# Weakly Supervised Multi-Label Learning via Label Enhancement

**Jia-Qi Lv**, **Ning Xu**, **Ren-Yi Zheng** and **Xin Geng**[*]
MOE Key Laboratory of Computer Network and Information Integration, China
School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
{lvjiaqi, xning, zhengry, xgeng}@seu.edu.cn

## Abstract

Weakly supervised multi-label learning (WSML) concentrates on a more challenging multi-label classification problem, where some labels in the training set are missing. Existing approaches make multi-label prediction by exploiting the incomplete logical labels directly without considering the relative importance of each label to an instance. In this paper, a novel two-stage strategy named *Weakly Supervised Multi-label Learning via Label Enhancement* (WSMLLE) is proposed to learn from weakly supervised data via label enhancement. Firstly, the relative importance of each label, i.e., the *description degrees* are recovered by leveraging the structural information in the feature space and local correlations learned from the label space. Then, a tailored multi-label predictive model is induced by learning from the training instances with the recovered description degrees. To our best knowledge, it is the first attempt to unify the complement of the missing labels and the recovery of the description degrees into the same framework. Extensive experiments across a wide range of real-world datasets clearly validate the superiority of the proposed approach.

## 1 Introduction

Multi-label learning (MLL) [Zhang and Zhou, 2014] framework has been widely studied because of its success in fitting multiple semantic meanings problems. In MLL, each instance is associated with multiple labels simultaneously, so it requires investigation of a large number of candidate labels one by one, which is usually impracticable due to the high cost of data labeling process. Thus, labels are usually missing in the training set. To deal with the performance deterioration of MLL caused by the missing labels, a paradigm which is often called *weakly supervised multi-label learning* (WSML) [Xu *et al.*, 2018a] is proposed. It is worth mentioning that semi-supervised MLL is a special case of WSML when the observed label sets of some instances are empty. Figure 1 graphically illustrates the data setting of WSML studied in this paper.

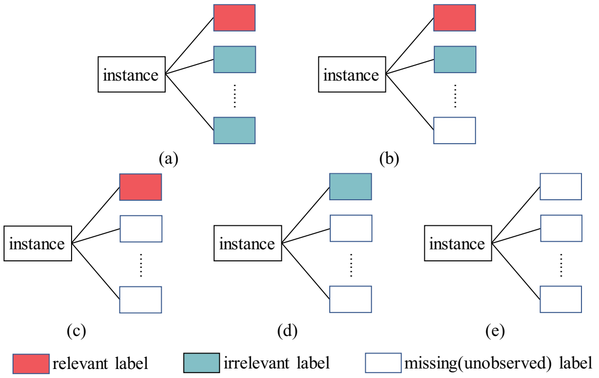

Figure 1: The data setting of weakly supervised multi-label learning. There may be five kinds of instances in the training set: (a) instances with no missing labels; (b) instances with both relevant and irrelevant labels; (c) instances with only relevant labels; (d) instances with only irrelevant labels; (e) instances with empty observed label set.

Formally speaking, let $\mathcal{X} = \mathbb{R}^d$ be the $d$-dimensional feature space and $\mathcal{Y} = \{1, -1\}^c$ be the $c$-dimensional label space. Given the WSML training set $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid 1 \leq i \leq n\}$, where $\boldsymbol{x}_i \in \mathcal{X}$ is the $d$-dimensional feature vector and $\boldsymbol{y}_i \in \mathcal{Y}$ is the corresponding $c$-dimensional binary label vector with some entries missing. The task of WSML is to learn a predictive model $h : \mathcal{X} \to \mathcal{Y}$ from $\mathcal{D}$, which maps an instance to a relevant label set rather than a single label.

There are some previous work studying the WSML problem. Some approaches formulate this task transductively, i.e., assume the training instances with missing labels include the test instances [Goldberg *et al.*, 2010; Xu *et al.*, 2013; Wu *et al.*, 2014], and others are inductive, i.e., try to assign a set of proper labels for any unseen instance [Yu *et al.*, 2014; Wu *et al.*, 2018; Dong *et al.*, 2018; Zhu *et al.*, 2018]. The accessible labeling information of the training set is *logical*, i.e., each label is either regarded to be relevant or irrelevant. Accordingly, the previous approaches deal with the WSML problem by exploiting such incomplete logical labels directly.

Nonetheless, logical label only reflects the absolute relationship between a label and an instance, but ignores the rel-

[*] Corresponding author.

ative importance of each label to an instance, i.e., logical label is essentially a simplification of the semantic information. Furthermore, the missing labels worsen the inherent semantic deficiency of the logical training set, i.e., the exploitable information becomes less. To handle such problem, we assign description degrees to each label to enrich the information, which is similar to *label distribution learning* (LDL) [Geng, 2016]. Moreover, we adopt *generalized description degree* (GDD) [Xu *et al.*, 2019] to accommodate more flexibility to WSML data. The GDDs of all the labels constitute a data form called *generalized label distribution* (GLD) for a particular instance. Specifically, GDD simulates the relative importance in two aspects:

- $d_{ij} \in (0, 1)$ represents the relevant degree, which varies among the relevant labels. For example, a multi-scenery image may exhibit different region size for each scenery, so the relevant degree of each scenery label to the image is different.
- $d_{ij} \in (-1, 0)$ denotes the irrelevant degree of the irrelevant label. For example, for an instance "chair", the irrelevant degree of label "television" is larger than the label "sofa".

A key yet under-addressed issue is that GLD is not explicitly available in the training set. It needs to be somehow recovered from the training set, a process which is named as *Label enhancement* (LE) [Xu *et al.*, 2018b]. Accordingly, a novel two-stage approach named *Weakly Supervised Multi-label Learning via Label Enhancement* (WSMLLE) is proposed under the inductive setting. The basic strategy of WSMLLE is to recover GLDs by leveraging structural information modeled by sparse reconstruction in the feature space, where the local correlations in the label space are incorporated into an alternative optimization procedure. After that, the multi-label predictive model is induced from the training instances with the GLDs based on tailored multivariate regression techniques. To our best knowledge, it is the first attempt to unify the complement of missing labels and the recovery of GLDs into the same framework. Comprehensive experimental studies clearly validate the effectiveness of WSMLLE.

The rest of this paper is organized as follows. Firstly, related works on WSML are briefly reviewed. Secondly, the technical details of the proposed approach is presented. Thirdly, the comparative experimental results on different tasks are reported. Finally, we conclude this paper.

## 2 Related Work

This work is related to two branches of studies, WSML and LE. WSML algorithms have been proposed in recent years, which were pioneered by [Sun *et al.*, 2010]. Then, many algorithms are designed subsequently and they can be roughly categorized into two groups based on the different assumptions about test data.

