-\sum\nolimits_{h=1}^{l}c_{ih}b_{hj}\right)^2\\ &\quad\textbf{s.t.}\quad c_{ij}\in[0,1],\ \forall y_{ij}=1;\quad c_{ij}=0,\ \forall y_{ij}=0\end{aligned}$$

Obviously, for each component $\widehat{y}_{ij}$, it contributes that any c_{ih} tends to be larger or smaller given larger value of b_{hj} (*i.e.,* a higher correlation between label j and h), if $\widehat{y}_{ij}$ corresponds to a candidate label (≥ 0) or a non-candidate one (≤ 0). That is, we actually recover $\mathbf{C}$ using the information from candidate and non-candidate labels simultaneously.

Optimization

As directly solving the objective of Eq.(8) is intractable, we optimize the variables of interest, *i.e.,* $\{\mathbf{C},\mathbf{B},\mathbf{W}\}$, via alternating fashion, by optimizing one variable with the other two fixed. Repeat this process until convergence or reaching the maximum number of iterations. We describe the update equations of $\{\mathbf{C},\mathbf{B},\mathbf{W}\}$ one by one.

[**Update** $\mathbf{C}$] When $\{\mathbf{B},\mathbf{W}\}$ are fixed, the sub-objective with respect to $\mathbf{C}$ can be reformulated as follows:

$$\begin{aligned}&\min_{\mathbf{C}} \|\widehat{\mathbf{Y}}-\mathbf{CB}\|_F^2+\|\mathbf{C}-\mathbf{XW}\|_F^2\\ &\textbf{s.t.}\quad \mathbf{0}_{n\times l}\preceq\mathbf{C}\preceq\mathbf{Y}\end{aligned} \tag{9}$$

This optimization refers to a convex optimization, so as to achieve the following truncated update equation:

$$c_{ij}\leftarrow\begin{cases}0\,, & \textbf{if}\ c'_{ij}\leq 0\\ 1\,, & \textbf{if}\ c'_{ij}\geq 1\\ c'_{ij}\,, & \textbf{otherwise}\end{cases}\quad \forall i\in[n],\forall j\in[l], \tag{10}$$

where $\mathbf{C}'=[c'_{ij}]_{n\times l}$:

$$\mathbf{C}'=(\widehat{\mathbf{Y}}\mathbf{B}^\top+\mathbf{XW})(\mathbf{BB}^\top+\mathbf{I}_l)^{-1}$$

Algorithm 3 Full algorithm of PML^{3}ER

```
Input: The PML dataset $\mathcal{D}=\{\mathbf{X},\mathbf{Y}\}$, $k=10$, $\alpha=0.5$ and
    $\{\lambda_1,\lambda_2\}$;
Output: The predictive parameter matrix $\mathbf{W}$.
  1: Compute the label enrichment matrix $\widehat{\mathbf{Y}}$ using Algorithm
     1
  2: Optimize $\mathbf{W}$ using Algorithm 2 given $\mathcal{D}=\{\mathbf{X},\widehat{\mathbf{Y}}\}$
```

[**Update** $\mathbf{B}$] When $\{\mathbf{C},\mathbf{W}\}$ are fixed, the sub-objective with respect to $\mathbf{B}$ can be reformulated as follows:

$$\min_{\mathbf{B}} \|\widehat{\mathbf{Y}}-\mathbf{CB}\|_F^2+\lambda_1\|\mathbf{B}\|_* \tag{11}$$

Following the spirit of Alternating Direction Method of Multipliers (ADMM) [Boyd *et al.*, 2011], we convert Eq.(11) into an augmented Lagrange problem with an auxiliary matrix $\widehat{\mathbf{B}}\in\mathbb{R}^{l\times l}$ and Lagrange parameter matrix $\boldsymbol{\Theta}\in\mathbb{R}^{l\times l}$:

$$\min_{\widehat{\mathbf{B}},\mathbf{B},\boldsymbol{\Theta}} \|\widehat{\mathbf{Y}}-\mathbf{C}\widehat{\mathbf{B}}\|_F^2+\lambda_1\|\mathbf{B}\|_*+\frac{\tau}{2}\|\mathbf{B}-\widehat{\mathbf{B}}+\frac{\boldsymbol{\Theta}}{\tau}\|_F^2, \tag{12}$$

