Partial Label Learning by Semantic Difference Maximization

Lei Feng[1,2] and **Bo An**[1]
[1]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2]Alibaba-NTU Singapore Joint Research Institute, Singapore
feng0093@e.ntu.edu.sg, boan@ntu.edu.sg

Abstract

Partial label learning is a weakly supervised learning framework, in which each instance is provided with multiple candidate labels while only one of them is correct. Most of the existing approaches focus on leveraging the instance relationships to disambiguate the given noisy label space, while it is still unclear whether we can exploit potentially useful information in label space to alleviate the label ambiguities. This paper gives a positive answer to this question for the first time. Specifically, if two instances do not share any common candidate labels, they cannot have the same ground-truth label. By exploiting such dissimilarity relationships from label space, we propose a novel approach that aims to maximize the latent semantic differences of the two instances whose ground-truth labels are definitely different, while training the desired model simultaneously, thereby continually enlarging the gap of label confidences between two instances of different classes. Extensive experiments on artificial and real-world partial label datasets show that our approach significantly outperforms state-of-the-art counterparts.

1 Introduction

Partial label (PL) learning [Jin and Ghahramani, 2003; Cour *et al.*, 2011] belongs to the family of weakly supervised learning frameworks. It aims to deal with the problem that each instance is provided with a set of candidate labels, only one of which is the ground-truth label. Partial label learning is also termed as *ambiguous label learning* [Hüllermeier and Beringer, 2006; Zeng *et al.*, 2013; Chen *et al.*, 2014; Chen *et al.*, 2018] and *superset label learning* [Liu and Dietterich, 2012; Liu and Dietterich, 2014; Gong *et al.*, 2018]. As a result of the difficulty in collecting perfect data with completely correct labels in many real-world scenarios, partial label learning has been applied to various domains. Examples include automatic face naming [Zeng *et al.*, 2013], object detection [Liu and Dietterich, 2012], and web mining [Luo and Orabona, 2010].

Formally speaking, let $\mathcal{X} = \mathbb{R}^n$ be the n-dimensional feature space and $\mathcal{Y} = \{1, 2, \cdots, l\}$ be the label space including l labels. Suppose the PL dataset is denoted by $\mathcal{D} = \{(\mathbf{x}_i, S_i)\}_{i=1}^m$ where $\mathbf{x}_i \in \mathcal{X}$ is an n-dimensional feature vector and $S_i \subseteq \mathcal{Y}$ is the corresponding candidate label set where the ground-truth label y_i must be in this candidate label set, i.e., $y_i \in S_i$. Given such data, the goal of partial label learning is to train a multi-class classification model $f : \mathcal{X} \to \mathcal{Y}$ that tries to correctly predict the label of a test instance.

Due to the semantic ambiguities conveyed by the label space, the key of partial label learning is to disambiguate the candidate label set, thereby targeting the ground-truth label. To achieve this, most of the existing disambiguation-based approaches normally follow two typical strategies, including the average-based strategy [Hüllermeier and Beringer, 2006; Cour *et al.*, 2011] and identification-based strategy [Jin and Ghahramani, 2003; Liu and Dietterich, 2012; Zhang and Yu, 2015; Zhang *et al.*, 2016; Tang and Zhang, 2017; Gong *et al.*, 2018]. The average-based strategy treats each candidate label equally, and makes the final prediction by averaging the modeling outputs of candidate labels. The identification-based strategy aims to handle the candidate labels with discrimination, and usually employ an iterative process to gradually update the confidence of each candidate label.

By taking into account the different confidences of candidate labels, the identification-based strategy generally outperforms the average-based strategy, thereby having attracted increasing attention. Most approaches [Zhang and Yu, 2015; Zhang *et al.*, 2016; Feng and An, 2018; Gong *et al.*, 2018] following this strategy normally leverage the topological information in feature space to derive the confidence of each candidate label. Specifically, the conjecture that *nearby (similar) instances are supposed to have the same label* is widely used by these approaches. However, it is still unclear whether we can directly extract useful information in label space to help with the derivation of the confidences of candidate labels. This paper gives a positive answer to this question for the first time. There is a key observation that if two instances do not share any common candidate labels, they cannot have the same ground-truth label. For example, suppose there are two instances $\mathbf{x}_1$ and $\mathbf{x}_2$ whose corresponding label vectors are given as $\mathbf{y}_1 = [1, 1, 0, 0]$ and $\mathbf{y}_2 = [0, 0, 1, 1]$. Without knowing the ground-truth label of the each instance, we can still easily find that $\mathbf{x}_1$ and $\mathbf{x}_2$ cannot have the same ground-truth label, since they do not share any common candidate labels.