A straightforward strategy to this problem is formulating it transductively, i.e., assuming the training data with missing labels include the test data. For example, Goldberg et al. [Goldberg *et al.*, 2010] concatenate features and labels and apply the matrix completion technique to it. Xu et al. [Xu *et al.*, 2013] utilize the side information to accelerate matrix completion and develop strong theoretical guarantees. Wu et al. [Wu *et al.*, 2014] recover the missing labels through label consistency and label smoothness.

For better generalization to unseen instances, algorithms working in an inductive learning setting are also proposed, i.e., the test data are unknown when learning the predictive model. For example, Yu et al. [Yu *et al.*, 2014] assume there is a linear relationship between the feature and label matrices. Wu et al. [Wu *et al.*, 2018] decompose the whole label matrix as the sum of a sparse matrix and a low-rank matrix. Dong et al. [Dong *et al.*, 2018] consider both instance similarity and label similarity and further employ ensemble learning to improve robustness. Zhu et al. [Zhu *et al.*, 2018] exploit both global and local label correlations through learning a latent label representation.

LE aims at recovering the label distributions from the logical labels in the training set, which is conceptualized by Xu et al. [Xu *et al.*, 2018b]. And they propose a dedicated algorithm named GLLE. There are also other algorithms with similar function to GLLE, such as FCM [El Gayar *et al.*, 2006] and KM [Jiang *et al.*, 2006], which build the membership degrees for the labels, whereas LP [Li *et al.*, 2015] and ML [Hou *et al.*, 2016] establish the relationship between instances and labels by graph to enhance the label distributions.

## 3 The Proposed Approach

### 3.1 Problem Definition

As shown in Section 1, suppose $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d \times n}$ is the feature matrix and $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n] \in \{1, 0, -1\}^{c \times n}$ is the corresponding label matrix with randomly missing entries. $y_{ij} = 1$ indicates the $i$-th label is relevant to the $j$-th instance and $y_{ij} = -1$ otherwise. $y_{ij} = 0$ means the relationship between $i$-th label and $\boldsymbol{x}_j$ is unknown. The task of WSML is to learn a predictive model $h : \mathcal{X} \to \mathcal{Y}$ from the training set $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid 1 \leq i \leq n\}$. $\boldsymbol{Y}$ is called observation matrix and $\boldsymbol{D}$ is the reasonable GLD matrix recovered from $\boldsymbol{X}$ and $\boldsymbol{Y}$.

### 3.2 GLD Matrix Recovery

Our goal is to recover a complete GLD matrix $\boldsymbol{D}$ that satisfies the following three properties simultaneously. (1) The information in $\boldsymbol{D}$ is inherited from the observation matrix $\boldsymbol{Y}$. (2) The recovery of $\boldsymbol{D}$ leverages the structural information transferred from the feature space. (3) Label correlations conduce to estimate a certain label.

Accordingly, we formulate this problem as

$$\min_{\boldsymbol{D}} \mathcal{L}(\boldsymbol{D}) + \lambda \mathcal{R}(\boldsymbol{D}), \tag{1}$$

where $\mathcal{L}$ is a loss function and $\mathcal{R}$ is the function to mine the latent information from the feature space and the label space. $\lambda$ is the parameter trading off the two terms.

Since the information in $\boldsymbol{D}$ is inherited from the observation matrix $\boldsymbol{Y}$, the loss function is set as:

$$\mathcal{L}(\boldsymbol{D}) = -\sum_{i=1}^{c} \sum_{j=1}^{n} sign(Y_{ij} D_{ij}), \tag{2}$$

where the sign function $sign(x)$ equals to 1 if $x > 0$, $-1$ if $x < 0$ and 0 if $x = 0$, which enforces the sign consistency of $\boldsymbol{D}$ with $\boldsymbol{Y}$. We adopt the hyperbolic tangent function to approximate the sign function for solvability:

$$\mathcal{L}(\boldsymbol{D}) = -\sum_{i=1}^{c}\sum_{j=1}^{n} tanh(Y_{ij}D_{ij}). \tag{3}$$

To characterize the underlying structure of the feature space, a weighted graph $\mathcal{G} =< \mathcal{V}, \mathcal{E}, \boldsymbol{W} >$ is constructed, where $\mathcal{V}$ is the vertex set corresponding to the training instances, $\mathcal{E}$ is the sparsely connected edge set, and $\boldsymbol{W} = [\boldsymbol{w}_1, \dots, \boldsymbol{w}_n]$ is the weight matrix encoding the structural information. According to the smoothness assumption that instance can be reconstructed by a linear combination of its neighboring instances, we instantiate $\boldsymbol{W}$ by modeling the relationship between one instance and all the other instances via sparse reconstruction, in which $W_{ij}$ is regarded as the influence of $\boldsymbol{x}_j$ over $\boldsymbol{x}_i$. $\boldsymbol{W}$ can be obtained columnwisely by solving the minimization problem:

$$\min_{\boldsymbol{w}_i} \ \|\boldsymbol{X}_{-i}\boldsymbol{w}_i - \boldsymbol{x}_i\|_2^2 + \alpha\|\boldsymbol{w}_i\|_1, \tag{4}$$

where $\boldsymbol{X}_{-i}$ is the feature matrix excluding $\boldsymbol{x}_i$, i.e., $\boldsymbol{X}_{-i} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_{i-1}, \boldsymbol{x}_{i+1}, \dots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times(n-1)}$. The second term guarantees the sparsity and $\alpha$ is the tradeoff parameter. Here we solve Eq.(4) by Alternating Direction Method of Multiplier (ADMM) [Ghadimi *et al.*, 2015].

Suppose the structural information in the feature space is preserved in the label space, i.e., the influence of $\boldsymbol{x}_j$ over $\boldsymbol{x}_i$ can be transferred to $\boldsymbol{d}_j$ over $\boldsymbol{d}_i$. Then the GLD matrix can be learned through minimizing the reconstruction error in the label space $L(\boldsymbol{D}) = \sum_{i=1}^{n}\|\boldsymbol{D}_{-i}\boldsymbol{w}_i - \boldsymbol{d}_i\|_2^2$. This is a standard quadratic programming problem which can be efficiently solved by any off-the-shelf QP toolbox. Thus, from the perspective of feature structure, the recovered GLD is formulated as $\boldsymbol{d}_j \approx \boldsymbol{D}\boldsymbol{w}_j$.