where τ is the penalty parameter. We perform an inner iteration to alternatively optimizing each of $\{\widehat{\mathbf{B}},\mathbf{B},\boldsymbol{\Theta}\}$ with the other two fixed. After some simple algebra, the update equations are formulated as follows:

$$\widehat{\mathbf{B}}\leftarrow(2\mathbf{C}^\top\mathbf{C}+\tau\mathbf{I}_l)^{-1}(2\mathbf{C}^\top\widehat{\mathbf{Y}}+\tau\mathbf{B}+\boldsymbol{\Theta}) \tag{13}$$

For $\mathbf{B}$, it can be directly solved by

$$\mathbf{B}\leftarrow\mathbf{SVD}_{\frac{\lambda_1}{\tau}}\left(\widehat{\mathbf{B}}-\frac{\boldsymbol{\Theta}}{\tau}\right), \tag{14}$$

where $\mathbf{SVD}_{\frac{\lambda_1}{\tau}}(\cdot)$ is the singular thresholding with $\frac{\lambda_1}{\tau}$ [Cai *et al.*, 2010]. Then, for $\boldsymbol{\Theta}$, it can be updated by:

$$\boldsymbol{\Theta}\leftarrow\boldsymbol{\Theta}+\tau(\mathbf{B}-\widehat{\mathbf{B}}) \tag{15}$$

[**Update** $\mathbf{W}$] When $\{\mathbf{C},\mathbf{B}\}$ are fixed, the sub-objective with respect to $\mathbf{W}$ can be reformulated as follows:

$$\min_{\mathbf{W}} \|\mathbf{C}-\mathbf{XW}\|_F^2+\lambda_2\|\mathbf{W}\|_F^2 \tag{16}$$

The problem has an analytic solution as follows:

$$\mathbf{W}=(\mathbf{X}^\top\mathbf{X}+\lambda_2\mathbf{I}_d)^{-1}\mathbf{X}^\top\mathbf{C} \tag{17}$$

For clarity, we summarize the procedure of this predictive model induction in *Algorithm 2*.

3.3 PML^{3}ER Summary

We describe some implementation details of PML^{3}ER. First, following [Fang and Zhang, 2019], we fix the parameters of the unconstrained label propagation as: $k = 10$ and $\alpha = 0.05$. Second, we empirically fix the penalty parameter τ of ADMM as 1. Third, the maximum iteration number of both ADMM loops is set to 5, as to be widely known, ADMM basically converges fast. Overall, the PML^{3}ER algorithm is outlined in *Algorithm 3*.

Dataset	n	d	l	#AL	Domain
Genbase	662	1186	27	1.252	biology
Medical	978	1449	45	1.245	text
Arts	5000	462	26	1.636	text
Corel5k	5000	499	374	3.522	images
Bibtex	7395	1836	159	2.406	text

Table 1: Statistics of the original multi-label datasets. "#AL": average label number of each instance.

Time Complexity. We also discuss the time complexity of PML^{3}ER. First, in the unconstrained label propagation procedure, we require $\mathcal{O}(n^2d^2)$ time to construct the weighted *k*NN graph, and $\mathcal{O}(T_1n^2l)$ time to obtain the label enrichment matrix, referring to Eq.(3), where T_1 denotes its iteration number. Second, in predictive model induction, the major computational costs include matrix inversion and SVD, requiring roughly $\mathcal{O}(T_2(d^3+n^2l))$ time, where T_2 denotes the iteration number of the outer loop[2]. Therefore, the total time complexity of PML^{3}ER is $\mathcal{O}(n^2d^2 + T_1n^2l + T_2(d^3 + n^2l))$.

4 Experiment

4.1 Experimental Setup

Datasets

In the experiment, we downloaded five public multi-label datasets from the *mulan* website[3], including *Genbase*, *Medical*, *Arts*, *Corel5k* and *Bibtex*. The statistics of those datasets are outlined in Table 1.