By exploiting such dissimilarity relationships from label space, we propose a novel partial label learning approach called SDIM (**S**emantic **DI**fference **M**aximization), which aims to maximize the latent semantic differences of the two instances whose ground-truth labels are definitely different, while training the desired model simultaneously, thereby continually enlarging the gap of label confidences between two instances of different classes. The effectiveness of SIDM is clearly demonstrated by extensive experiments on 4 artificial and 6 real-world PL datasets.

2 Related Work

The key of effective partial label learning is to disambiguate the candidate labels. Existing disambiguation-based approaches mainly follow two strategies: the average-based strategy and the identification-based strategy.

The average-based strategy assumes that each candidate label contributes equally to model training, and the final prediction is made by averaging the modeling outputs of all the candidate labels. Following this strategy, instance-based approaches [Hüllermeier and Beringer, 2006; Gong *et al.*, 2018] predict the label of a test instance $\mathbf{x}$ by averaging the outputs of its nearest neighbors, i.e., $\arg\max_{y\in\mathcal{Y}} \sum_{\mathbf{x}_i\in\mathcal{N}(\mathbf{x})} \mathbb{I}(y \in S_i)$ where $\mathcal{N}(\mathbf{x})$ is the set of neighbors of $\mathbf{x}$. In addition, parametric approaches aim to train a parametric model [Cour *et al.*, 2011; Zhang *et al.*, 2016] that is able to differentiate the average modeling output of candidate labels $(F(\mathbf{x}_i, y; \theta), y \in S_i)$ from that of non-candidate labels $(F(\mathbf{x}_i, \hat{y}; \theta), \hat{y} \in \hat{S}_i)$, where $\hat{S}_i$ is the set of non-candidate labels and $\hat{y} \in S_i$ is a non-candidate label. Obviously, the average-based strategy is simple and clear, while it may suffer from the problem that without discrimination of candidate labels, the ground-truth label could be overwhelmed by the other false positive labels.

To address the drawback of the average-based strategy, the identification-based strategy tries to handle the candidate labels with discrimination, and derive different confidences of candidate labels. Following this strategy, conventional approaches aim to optimize the objective function according to the maximum likelihood criterion [Jin and Ghahramani, 2003] or the maximum margin criterion [Nguyen and Caruana, 2008]. Recently, there have been increasing interests [Zhang and Yu, 2015; Zhang *et al.*, 2016; Feng and An, 2018; Gong *et al.*, 2018] in leveraging the topological information in feature space to derive the confidence of each candidate label. These approaches normally iteratively update the confidences of candidate labels based on the widely used assumption that *nearby (similar) instances are supposed to have the same label*. One potential drawback of the identification-based strategy lies in that if the differentiated label is a false positive label, it would have a dramatically malignant influence on the follow-up model training. In addition, because of the redundant and noisy features naturally exist in feature space, the extracted topological information may be misleading. Hence there is an important question, i.e., whether we can extract useful information from label space to help with the iterative process of updating the confidences of candidate labels?

In this paper, a novel partial label learning approach called SDIM will be introduced, which provides a positive answer to the above question.

3 Preliminaries

Following the conventional notations used in Introduction, we denote the feature matrix and the label matrix given in the PL dataset by $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m]^\top \in \mathbb{R}^{m\times n}$ and $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_m]^\top \in \{0,1\}^{m\times l}$, respectively. Here, $y_{ij} = 1$ means that the j-th label is in the candidate label set of the instance $\mathbf{x}_i$ (i.e., $j \in S_i$), otherwise the j-th label is a non-candidate label of $\mathbf{x}_i$. In addition, we introduce the partial label confidence matrix $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_m] \in [0,1]^{m\times l}$ where $\mathbf{p}_i$ represents the label confidence vector of $\mathbf{x}_i$.

Note that many approaches [Zhang and Yu, 2015; Zhang *et al.*, 2016; Feng and An, 2018; Gong *et al.*, 2018; Feng and An, 2019] have introduced or defined such partial label confidence matrix $\mathbf{P}$. As the partial label confidence matrix $\mathbf{P}$ is not directly accessible from PL training examples, in this paper, we carefully illustrate some requirements for $\mathbf{P}$. Intuitively, since each candidate label has the potential to be the ground-truth label, the confidence of each candidate label should be in $[0, 1]$. While the confidence of each non-candidate label should be strictly 0, as non-candidate labels can never be the ground-truth label. We can use the constraint $\mathbf{0} \le \mathbf{P} \le \mathbf{Y}$ to compactly represent such logic. Moreover, since there is only one ground-truth label for each instance, competitive relationships naturally exist in the candidate label set. Therefore, we also assume that each label confidence vector $\mathbf{p}_i$ should be normalized, i.e., $\sum_j \mathbf{p}_{ij} = 1$. Such constraint implicitly shows that once the confidence of certain candidate label is enlarged, the confidences of other labels would be decreased. Based on the above descriptions of $\mathbf{P}$, we present the formal definition of partial label confidence matrix as follows.