Moreover, the label correlations conduce to the recovery of GLD matrix. Intuitively, the correlative labels tend to have similar description degrees. However, estimating label correlations from incomplete training set is noisy. Therefore, instead of specifying any label correlation matrix, we optimize the Laplacian matrices together with the GLD matrix iteratively. Inspired by the investigation of [Zhu *et al.*, 2018] that label correlations may vary among regions, we partition the whole training set into $g$ regions $\{\boldsymbol{X}^1, \dots, \boldsymbol{X}^g\}$ by clustering, where $\boldsymbol{X}^k \in \mathbb{R}^{d\times n_k}$ has $n_k$ instances. Let $\boldsymbol{Y}^k$ and $\boldsymbol{D}^k$ be the observation submatrix and GLD submatrix corresponding to $\boldsymbol{X}^k$, and $\boldsymbol{R}^k$ be the Laplacian matrix of region $k$. Also, $\boldsymbol{W}^k$ is calculated region by region. Accordingly, from the perspective of label correlations, the GLD in region $k$ can be denoted as $\boldsymbol{d}_j^k \approx \boldsymbol{R}^k\boldsymbol{d}_j^k$. Synthesizing the feature structure and label correlations, the GLD matrix is characterized by $\boldsymbol{D}^k \approx \boldsymbol{R}^k\boldsymbol{D}^k\boldsymbol{W}^k$, which leads to the following function to minimize:

$$\mathcal{R}(\boldsymbol{D}) = \|\boldsymbol{D}^k - \boldsymbol{R}^k\boldsymbol{D}^k\boldsymbol{W}^k\|_F^2. \tag{5}$$

Substituting Eq.(3) and Eq.(5) into Eq.(1), we have the following objective function:

$$\begin{aligned}\min_{\boldsymbol{D},\boldsymbol{L}} \quad & -\sum_{k=1}^{g}\Big(\sum_{i=1}^{c}\sum_{j=1}^{n} tanh(Y_{ij}D_{ij}) - \\ & \lambda\|\boldsymbol{D}^k - \boldsymbol{L}^k\boldsymbol{L}^{k\top}\boldsymbol{D}^k\boldsymbol{W}^k\|_F^2\Big) \qquad (6)\\ \text{s.t.} \quad & \operatorname{diag}(\boldsymbol{L}^k\boldsymbol{L}^{k\top}) = \boldsymbol{1}, k = 1, 2, ..., g \\ & 0 < D_{ij}^2 < 1, \forall 1 \le i \le n, \forall 1 \le j \le c,\end{aligned}$$

where $\boldsymbol{R}^k$ is replaced by $\boldsymbol{L}^k\boldsymbol{L}^{k\top}$ and a constraint $\operatorname{diag}(\boldsymbol{L}^k\boldsymbol{L}^{k\top}) = \boldsymbol{1}$ is added for avoiding the trivial solution $\boldsymbol{R}^k = 0$ and guaranteeing $\boldsymbol{R}^k$ to be a normalized Laplacian matrix.

Eq.(6) can be solved by alternating minimization. In each iteration, fix one of $\{\boldsymbol{D}, \boldsymbol{L}\}$ and update the other with gradient descent. Specifically, with $\boldsymbol{L}^k$s (i.e., $\boldsymbol{R}^k$s) fixed, the target function of $\boldsymbol{D}^k$ yields:

$$\begin{aligned}T(\boldsymbol{D}^k) \ = \ & -\sum_{i=1}^{c}\sum_{j=1}^{n} tanh(Y_{ij}^kD_{ij}^k) + \qquad (7)\\ & \lambda\mathrm{tr}\Big[(\boldsymbol{D}^k - \boldsymbol{R}^k\boldsymbol{D}^k\boldsymbol{W}^k)^\top(\boldsymbol{D}^k - \boldsymbol{R}^k\boldsymbol{D}^k\boldsymbol{W}^k)\Big].\end{aligned}$$

The gradient of the objective w.r.t. $\boldsymbol{D}^k$ is

$$\begin{aligned}\nabla_{\boldsymbol{D}^k} \ = \ & (tanh(\boldsymbol{Y}\circ\boldsymbol{D})\circ tanh(\boldsymbol{Y}\circ\boldsymbol{D}) - \boldsymbol{1})\circ\boldsymbol{Y} + \qquad (8)\\ & 2\lambda(\boldsymbol{D} - \boldsymbol{R}\boldsymbol{D}\boldsymbol{W} - \boldsymbol{R}^\top\boldsymbol{D}\boldsymbol{W}^\top + \boldsymbol{R}^\top\boldsymbol{R}\boldsymbol{D}\boldsymbol{W}\boldsymbol{W}^\top),\end{aligned}$$

where $\circ$ is the Hadamard product. The superscripts $k$ of $\{\boldsymbol{Y}, \boldsymbol{D}, \boldsymbol{W}, \boldsymbol{R}\}$ on the right-hand side of this equality are omitted for brevity. When $\boldsymbol{D}^k$s are fixed, Eq.(6) reduces to:

$$\begin{aligned}T(\boldsymbol{L}^k) \ = \ & \mathrm{tr}\Big[(\boldsymbol{D}^k - \boldsymbol{L}^k\boldsymbol{L}^{k\top}\boldsymbol{D}^k\boldsymbol{W}^k)^\top \qquad (9)\\ & (\boldsymbol{D}^k - \boldsymbol{L}^k\boldsymbol{L}^{k\top}\boldsymbol{D}^k\boldsymbol{W}^k)\Big],\end{aligned}$$

so the gradient w.r.t. $\boldsymbol{L}^k$ is

$$\nabla_{\boldsymbol{L}^k} = 4(\boldsymbol{L}\boldsymbol{L}^\top\boldsymbol{D}\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{D}^\top - \boldsymbol{D}\boldsymbol{W}^\top\boldsymbol{D}^\top)\boldsymbol{L}, \tag{10}$$

where the superscripts $k$ of $\{\boldsymbol{D}, \boldsymbol{W}, \boldsymbol{L}\}$ are also omitted. To satisfy the constraint $\operatorname{diag}(\boldsymbol{L}_i\boldsymbol{L}_i^\top) = \boldsymbol{1}$, each row of $\boldsymbol{L}_i$ is projected onto the unit norm ball after each update.