To conduct experiments under the scenario of noisy supervision, we generate synthetic PML datasets from each original dataset by randomly drawing irrelevant noisy labels. Specifically, for each instance, we create the candidate label set by adding several randomly drawn irrelevant labels with $a\%$ number of ground-truth labels, where we vary a over the range $\{50, 100, 150, 200\}$. Besides, to avoid useless PML instances annotated with all l labels, we fix each candidate label set at most $l-1$ labels. Accordingly, a total of twenty synthetic PML datasets are generated.

Baseline Methods

We employed five baseline methods for comparison, including three PML methods and two traditional methods of Multi-label Learning (ML). For the ML ones, we directly train them over the synthetic PML datasets by considering all candidate labels as ground-truth ones. We outline the method-specific settings below.

- **PML-*fp*** [Xie and Huang, 2018]: A PML method with a ranking loss objective weighted by ground-truth confidences. We utilize the code provided by its authors, and tune the parameters following the original paper. Here, the other version of [Xie and Huang, 2018], *i.e.,* PML-*lc*, was neglected, since it performed worse than PML-*fp* in our early experiments.
- **PARTICLE-MAP** [Fang and Zhang, 2019]: A two-stage PML method with label propagation. We employ the public code[4], and tune the parameters following the original paper. Here, the other version of [Fang and Zhang, 2019], *i.e.,* PARTICLE-VLS, was neglected, since it performed worse than PARTICLE-MAP in our early experiments.
- **PML-LRS** [Sun *et al.*, 2019]: A PML method with candidate label decomposition. We utilize the code provided by its authors, and tune the regularization parameters over $\{10^i | i = -3, \cdots, 1\}$ using cross-validations.
- **M**ulti-**L**abel ***k*** **N**earest **N**eighbor (**ML-*k*NN**) [Zhang and Zhou, 2007]: A *k*NN ML method. We employ the public code[5] implemented by its authors, and tune its parameters following the original paper.
- Multi-label learning with **L**abel spec**I**fic **F**ea**T**ures (**LIFT**) [Zhang and Wu, 2015]: A binary relevance ML method. We employ the public code[6] implemented by its authors, and tune its parameters following the original paper.

For our PML^{3}ER, the regularization parameter λ_1 is fixed to 1, and λ_2 is tuned over $\{10^i | i = 1, 2\}$ using 5-fold cross validation results.

Evaluation Metrics

We employed seven evaluation metrics [Zhang and Zhou, 2014], including *Subset Accuracy* (SAccuracy), *Hamming Loss* (HLoss), *One Error* (OError), *Ranking Loss* (RLoss), *Average Precision* (AP), *Macro Averaging F1* (Macro-F1), *Micro Averaging F1* (Micro-F1), where both instance-based and label-based metrics are covered. For SAccuracy, AP, Macro-F1 and Micro-F1, the higher value is better, while for HLoss, OError and RLoss, a smaller value is better.

4.2 Experimental Results

For each PML dataset, we randomly generate five 50%/50% training/test splits, and evaluate the average scores (±standard deviation) of comparing algorithms. Due to the space limitation, we only present detailed results of RLoss, AP and Macro-F1 in Table 2, while the observations of other metrics tend to be similar. First, we can observe that PML^{3}ER outperforms other three PML methods in most cases, where PML^{3}ER dominates the scenarios of AP and Macro-F1 on different noise levels. Especially, the performance gain over PARTICLE-MAP indicates that using the information from non-candidate labels is beneficial for PML. Besides, we can see that PML^{3}ER significantly performs better than the two traditional ML methods in most cases, since they directly use noisy candidate labels for training.

Additionally, for each PML dataset and evaluation metric, we conduct a pairwise *t*-test (at 0.05 significance level) to examine whether PML^{3}ER is statistically different to baselines. The win/tie/loss counts over 20 PML datasets and 7 evaluation metrics are presented in Table 3. We can observe

[2] Here, we omit the iteration numbers of inner ADMM loop for $\mathbf{B}$, since it converges very fast.