Definition 1 ***(Partial Label Confidence Matrix).*** *Given the label matrix $\mathbf{Y}$ in the partial label dataset, we define the partial label confidence matrix $\mathbf{P}$ as:*

- ***candidacy**: $\mathbf{0} \le \mathbf{p}_i \le \mathbf{y}_i, \forall i \in [m]$*
- ***normalization**: $\sum_j p_{ij} = 1, \forall i \in [m]$*

where $[m] := \{1, 2, \cdots, m\}$. By compact representation, we also define a partial label simplex as $\mathbf{\Delta} := \{\mathbf{P} \in [0,1]^{m\times l} : \mathbf{P} \le \mathbf{Y}, \mathbf{P}\mathbf{1}_l = \mathbf{1}_m\}$ where $\mathbf{1}_m$ is a vector of size m with all of its elements equal to 1, thus $\mathbf{P} \in \mathbf{\Delta}$.

4 Approach

As stated before, SDIM aims to maximize the latent semantic differences of the two instances whose ground-truth labels are definitely different while model training. Formally, suppose $\mathbf{x}_i$ and $\mathbf{x}_j$ have different ground-truth labels, i.e., $\mathbf{y}_i^\top \mathbf{y}_j = 0$, the gap between the label confidence vectors $\mathbf{p}_i$ and $\mathbf{p}_j$ should be maximized. In this paper, we adopt the widely-used Euclidean distance. Therefore, our goal is to maximize $\|\mathbf{p}_i - \mathbf{p}_j\|_2^2$, if $\mathbf{y}_i^\top \mathbf{y}_j = 0$. Here, we dig more about why our proposed regularization approach can work. Suppose $\mathbf{y}_i = [1, 1, 0, 0]$ and $\mathbf{y}_j = [0, 0, 1, 1]$, we

have $\mathbf{p}_i = [p_{i1}, p_{i2}, 0, 0]$ and $\mathbf{p}_j = [0, 0, p_{j3}, p_{j4}]$ (according to Definition 1). If $p_{i1}, p_{i2} = 0.5$, and $p_{j3}, p_{j4} = 0.5$, we could get $\|\mathbf{p}_i - \mathbf{p}_j\|_2^2 = 1$. While if $p_{i1} = 1, p_{i2} = 0$ and $p_{j3} = 1, p_{j4} = 0$ (the values could be exchanged), we would obtain $\|\mathbf{p}_i - \mathbf{p}_j\|_2^2 = 2$. In this way, it would be clearly observed that $\|\mathbf{p}_i - \mathbf{p}_j\|_2^2$ hits the lowest value when the confidence of each candidate label is equal to 0.5. While $\|\mathbf{p}_i - \mathbf{p}_j\|_2^2$ gradually increases when the confidence of each candidate label approaches to 0 or 1. As a result, such regularization approach would enhance the discrimination abilities to disambiguate the candidate labels, and the obtained confidence vectors $\mathbf{p}_i$ and $\mathbf{p}_j$ would be more confident, thereby reducing the label ambiguities.

For better representation, we introduce an indicating matrix $\mathbf{R} = [r_{ij}]_{m\times m}$, indicating whether two instances **definitely** have different ground-truth labels:

$$r_{ij} = \begin{cases} 1, & \text{if } \mathbf{y}_i^\top \mathbf{y}_j = 0 \\ 0, & \text{if } \mathbf{y}_i^\top \mathbf{y}_j \neq 0 \end{cases} \tag{1}$$

In this way, we present our proposed regularization approach as follows:

$$\max_{\mathbf{P}} \sum_{i=1}^{m}\sum_{j=1}^{m} r_{ij} \|\mathbf{p}_i - \mathbf{p}_j\|_2^2 = \max_{\mathbf{P}} \mathrm{tr}(\mathbf{P}^\top \mathbf{L}\mathbf{P}) \tag{2}$$

where $\mathbf{L} = \mathrm{diag}(\mathbf{R}\mathbf{1}) - \mathbf{R}$ is the Laplacian matrix, and $\mathrm{tr}(\cdot)$ is the trace operator. Note that our proposed regularization approach aims to maximize the convex objective, which is diametrically opposed to the common convex manifold regularization [Belkin *et al.*, 2006] that minimizes the convex objective. In other words, problem (2) is not a convex problem.