### 3.3 Predictive Model Induction

Replace the observed labels with the recovered GLDs, the training set can be transformed into $\widetilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \boldsymbol{d}_i)\}$. As the output $\boldsymbol{d}_i$ is real-valued, it is natural to induce the predictive model by employing multi-output regression techniques. Here we proposed a predictor based on multi-regression support vector machines. We tailor the following regression problem:

$$\begin{aligned}\Omega(\boldsymbol{\Theta}, \boldsymbol{b}) \ = \ & \frac{1}{2}\sum_{j=1}^{c}\|\boldsymbol{\theta}_j\|_2^2 + \beta_1\sum_{i=1}^{n}\Omega_1(u_i) + \qquad (11)\\ & \beta_2\sum_{i=1}^{n}\sum_{j=1}^{c}\Omega_2(v_{ij}) + \beta_3\sum_{i=1}^{n}\Omega_3(w_i),\end{aligned}$$

| Datasets | FCM | KM | LP | ML | GLLE | WSMLLE |
|---|---|---|---|---|---|---|
| SJAFFE | 0.154(2) | 0.465(6) | 0.178(4) | 0.211(5) | 0.174(3) | **0.107(1)** |
| Natural Scene | 0.363(5) | 0.463(6) | 0.320(2) | 0.330(3) | **0.311(1)** | 0.343(4) |
| Yeast-spoem | 0.181(3) | 0.482(6) | 0.411(5) | 0.280(4) | 0.136(2) | **0.084(1)** |
| Yeast-spo5 | 0.161(3) | 0.465(6) | 0.294(4) | 0.403(5) | 0.121(2) | **0.092(1)** |
| Yeast-dtt | 0.105(2) | 0.492(6) | 0.278(4) | 0.442(5) | 0.112(3) | **0.070(1)** |
| Yeast-cold | 0.127(2) | 0.487(6) | 0.291(4) | 0.438(5) | 0.128(3) | **0.076(1)** |
| Yeast-heat | 0.109(3) | 0.426(6) | 0.183(4) | 0.398(5) | 0.081(2) | **0.056(1)** |
| Yeast-spo | 0.114(3) | 0.425(6) | 0.174(4) | 0.401(5) | 0.089(2) | **0.069(1)** |
| Yeast-diau | 0.087(2) | 0.373(6) | 0.146(4) | 0.351(5) | 0.100(3) | **0.046(1)** |
| Yeast-elu | 0.046(3) | 0.210(6) | 0.073(4) | 0.195(5) | 0.046(2) | **0.022(1)** |
| Yeast-cdc | 0.048(3) | 0.199(6) | 0.070(4) | 0.184(5) | 0.044(2) | **0.021(1)** |
| Yeast-alpha | 0.038(2) | 0.163(6) | 0.055(4) | 0.147(5) | 0.040(3) | **0.014(1)** |
| SBU_3DFE | 0.158(3) | 0.467(6) | 0.183(4) | 0.359(5) | 0.144(2) | **0.118(1)** |
| Movie | 0.221(4) | 0.462(6) | 0.184(3) | 0.419(5) | 0.167(2) | **0.141(1)** |

Table 1: Recovery results (value(rank)) of each comparing algorithm with 60% M.Ratio on Cheb ↓.

| Datasets | FCM | KM | LP | ML | GLLE | WSMLLE |
|---|---|---|---|---|---|---|
| SJAFFE | 0.866(3) | 0.680(6) | 0.865(4) | 0.836(5) | 0.874(2) | **0.943(1)** |
| Natural Scene | 0.588(6) | 0.671(5) | 0.745(2) | 0.718(3) | **0.752(1)** | 0.671(4) |
| Yeast-spoem | 0.916(3) | 0.876(5) | 0.803(6) | 0.878(4) | 0.951(2) | **0.981(1)** |
| Yeast-spo5 | 0.931(3) | 0.781(6) | 0.853(4) | 0.784(5) | 0.957(2) | **0.975(1)** |
| Yeast-dtt | 0.938(3) | 0.687(6) | 0.808(4) | 0.692(5) | 0.950(2) | **0.981(1)** |
| Yeast-cold | 0.944(2) | 0.701(6) | 0.809(4) | 0.706(5) | 0.942(3) | **0.977(1)** |
| Yeast-heat | 0.940(3) | 0.639(6) | 0.835(4) | 0.646(5) | 0.956(2) | **0.979(1)** |
| Yeast-spo | 0.913(3) | 0.655(6) | 0.853(4) | 0.661(5) | 0.947(2) | **0.968(1)** |
| Yeast-diau | 0.944(2) | 0.635(6) | 0.862(4) | 0.642(5) | 0.921(3) | **0.983(1)** |
| Yeast-elu | 0.951(2) | 0.580(6) | 0.855(4) | 0.588(5) | 0.949(3) | **0.988(1)** |
| Yeast-cdc | 0.946(3) | 0.578(6) | 0.850(4) | 0.595(5) | 0.950(2) | **0.988(1)** |
| Yeast-alpha | 0.948(3) | 0.575(6) | 0.865(4) | 0.584(5) | 0.949(2) | **0.992(1)** |
| SBU_3DFE | 0.855(3) | 0.677(6) | 0.849(4) | 0.744(5) | 0.890(2) | **0.935(1)** |
| Movie | 0.789(4) | 0.757(6) | 0.896(3) | 0.761(5) | 0.899(2) | **0.921(1)** |

Table 2: Recovery results (value(rank)) of each comparing algorithm with 60% M.Ratio on Cosine ↑.

| Criterion | FCM | KM | LP | ML | GLLE | WSMLLE |
|---|---|---|---|---|---|---|
| Cheb | 4.14/2.86 | 5.86/6.00 | 2.86/3.86 | 4.64/4.79 | 2.14/2.29 | 1.36/1.21 |
| Clark | 3.64/2.64 | 5.79/6.00 | 3.57/4.07 | 4.71/4.64 | 1.93/2.43 | 1.36/1.21 |
| Canber | 3.43/2.64 | 5.93/6.00 | 3.71/4.07 | 4.71/4.64 | 1.93/2.43 | 1.29/1.21 |
| KL | 3.86/2.71 | 5.86/5.93 | 3.21/3.86 | 4.71/4.79 | 2.00/2.43 | 1.36/1.29 |
| Cosine | 3.93/3.07 | 5.80/5.86 | 3.00/3.93 | 4.64/4.79 | 2.21/2.14 | 1.43/1.21 |
| Intersec | 3.60/2.93 | 5.93/5.93 | 3.47/3.86 | 4.73/4.79 | 1.87/2.29 | 1.40/1.21 |
| Average | 3.77/2.81 | 5.86/5.95 | 3.30/3.94 | 4.69/4.74 | 2.01/2.34 | 1.37/1.22 |

Table 3: Average ranks (with 0% M.Ratio/with 60% M.Ratio) of each comparing algorithm on 14 datasets.

where $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_c]$ is weight matrix and $\boldsymbol{b} = [b_1, \ldots, b_c]^\top$ is bias vector. $\beta_1$, $\beta_2$ and $\beta_3$ balance the model complexity (the first term) and the empirical regression loss (the rest of terms). Specifically, the second term penalizes the case where the model predictions do not fit the GLDs:

$$\Omega_1(u_i) = \begin{cases} 0, & u_i < \epsilon \\ (u_i - \epsilon)^2, & u_i \geq \epsilon \end{cases} . \tag{12}$$