[3] http://mulan.sourceforge.net/datasets-mlc.html

[4] http://palm.seu.edu.cn/zhangml/files/PARTICLE.rar

[5] http://palm.seu.edu.cn/zhangml/files/ML-kNN.rar

[6] http://palm.seu.edu.cn/zhangml/files/LIFT.rar

Dataset	a	PML^{3}ER (Ours)	PARTICLE-MAP	PML-LRS	PML-*fp*	ML-*k*NN	LIFT
RLoss ↓							
Genbase	50	**.005 ± .002**	.010 ± .002	.006 ± .002	.019 ± .006	.012 ± .003	.009 ± .003
	100	**.005 ± .003**	.010 ± .002	**.005 ± .003**	.018 ± .007	.014 ± .006	.008 ± .004
	150	**.008 ± .003**	.010 ± .001	.009 ± .002	.019 ± .010	.019 ± .006	.012 ± .004
	200	**.007 ± .003**	.009 ± .001	.010 ± .003	.014 ± .004	.017 ± .005	.012 ± .005
Medical	50	**.028 ± .005**	.048 ± .005	.033 ± .006	.042 ± .009	.075 ± .009	.044 ± .005
	100	**.030 ± .006**	.050 ± .006	.032 ± .006	.043 ± .011	.078 ± .012	.046 ± .007
	150	**.034 ± .007**	.054 ± .005	.035 ± .007	.042 ± .009	.094 ± .012	.055 ± .006
	200	**.031 ± .005**	.049 ± .004	.035 ± .007	.043 ± .015	.088 ± .012	.056 ± .008
Arts	50	.154 ± .003	.142 ± .002	.162 ± .002	**.132 ± .002**	.165 ± .003	.137 ± .003
	100	.162 ± .004	.152 ± .002	.170 ± .005	**.131 ± .002**	.166 ± .003	.143 ± .005
	150	.175 ± .002	.158 ± .003	.186 ± .003	**.140 ± .004**	.174 ± .005	.155 ± .003
	200	.180 ± .002	.165 ± .003	.192 ± .003	**.146 ± .001**	.172 ± .004	.156 ± .004
Corel5k	50	.174 ± .002	**.128 ± .002**	.206 ± .002	.198 ± .006	.146 ± .002	.144 ± .001
	100	.179 ± .003	**.132 ± .002**	.216 ± .003	.178 ± .008	.152 ± .001	.154 ± .002
	150	.185 ± .003	**.134 ± .002**	.230 ± .004	.169 ± .003	.156 ± .002	.164 ± .002
	200	.186 ± .002	**.135 ± .002**	.232 ± .003	.176 ± .005	.161 ± .003	.175 ± .003
Bibtex	50	**.094 ± .002**	.190 ± .004	.126 ± .003	.112 ± .004	.240 ± .002	.121 ± .003
	100	**.100 ± .002**	.187 ± .003	.138 ± .002	.107 ± .003	.250 ± .003	.130 ± .004
	150	.112 ± .002	.187 ± .002	.157 ± .002	**.109 ± .003**	.260 ± .003	.143 ± .003
	200	.116 ± .001	.189 ± .005	.165 ± .001	**.111 ± .001**	.266 ± .009	.145 ± .002
AP ↑							
Genbase	50	**.991 ± .005**	.978 ± .005	.988 ± .004	.981 ± .006	.968 ± .007	.982 ± .006
	100	**.992 ± .004**	.978 ± .004	.989 ± .004	.981 ± .007	.967 ± .008	.984 ± .003
	150	**.988 ± .003**	.978 ± .004	.979 ± .002	.976 ± .011	.958 ± .011	.979 ± .008
	200	**.989 ± .003**	.979 ± .003	.981 ± .005	.979 ± .006	.964 ± .014	.977 ± .011
Medical	50	**.882 ± .014**	.798 ± .018	.853 ± .021	.852 ± .023	.742 ± .029	.830 ± .017
	100	**.881 ± .018**	.791 ± .021	.861 ± .021	.855 ± .022	.741 ± .029	.822 ± .015
	150	**.867 ± .022**	.781 ± .024	.858 ± .018	.845 ± .019	.720 ± .033	.804 ± .006
	200	**.870 ± .016**	.768 ± .013	.855 ± .020	.853 ± .029	.715 ± .034	.797 ± .020