$u_i = ||\boldsymbol{e}_i|| = \sqrt{\boldsymbol{e}_i^\top \boldsymbol{e}_i}$ with $\boldsymbol{e}_i = \boldsymbol{d}_i - \boldsymbol{\Theta}^\top \varphi(\boldsymbol{x}_i) - \boldsymbol{b}$. $\epsilon$ creates an insensitive zone around the estimate where the loss $u$ less than $\epsilon$ is ignored. The third term restricts the sign consistency:

$$\Omega_2(v_{ij}) = \begin{cases} 0, & v_{ij} \geq 0 \\ -v_{ij}, & v_{ij} < 0 \end{cases} . \tag{13}$$

Here, $v_{ij} = D_{ij} \left( \boldsymbol{\theta}_i^\top \varphi(\boldsymbol{x}_j) + b_j \right)$. The last term ensures the average output from relevant labels to be larger than the average output from irrelevant ones, which has been widely used in inexact MLL algorithms [Cour *et al.*, 2011]:

$$\Omega_3(w_i) = -\left( \frac{1}{|S_i|} \cdot \boldsymbol{1}_{S_i}^\top - \frac{1}{|\overline{S}_i|} \cdot \boldsymbol{1}_{\overline{S}_i}^\top \right) \left( \boldsymbol{\Theta}^\top \varphi(\boldsymbol{x}_i) + \boldsymbol{b} \right), \tag{14}$$

where $S_i$ and $\overline{S}_i$ are the relevant and irrelevant label set of $\boldsymbol{x}_i$ respectively. $\boldsymbol{1}_{S_i}^\top$ ($\boldsymbol{1}_{\overline{S}_i}^\top$) corresponds to a $c$-dimensional vector whose $k$-th entry equals to 1 if $D_{ki} > 0$ ($D_{ki} < 0$) and 0 otherwise.

To minimize $\Omega(\boldsymbol{\Theta}, \boldsymbol{b})$, we use an iterative quasi-Newton method called Iterative Re-Weighted Least Square [Tuia *et al.*, 2011]. According to the Representer Theorem [Smola and Schölkopf, 1998], under fairly general conditions, a learning problem can be expressed as a linear combination of the training examples in the feature space, i.e., $\boldsymbol{\theta}_j = \sum_{i=1}^{n} \varphi(\boldsymbol{x}_i)\kappa_{ij}$, and then kernel trick can be applied.

# 4 Experiments

To evaluate the effectiveness of WSMLLE, extensive experiments are conducted on 14 LDL datasets and 10 MLL datasets compared with several state-of-the-art LE algorithms and WSML algorithms respectively.

## 4.1 LE with Full Labels and Missing Labels

### Experimental Setup

A total of 14 real-world LDL datasets are employed for performance evaluation [1]. The binarization method [Xu *et al.*, 2018b] is adopted to get the logical labels from the real label distributions. Considering the missing ratio (M.Ratio) by randomly dropping $\{0\%, 60\%\}$ logical labels, we recover the label distributions and compare them with the ground-truth label distributions. As mentioned in Section 3, in the training sets, $Y_{ij} \in \{1, 0, -1\}$ where the missing labels are set to 0. The label distributions recovered by all algorithms are normalized before evaluation. All algorithms use the same data setting for each dataset.

According to [Geng, 2016], we choose 6 LDL evaluation metrics, i.e., Chebyshev distance (Cheb), Clark distance (Clark), Canberra metric (Canber), Kullback-Leibler divergence (KL), cosine coefficient (Cosine) and intersection similarity (Intersec). The first four are distance measures and the last two are similarity measures.

WSMLLE is compared with 5 LE approaches reviewed in Section 2, including FCM, KM, LP, ML and GLLE. The parameter $\lambda$ in WSMLLE is chosen among $\{0.01, 0.1, 1\}$ and the number of clusters $g$ is chosen among $\{1, 2, \cdots, 10\}$. The kernel function is Gaussian kernel. The clustering algorithm is k-means. The parameters of the comparing algorithms are the suggested configuration in [Xu *et al.*, 2018b].

### Results

For quantitative analysis, Table 1 and Table 2 tabulate the recovery results on Cheb and Cosine with 60% M.Ratio, and The best results are highlighted in bold face. For each evaluation metric, ↓ indicates the smaller the better while ↑ indicates

[1] http://palm.seu.edu.cn/xgeng/LDL/index.htm#data

| Dataset | Measure | M.Ratio | WSMLLE | GLOCAL | MLML-APPROX | SSWL | WELL | MAX.+RANK-SVM | MAX.+ML-kNN |
|---|---|---|---|---|---|---|---|---|---|
| Emotions | Ranking Loss↓ | 0.2 | 0.213±0.011 | 0.284±0.019● | 0.271±0.010● | **0.207±0.013** | 0.422±0.020● | 0.461±0.020● | 0.456±0.015● |
| | | 0.4 | **0.220±0.008** | 0.298±0.028● | 0.285±0.010● | 0.226±0.015 | 0.426±0.018● | 0.447±0.027● | 0.435±0.029● |
| | | 0.6 | **0.238±0.017** | 0.325±0.029● | 0.306±0.018● | 0.260±0.010● | 0.428±0.019● | 0.464±0.021● | 0.448±0.019● |
| | AveragePre.↑ | 0.2 | **0.754±0.009** | 0.679±0.017● | 0.704±0.011● | 0.753±0.015 | 0.567±0.015● | 0.561±0.012● | 0.569±0.011● |
| | | 0.4 | **0.747±0.008** | 0.666±0.021● | 0.692±0.014● | 0.738±0.018 | 0.566±0.014● | 0.571±0.022● | 0.578±0.018● |
| | | 0.6 | **0.724±0.016** | 0.641±0.025● | 0.674±0.012● | 0.706±0.013● | 0.566±0.015● | 0.555±0.018● | 0.570±0.014● |
| Medical | Ranking Loss↓ | 0.2 | **0.025±0.003** | 0.053±0.009● | 0.065±0.010● | 0.284±0.037● | 0.266±0.025● | 0.107±0.011● | 0.058±0.008● |
| | | 0.4 | **0.026±0.004** | 0.054±0.009● | 0.075±0.010● | 0.333±0.021● | 0.288±0.060● | 0.109±0.011● | 0.061±0.008● |
| | | 0.6 | **0.031±0.006** | 0.071±0.012● | 0.093±0.011● | 0.365±0.023● | 0.271±0.032● | 0.114±0.010● | 0.067±0.010● |
| | AveragePre.↑ | 0.2 | **0.869±0.012** | 0.834±0.016● | 0.749±0.028● | 0.379±0.051● | 0.370±0.024● | 0.503±0.031● | 0.769±0.019● |
| | | 0.4 | **0.860±0.010** | 0.809±0.016● | 0.723±0.025● | 0.341±0.027● | 0.323±0.070● | 0.500±0.031● | 0.762±0.020● |
| | | 0.6 | **0.838±0.009** | 0.739±0.024● | 0.698±0.019● | 0.325±0.029● | 0.275±0.066● | 0.491±0.032● | 0.746±0.025● |
| Scene | Ranking Loss↓ | 0.2 | **0.072±0.004** | 0.100±0.004● | 0.073±0.006 | 0.126±0.005● | 0.404±0.012● | 0.085±0.005● | 0.099±0.006● |
| | | 0.4 | **0.078±0.003** | 0.103±0.003● | 0.080±0.006 | 0.144±0.006● | 0.421±0.009● | 0.087±0.005● | 0.101±0.007● |
| | | 0.6 | **0.083±0.004** | 0.110±0.006● | 0.088±0.010 | 0.175±0.008● | 0.437±0.013● | 0.088±0.006● | 0.102±0.008● |
| | AveragePre.↑ | 0.2 | **0.872±0.005** | 0.833±0.006● | 0.870±0.009 | 0.795±0.007● | 0.488±0.014● | 0.843±0.007● | 0.834±0.006● |
| | | 0.4 | **0.863±0.006** | 0.828±0.005● | 0.862±0.009 | 0.772±0.007● | 0.475±0.009● | 0.840±0.008● | 0.832±0.008● |
| | | 0.6 | 0.854±0.006 | 0.819±0.008● | **0.855±0.011** | 0.734±0.011● | 0.465±0.012● | 0.839±0.011● | 0.830±0.009● |
| Yeast | Ranking Loss↓ | 0.2 | **0.172±0.004** | 0.358±0.003● | 0.172±0.004 | 0.201±0.004● | 0.201±0.005● | 0.191±0.004● | 0.190±0.005● |
| | | 0.4 | **0.176±0.003** | 0.362±0.006● | 0.179±0.004● | 0.203±0.010● | 0.204±0.006● | 0.192±0.004● | 0.193±0.005● |
| | | 0.6 | **0.168±0.002** | 0.368±0.006● | 0.189±0.004● | 0.214±0.009● | 0.211±0.006● | 0.193±0.004● | 0.197±0.005● |
| | AveragePre.↑ | 0.2 | **0.756±0.006** | 0.596±0.004● | 0.756±0.005 | 0.709±0.006● | 0.721±0.006● | 0.732±0.004● | 0.735±0.006● |
| | | 0.4 | **0.751±0.005** | 0.590±0.006● | 0.750±0.005 | 0.707±0.017● | 0.718±0.006● | 0.731±0.005● | 0.731±0.007● |
| | | 0.6 | **0.766±0.005** | 0.580±0.006● | 0.742±0.005● | 0.690±0.017● | 0.714±0.007● | 0.731±0.006● | 0.724±0.008● |
| Arts | Ranking Loss↓ | 0.2 | **0.123±0.003** | 0.161±0.004● | 0.185±0.009● | 0.174±0.003● | 0.189±0.007● | 0.158±0.004● | 0.153±0.002● |
| | | 0.4 | **0.133±0.002** | 0.161±0.006● | 0.203±0.007● | 0.192±0.005● | 0.185±0.002● | 0.159±0.004● | 0.155±0.003● |
| | | 0.6 | **0.147±0.003** | 0.164±0.005● | 0.229±0.008● | 0.225±0.007● | 0.192±0.004● | 0.159±0.004● | 0.157±0.003● |
| | AveragePre.↑ | 0.2 | **0.622±0.007** | 0.594±0.006● | 0.507±0.008● | 0.557±0.007● | 0.418±0.007● | 0.483±0.010● | 0.492±0.007● |
| | | 0.4 | **0.609±0.005** | 0.586±0.007● | 0.489±0.009● | 0.534±0.005● | 0.431±0.015● | 0.475±0.012● | 0.490±0.007● |
| | | 0.6 | **0.589±0.005** | 0.570±0.007● | 0.464±0.010● | 0.496±0.007● | 0.418±0.009● | 0.479±0.011● | 0.484±0.007● |
| Rcv1subset1 | Ranking Loss↓ | 0.2 | **0.047±0.001** | 0.061±0.002● | 0.133±0.007● | 0.075±0.002● | 0.165±0.003● | / | 0.127±0.004● |
| | | 0.4 | **0.050±0.002** | 0.062±0.002● | 0.150±0.006● | 0.091±0.002● | 0.169±0.003● | / | 0.129±0.004● |
| | | 0.6 | **0.059±0.002** | 0.067±0.002● | 0.185±0.008● | 0.123±0.003● | 0.172±0.003● | / | 0.135±0.003● |
| | AveragePre.↑ | 0.2 | **0.601±0.005** | 0.575±0.004● | 0.494±0.005● | 0.559±0.005● | 0.268±0.003● | / | 0.450±0.009● |
| | | 0.4 | **0.592±0.005** | 0.570±0.004● | 0.467±0.006● | 0.529±0.005● | 0.266±0.003● | / | 0.441±0.010● |
| | | 0.6 | **0.576±0.006** | 0.560±0.004● | 0.423±0.007● | 0.484±0.005● | 0.262±0.003● | / | 0.430±0.008● |
| Rcv1subset2 | Ranking Loss↓ | 0.2 | **0.057±0.001** | 0.065±0.002● | 0.134±0.008● | 0.075±0.002● | 0.166±0.002● | / | 0.127±0.003● |
| | | 0.4 | **0.062±0.001** | 0.066±0.001● | 0.153±0.008● | 0.092±0.002● | 0.169±0.002● | / | 0.130±0.003● |
| | | 0.6 | **0.063±0.001** | 0.071±0.002● | 0.185±0.009● | 0.120±0.003● | 0.173±0.002● | / | 0.134±0.002● |
| | AveragePre.↑ | 0.2 | **0.603±0.002** | 0.566±0.006● | 0.500±0.006● | 0.569±0.005● | 0.313±0.003● | / | 0.462±0.005● |
| | | 0.4 | **0.593±0.003** | 0.564±0.006● | 0.474±0.009● | 0.541±0.005● | 0.312±0.003● | / | 0.456±0.006● |
| | | 0.6 | **0.581±0.004** | 0.551±0.008● | 0.438±0.008● | 0.498±0.007● | 0.310±0.003● | / | 0.444±0.004● |
| Rcv1subset3 | Ranking Loss↓ | 0.2 | **0.049±0.002** | 0.066±0.002● | 0.125±0.008● | 0.077±0.002● | 0.166±0.005● | / | 0.129±0.003● |
| | | 0.4 | **0.052±0.002** | 0.068±0.002● | 0.147±0.005● | 0.094±0.003● | 0.168±0.006● | / | 0.133±0.003● |
| | | 0.6 | **0.059±0.002** | 0.072±0.002● | 0.177±0.006● | 0.122±0.004● | 0.170±0.008● | / | 0.139±0.003● |
| | AveragePre.↑ | 0.2 | **0.595±0.004** | 0.562±0.004● | 0.514±0.006● | 0.560±0.003● | 0.315±0.005● | / | 0.468±0.007● |
| | | 0.4 | **0.590±0.003** | 0.557±0.004● | 0.488±0.004● | 0.535±0.005● | 0.313±0.005● | / | 0.461±0.007● |
| | | 0.6 | **0.579±0.003** | 0.544±0.006● | 0.451±0.008● | 0.493±0.005● | 0.311±0.005● | / | 0.446±0.007● |
| Rcv1subset4 | Ranking Loss↓ | 0.2 | **0.044±0.000** | 0.061±0.001● | 0.115±0.006● | 0.068±0.002● | 0.141±0.002● | / | 0.111±0.002● |
| | | 0.4 | **0.047±0.001** | 0.062±0.001● | 0.135±0.012● | 0.084±0.003● | 0.144±0.001● | / | 0.115±0.003● |
| | | 0.6 | **0.053±0.001** | 0.066±0.002● | 0.160±0.006● | 0.113±0.004● | 0.147±0.002● | / | 0.121±0.002● |
| | AveragePre.↑ | 0.2 | **0.627±0.004** | 0.584±0.006● | 0.536±0.004● | 0.585±0.005● | 0.361±0.003● | / | 0.495±0.005● |
| | | 0.4 | **0.615±0.002** | 0.582±0.006● | 0.509±0.012● | 0.559±0.005● | 0.359±0.002● | / | 0.489±0.008● |
| | | 0.6 | **0.053±0.004** | 0.570±0.009● | 0.474±0.011● | 0.518±0.006● | 0.356±0.003● | / | 0.476±0.008● |
| Rcv1subset5 | Ranking Loss↓ | 0.2 | **0.049±0.002** | 0.067±0.002● | 0.124±0.007● | 0.072±0.003● | 0.158±0.002● | / | 0.125±0.004● |
| | | 0.4 | **0.053±0.002** | 0.068±0.002● | 0.145±0.005● | 0.088±0.002● | 0.161±0.003● | / | 0.129±0.004● |
| | | 0.6 | **0.061±0.002** | 0.073±0.004● | 0.174±0.005● | 0.118±0.004● | 0.163±0.004● | / | 0.135±0.004● |
| | AveragePre.↑ | 0.2 | **0.601±0.005** | 0.569±0.007● | 0.518±0.008● | 0.570±0.004● | 0.333±0.006● | / | 0.471±0.009● |
| | | 0.4 | **0.595±0.005** | 0.566±0.007● | 0.492±0.008● | 0.544±0.005● | 0.332±0.006● | / | 0.464±0.010● |
| | | 0.6 | **0.578±0.005** | 0.550±0.009● | 0.455±0.006● | 0.501±0.005● | 0.329±0.005● | / | 0.455±0.010● |