Arts	50	**.598 ± .003**	.528 ± .004	.588 ± .003	.577 ± .007	.488 ± .005	.595 ± .005
	100	**.597 ± .004**	.513 ± .004	.584 ± .005	.578 ± .005	.486 ± .005	.591 ± .007
	150	**.577 ± .005**	.499 ± .006	.564 ± .005	.558 ± .005	.478 ± .006	**.577 ± .003**
	200	.572 ± .003	.491 ± .006	.557 ± .005	.554 ± .005	.477 ± .003	**.578 ± .005**
Corel5k	50	**.295 ± .004**	.263 ± .005	.282 ± .003	.240 ± .003	.233 ± .003	.244 ± .005
	100	**.293 ± .004**	.260 ± .003	.276 ± .004	.242 ± .003	.229 ± .003	.217 ± .004
	150	**.289 ± .004**	.264 ± .005	.266 ± .003	.241 ± .003	.226 ± .003	.194 ± .003
	200	**.288 ± .004**	.260 ± .004	.266 ± .004	.241 ± .003	.224 ± .003	.185 ± .005
Bibtex	50	**.567 ± .004**	.383 ± .007	.532 ± .003	.517 ± .003	.295 ± .004	.487 ± .007
	100	**.555 ± .004**	.380 ± .006	.509 ± .005	.522 ± .005	.282 ± .005	.467 ± .007
	150	**.536 ± .004**	.369 ± .003	.476 ± .006	.519 ± .004	.270 ± .004	.448 ± .007
	200	**.528 ± .006**	.365 ± .004	.460 ± .005	.511 ± .006	.266 ± .005	.440 ± .007
Macro-F1 ↑							
Genbase	50	**.710 ± .029**	.543 ± .053	.680 ± .015	.598 ± .023	.622 ± .022	.619 ± .044
	100	**.722 ± .044**	.522 ± .017	.710 ± .027	.594 ± .056	.594 ± .022	.600 ± .039
	150	**.649 ± .033**	.536 ± .034	.618 ± .039	.603 ± .030	.540 ± .016	.566 ± .046
	200	**.652 ± .054**	.533 ± .019	.559 ± .033	.603 ± .026	.562 ± .058	.579 ± .039
Medical	50	**.405 ± .027**	.270 ± .013	.301 ± .016	.296 ± .015	.243 ± .034	.309 ± .022
	100	**.363 ± .017**	.254 ± .015	.314 ± .021	.320 ± .020	.235 ± .020	.293 ± .015
	150	**.348 ± .026**	.238 ± .006	.316 ± .013	.294 ± .014	.208 ± .016	.264 ± .010
	200	**.373 ± .024**	.227 ± .016	.315 ± .015	.315 ± .036	.192 ± .019	.266 ± .020
Arts	50	.244 ± .012	.201 ± .007	.220 ± .008	.159 ± .008	.123 ± .007	**.249 ± .009**
	100	**.251 ± .012**	.190 ± .004	.227 ± .006	.156 ± .004	.119 ± .008	.247 ± .006
	150	**.240 ± .007**	.180 ± .005	.229 ± .006	.129 ± .004	.112 ± .010	.237 ± .009
	200	**.226 ± .006**	.184 ± .007	.210 ± .006	.126 ± .004	.108 ± .005	.217 ± .005
Corel5k	50	.040 ± .001	.027 ± .001	.039 ± .001	.005 ± .000	.020 ± .001	**.046 ± .002**
	100	.038 ± .000	.028 ± .002	.038 ± .001	.004 ± .000	.020 ± .002	**.040 ± .002**
	150	**.037 ± .001**	.034 ± .002	**.037 ± .001**	.004 ± .000	.019 ± .002	.035 ± .002
	200	**.036 ± .000**	.032 ± .002	.032 ± .000	.004 ± .000	.018 ± .001	.033 ± .001
Bibtex	50	**.375 ± .002**	.163 ± .004	.359 ± .004	.299 ± .006	.133 ± .001	.299 ± .008
	100	**.360 ± .002**	.157 ± .004	.332 ± .004	.311 ± .002	.120 ± .004	.271 ± .008
	150	**.340 ± .002**	.146 ± .003	.296 ± .006	.306 ± .007	.109 ± .001	.247 ± .009
	200	**.331 ± .004**	.145 ± .005	.278 ± .004	.301 ± .002	.103 ± .005	.232 ± .010