Table 4: Predictive results of each comparing algorithm (mean±std). The best results are highlighted in bold face, and the ● indicates whether WSMLLE is statistically superior to the comparing algorithm.

the larger the better. The results on other evaluation metrics are similar and are not shown due to page limitation. The tables show that WSMLLE achieves the best performance in most cases excluding Natural Scene. This is because GDDs of relevant labels are very close and the GDDs of irrelevant labels are almost equal, but WSMLLE cannot exert its advantage fully.

The average rank of each algorithm on all datasets is shown in Table 3. As can be seen that WSMLLE achieves optimal (lowest) average rank in terms of all evaluation metrics. WSMLLE ranks $1st$ in 81.0% cases across all evaluation metrics with 0% M.Ratio, while ranks $1st$ in 92.9% cases with $60\%$ M.Ratio. The success of WSMLLE can be attributed to the loosened contraint and simultaneous learning of the recovered label distributions, weighted graph encoding the structural information and Laplacian matrices encoding the label correlations. Specifically, WSMLLE loosens the least squares loss constraint to sign constraint, which accommodates more flexibility to description degrees. The weighted graph utilizes the topological structure of the feature space. And the learning of Laplacian matrices circumvents the difficulty of specifying label correlations manually. Overall, the above experimental results clearly show that WSMLLE can effectively recover the label distributions from the logical labels.

## 4.2 WSML with Missing Labels

### Experimental Setup

To thoroughly evaluate the classification performance of learning from weakly supervised data, we conduct comparisons across 10 real-world datasets [2] [3] with $\{20\%, 40\%, 60\%\}$ M.Ratio. Half of the instances in each dataset are randomly chosen as the training set while the other half as the test set. To reduce statistical variability, the mean results and the standard deviation over 10 independent repetitions are recorded. All algorithms use the same data setting for each dataset.

7 widely-used multi-label measures are employed for evaluating from various aspects, including one-error, coverage, ranking loss, average precision, instance-AUC, macro-averaging F1 and micro-averaging F1.

WSMLLE is compared with 4 state-of-the-art WSML algorithms described briefly in Section 2: GLOCAL [Zhu *et al.*, 2018], MLML-APPROX [Wu *et al.*, 2014], SSWL [Dong *et al.*, 2018], WELL [Sun *et al.*, 2010], and 2 representative MLL algorithms: RANK-SVM [Elisseeff and Weston, 2002] and ML-$k$NN [Zhang and Zhou, 2007]. The setting in the first stage of WSMLLE is in accordance with Section 4.1, and $\beta_1$, $\beta_2$ and $\beta_3$ chosen among $\{0.1, 1, 10\}$. For the comparing algorithms, parameter configurations suggested in the literatures are used, i.e., GLOCAL: tradeoff parameter $\lambda = 1$; MLML-APPROX: class similarity parameter $\eta = 10$, sample similarity parameters $k = 20$ and $h = 7$; SSWL: tradeoff parameters $\alpha$, $\beta$ and $\zeta$ chosen among $\{10^{-2}, \cdots, 10^{2}\}$; WELL: tradeoff parameters $\alpha = 100$, $\beta = 10$, and $\gamma$ chosen among $\{10^{0}, \cdots, 10^{4}\}$; RANK-SVM: RBF kernel with $h = 0.01$; ML-$k$NN: neighbor size $k = 10$, smoothing parameter $s = 1$. What needs to be pointed out is that RANK-SVM and ML-$k$NN cannot handle the missing labels directly, so the missing labels must be recovered at first, here we use MAXIDE [Xu *et al.*, 2013] and the resultant combinations are represented as MAX.+RANK-SVM and MAX.+ML-$k$NN.

### Results

We roughly organize the 10 datasets with 5 being regular-scale (the first five in Table 4, each of which includes no more than 5000 instances) and 5 being large-scale (the last five in Table 4, each of which includes more than 5000 instances). RANK-SVM has difficulty in large-scale datasets so it is not applied to the large-scale datasets. Due to page limitation, only parts of the most representative evaluation metrics are listed in Tabel 4 while other results are similar.

The two-tailed *t*-tests at the $5\%$ significance level are performed. Based on the experimental results of comparative studies, it is impressive to observe that:

- Among these 1155 statistical tests (5 regular-scale datasets×3 kinds of M.Ratio×7 evaluation metrics×6 comparing algorithms + 5 large-scale datasets×3×7×5 comparing algorithms excluding RANK-SVM), WSMLLE achieves superior performance against the comparing approaches in $95.2\%$ cases.
- On regular-scale datasets, WSMLLE is significantly better in $93.3\%$ cases, and on large-scale datasets, WSMLLE is signficiantly better in $97.5\%$ cases.
- On large-scale datasets, WSMLLE significantly outperforms all the comparing algorithms in terms of ranking loss, average precision, instance-AUC and macro-averaging F1, and is comparable to all the comparing algorithms in terms of other evaluation metrics.

Across all the datasets, WSMLLE performs better against other WSML algorithms, especially on large-scale datasets. It is because that the structural information and label correlations on large-scale datasets are easier to be fully utilized. These results further demonstrate that WSMLLE has strong capability to recover the missing labels and enhances the reasonable GLDs, and verify the effectiveness of these enhanced GLDs, which can offer richer labeling information to the subsequent learning process. Thus, the proposed approach can make more accurate multi-label predictions.

# 5 Conclusion

In this paper, the problem of WSML is studied where an innovative two-stage approach named WSMLLE is proposed. Different from existing approaches, WSMLLE considers GLDs which are not explicitly available in the training sets. WSMLLE recovers the missing labels and enhances the GLDs from the logical labels simultaneously through utilizing the structural information in the feature space and label correlations learned from the label space. Then a tailored predictive model is induced to make multi-label prediction. Comprehensive experiments over a range of tasks clearly validate the reasonability of the recovered GLDs and the effectiveness of these GLDs for weakly supervised multi-label learning.

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

[3] http://www.kecl.ntt.co.jp/as/members/ueda/yahoo.tar

## Acknowledgements

This research was supported by the National Key Research & Development Plan of China (No. 2017YFB1002801), the National Science Foundation of China (61622203), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Collaborative Innovation Center of Wireless Communications Technology.

## References

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

[Dong *et al.*, 2018] Hao-Chen Dong, Yu-Feng Li, and Zhi-Hua Zhou. Learning from semi-supervised weak-label data. In *Proceedings of the 32nd AAAI Conference on Aritificial Intelligence*, New Orleans, LA, 2018.

[El 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 *Proceedings of the International Conference on Artificial Neural Network in Pattern Recognition*, pages 67–80, Ulm, Germany, 2006.

[Elisseeff and Weston, 2002] André Elisseeff and Jason Weston. A kernel method for multi-labelled classification. In *Advances in Neural Information Processing Systems*, pages 681–687, Vancouver, Canada, 2002.

[Geng, 2016] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, 2016.

[Ghadimi *et al.*, 2015] Euhanna Ghadimi, André Teixeira, Iman Shames, and Mikael Johansson. Optimal parameter selection for the alternating direction method of multipliers (admm): quadratic problems. *IEEE Transactions on Automatic Control*, 60(3):644–658, 2015.

[Goldberg *et al.*, 2010] Andrew Goldberg, Ben Recht, Junming Xu, Robert Nowak, and Jerry Zhu. Transduction with matrix completion: Three birds with one stone. In *Advances in Neural Information Processing Systems*, pages 757–765, Vancouver, Canada, 2010.

[Hou *et al.*, 2016] Peng Hou, Xin Geng, and Min-Ling Zhang. Multi-label manifold learning. In *Proceedings of the 30th AAAI Conference on Aritificial Intelligence*, pages 1680–1686, Phoenix, AZ, 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 *Proceedings of the IEEE International Conference on Data Mining*, pages 251–260, Atlantic City, NJ, 2015.

[Smola and Schölkopf, 1998] Alex J Smola and Bernhard Schölkopf. *Learning with kernels*, volume 4. Citeseer, 1998.

[Sun *et al.*, 2010] Yu-Yin Sun, Yin Zhang, and Zhi-Hua Zhou. Multi-label learning with weak label. In *Proceedings of the 24th AAAI Conference on Aritificial Intelligence*, 2010.

[Tuia *et al.*, 2011] Devis Tuia, Jochem Verrelst, Luis Alonso, Fernando Pérez-Cruz, and Gustavo Camps-Valls. Multi-output support vector regression for remote sensing biophysical parameter estimation. *IEEE Geoscience and Remote Sensing Letters*, 8(4):804–808, 2011.

[Wu *et al.*, 2014] Baoyuan Wu, Zhilei Liu, Shangfei Wang, Bao-Gang Hu, and Qiang Ji. Multi-label learning with missing labels. In *Proceedings of the 22nd International Conference on Pattern Recognition*, pages 1964–1968, Stockholm, Sweden, 2014.

[Wu *et al.*, 2018] Baoyuan Wu, Fan Jia, Wei Liu, Bernard Ghanem, and Siwei Lyu. Multi-label learning with missing labels using mixed dependency graphs. *International Journal of Computer Vision*, pages 1–22, 2018.

[Xu *et al.*, 2013] Miao Xu, Rong Jin, and Zhi-Hua Zhou. Speedup matrix completion with side information: Application to multi-label learning. In *Advances in Neural Information Processing Systems*, pages 2301–2309, Lake Tahoe, NV, 2013.

[Xu *et al.*, 2018a] Miao Xu, Gang Niu, Bo Han, Ivor W Tsang, Zhi-Hua Zhou, and Masashi Sugiyama. Matrix co-completion for multi-label classification with missing features and labels. *arXiv preprint arXiv:1805.09156*, 2018.

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

[Xu *et al.*, 2019] Ning Xu, Jia-Qi Lv, and Xin Geng. Partial label learning via label enhancement. In *Proceedings of the 33rd AAAI Conference on Aritificial Intelligence*, page in press, Honolulu, HI, 2019.

[Yu *et al.*, 2014] Hsiang-Fu Yu, Prateek Jain, Purushottam Kar, and Inderjit Dhillon. Large-scale multi-label learning with missing labels. In *Proceedings of the International Conference on Machine Learning*, pages 593–601, Beijing, China, 2014.

[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.

[Zhu *et al.*, 2018] Yue Zhu, James T Kwok, and Zhi-Hua Zhou. Multi-label learning with global and local label correlation. *IEEE Transactions on Knowledge and Data Engineering*, 30(6):1081–1094, 2018.