Table 2: Experimental results (mean ± std) in terms of *RLoss*, *AP*, *Macro-F1*, where the best performance is shown in boldface.

that PML^{3}ER significantly outperforms the PML baseline methods PARTICLE-MAP, PML-LRS and PML-*fp* in 91.4%, 86.4% and 91.4% cases, and also outperforms the two traditional MLL methods ML-*k*NN and LIFT in 95.7% and 85% cases. Besides, on the results of evaluation metrics, PML^{3}ER achieves significantly better scores. For example, the SAccuracy, OError, AP, Macro-F1 and Micro-F1 of PML^{3}ER are better than all comparing algorithms in 95% cases.

4.3 Parameter Sensitivity

We empirically analyze the sensitivities of the regularization parameters $\{\lambda_1, \lambda_2\}$ of PML^{3}ER. For each one, we examine the AP scores by varying its value from $\{10^i | i = -3, \cdots, 3\}$ across PML datasets with $a = 100$ by holding the other two fixed. The experimental results are presented in Fig.2. First, we can observe that the AP scores of λ_1 seem quite stable over different types of PML datasets. That is, we empirically conclude that PML^{3}ER is insensitive to λ_1, making PML^{3}ER more practical. Second, for λ_2, it performs better with the values of $\{10^i | i = 1, 2\}$ across all PML datasets, which are

Baseline Method	SAccuracy	HLoss	OError	RLoss	AP	Macro-F1	Micro-F1	**Total**
Particle-Map	20/0/0	16/4/0	20/0/0	12/0/8	20/0/0	20/0/0	20/0/0	**128/4/8**
PML-LRS	17/2/1	9/11/0	20/0/0	17/3/0	20/0/0	18/2/0	20/0/0	**121/18/1**
PML-*fp*	20/0/0	18/2/0	19/1/0	11/1/8	20/0/0	20/0/0	20/0/0	**128/4/8**
ML-*k*NN	20/0/0	20/0/0	20/0/0	14/0/6	20/0/0	20/0/0	20/0/0	**134/0/6**
Lift	19/0/1	17/3/0	17/2/1	12/0/8	18/1/1	17/0/3	19/1/0	**119/7/14**

Table 3: Win/tie/loss counts of pairwise t-test (at 0.05 significance level) between PML^{3}ER and each comparing approach.

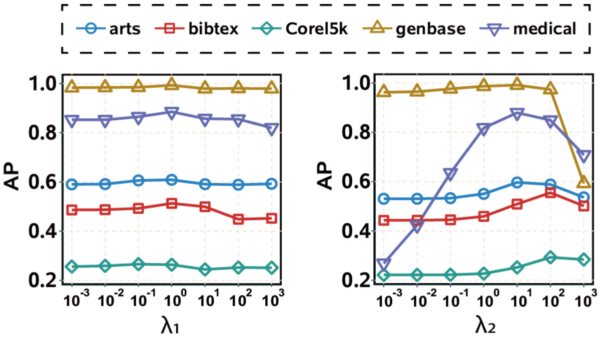

Figure 2: Sensitivity analysis of the regularization parameters $\{\lambda_1, \lambda_2\}$

the settings used in our experiment.

5 Conclusion

We concentrate on the task of PML, and propose a novel two-stage PML^{3}ER algorithm. In the first stage, PML^{3}ER performs an unconstrained label propagation procedure to estimate the label enrichment, which simultaneously involves the relevance degrees of candidate labels and irrelevance degrees of non-candidate labels. In the second stage, PML^{3}ER jointly learns the ground-truth confidence and multi-label predictor given the label enrichment. Extensive experiments on PML datasets indicate the superior performance of PML^{3}ER.

Acknowledgments

We would like to acknowledge support for this project from the National Natural Science Foundation of China (NSFC) (No.61602204, No.61876071, No.61806035, No.U1936217)

References

[Boyd *et al.*, 2011] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[Cai *et al.*, 2010] Jian-Feng Cai, Emmanuel J. Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[Chen *et al.*, 2014] Yi-Chen Chen, Vishal M. Patel, Rama Chellappa, and P. Jonathon Phillips. Ambiguously labeled learning using dictionaries. *IEEE Transactions on Information Forensics and Security*, 9(12):2076–2088, 2014.

[Chen *et al.*, 2018] Ching-Hui Chen, Vishal M. Patel, and Rama Chellappa. Learning from ambiguously labeled face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(7):1653–1667, 2018.

[Cour *et al.*, 2011] Timothee Cour, Benjamin Sapp, and Ben Taskar. Learning from partial labels. *Journal of Machine Learning Research*, 12:1501–1536, 2011.

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

[Feng and An, 2018] Lei Feng and Bo An. Leveraging latent label distributions for partial label learning. In *International Joint Conference on Artificial Intelligence*, pages 2107–2113, 2018.

[Feng and An, 2019a] Lei Feng and Bo An. Partial label learning by semantic difference maximization. In *International Joint Conference on Artificial Intelligence*, pages 2294–2300, 2019.

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

[Gayar *et al.*, 2006] Neamat El Gayar, Friedhelm Schwenker, and Günther Palm. A study of the robustness of kNN classifiers trained using soft labels. In *International Conference on Artificial Neural Network in Pattern Recognition*, pages 67–80, 2006.

[Gong *et al.*, 2018] Chen Gong, Tongliang Liu, Yuanyan Tang, Jian Yang, Jie Yang, and Dacheng Tao. A regularization approach for instance-based superset label learning. *IEEE Transactions on Cybernetics*, 48(3):967–978, 2018.

[Hou *et al.*, 2016] Peng Hou, Xin Geng, and Min-Ling Zhang. Multi-label manifold learning. In *AAAI Conference on Artificial Intelligence*, pages 1680–1686, 2016.

[Jiang *et al.*, 2006] Xiufeng Jiang, Zhang Yi, and Jian Cheng Lv. Fuzzy SVM with a new fuzzy membership function. *Neural Computing & Applications*, 15(3-4):268–276, 2006.

[Li *et al.*, 2015] Yu-Kun Li, Min-Ling Zhang, and Xin Geng. Leveraging implicit relative labeling-importance information for effective multi-label learning. In *IEEE International Conference on Data Mining*, pages 251–260, 2015.

[Liu and Dietterich, 2012] Li-Ping Liu and Thomas G. Dietterich. A conditional multinomial mixture model for superset label learning. In *Neural Information Processing Systems*, pages 548–556, 2012.

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

[Wang *et al.*, 2019] Deng-Bao Wang, Li Li, and Min-Ling Zhang. Adaptive graph guided disambiguation for partial label learning. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 83–91, 2019.

[Wu and Zhang, 2018] Xuan Wu and Min-Ling Zhang. Towards enabling binary decomposition for partial label learning. In *International Joint Conference on Artificial Intelligence*, pages 2868–2874, 2018.

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

[Xu *et al.*, 2018] Ning Xu, An Tao, and Xin Geng. Label enhancement for label distribution learning. In *International Joint Conference on Artificial Intelligence*, pages 2926–2932, 2018.

[Yu and Zhang, 2017] Fei Yu and Min-Ling Zhang. Maximum margin partial label learning. *Machine Learning*, 106:573–593, 2017.

[Yu *et al.*, 2018] Guoxian Yu, Xia Chen, Carlotta Domeniconi, Jun Wang, Zhao Li, Zili Zhang, and Xindong Wu. Feature-induced partial multi-label learning. In *IEEE International Conference on Data Mining*, pages 1398–1403, 2018.

[Zhang and Wu, 2015] Min-Ling Zhang and Lei Wu. LIFT: Multi-label learning with label-specific features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):107–120, 2015.

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

[Zhang and Zhou, 2014] Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.

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