By integrating the proposed regularization term into the widely-used model, we obtain the final optimization problem:

$$\min_{\mathbf{W},\mathbf{P}} \frac{1}{2}\|\mathbf{X}\mathbf{W} - \mathbf{P}\|_F^2 + \frac{\lambda}{2}\|\mathbf{W}\|_F^2 - \frac{\beta}{2}\mathrm{tr}(\mathbf{P}^\top \mathbf{L}\mathbf{P}) \tag{3}$$
$$\text{s.t. } \mathbf{P} \in \mathbf{\Delta}$$

where $\mathbf{W} \in \mathbb{R}^{n\times l}$ is the model parameter. Note that in problem (3), we aim to learn from the partial label confidence matrix $\mathbf{P}$ while updating $\mathbf{P}$ simultaneously. Such two tasks would be mutually promoted. Due to the difficulty in optimizing the two variables $\mathbf{W}$ and $\mathbf{P}$ together, we adopt the simple alternating optimization method, which enables us to iteratively optimize one variable with the other fixed.

4.1 Model Training

With $\mathbf{P}$ fixed, problem (3) with respect to $\mathbf{W}$ reduces to:

$$\min_{\mathbf{W}} \frac{1}{2}\|\mathbf{X}\mathbf{W} - \mathbf{P}\|_F^2 + \frac{\lambda}{2}\|\mathbf{W}\|_F^2 \tag{4}$$

which is the common linear regression model. Simple closed-form solution could be easily obtained:

$$\mathbf{W} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{n\times n})^{-1}\mathbf{X}^\top\mathbf{P} \tag{5}$$

where $\mathbf{I}_{n\times n}$ is an identity matrix whose scale is $n \times n$. However, such linear model may not be able to deal with the complex nonlinear case. To solve this problem, we adopt a kernel extension to train a kernel ridge regression model. Specifically, we resort to a feature mapping $\phi(\cdot) : \mathbb{R}^n \to \mathbb{R}^{\mathcal{H}}$, which maps the original feature space ($\mathbf{X}$) to some higher dimensional Hilbert space ($\phi(\mathbf{X})$). By representor theorem [Schölkopf *et al.*, 2002], the model parameter $\mathbf{W}$ can be represented by a linear combination of the input features $\phi(\mathbf{X})$, i.e., $\mathbf{W} = \phi(\mathbf{X})^\top\mathbf{A}$ where $\mathbf{A} = [a_{ij}]_{m\times l}$ is the matrix storing the weights. Hence $\phi(\mathbf{X})\mathbf{W} = \mathbf{K}\mathbf{A}$ where $\mathbf{K} = \phi(\mathbf{X})\phi(\mathbf{X})^\top \in \mathbb{R}^{m\times m}$ is the kernel matrix with each element $k_{ij} = \phi(\mathbf{x}_i)^\top\phi(\mathbf{x}_j) = \kappa(\mathbf{x}_i, \mathbf{x}_j)$, and $\kappa(\cdot,\cdot)$ denotes the kernel function. By incorporating such kernel extension, problem (4) can be stated as:

$$\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{K}\mathbf{A} - \mathbf{P}\|_F^2 + \frac{\lambda}{2}\mathrm{tr}(\mathbf{A}^\top\mathbf{K}\mathbf{A}) \tag{6}$$

where we have used the property of the trace operator, i.e., $\|\mathbf{W}\|_F^2 = \mathrm{tr}(\mathbf{W}^\top\mathbf{W}) = \mathrm{tr}(\mathbf{A}^\top\mathbf{K}\mathbf{A})$. Setting the gradient w.r.t. $\mathbf{A}$ to 0, the closed-form solution is reported as:

$$\mathbf{A} = (\mathbf{K} + \lambda\mathbf{I}_{m\times m})^{-1}\mathbf{P} \tag{7}$$

In this paper, we adopt the popular Gaussian kernel function $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp((\mathbf{x}_i - \mathbf{x}_j)/2\sigma^2)$ with σ set to the averaged pairwise Euclidean distances of instances.

4.2 Confidence Updating

With $\mathbf{A}$ fixed, we denote by the modeling output $\mathbf{Q} = \phi(\mathbf{X})\mathbf{W} = \mathbf{K}\mathbf{A}$, thus problem (3) w.r.t. $\mathbf{P}$ reduces to:

$$\min_{\mathbf{P}} \frac{1}{2}\|\mathbf{P} - \mathbf{Q}\|_F^2 - \frac{\beta}{2}\mathrm{tr}(\mathbf{P}^\top\mathbf{L}\mathbf{P}) \tag{8}$$
$$\text{s.t. } \mathbf{P} \in \mathbf{\Delta}$$

Note that in problem (8), the first term is convex, while the last term is concave. Therefore, problem (8) is a constrained convex-concave problem [Yuille and Rangarajan, 2003; Sriperumbudur and Lanckriet, 2009]. Fortunately, since the constraints are linear, we can directly employ the Convex-ConCave Procedure (CCCP) [Yuille and Rangarajan, 2003] to update $\mathbf{P}$. CCCP can be regarded as a majorization-minimization algorithm [Sriperumbudur *et al.*, 2011; Gong *et al.*, 2018] that optimizes the original nonconvex problem by solving a sequence of convex problems. Specifically, problem (8) can be regarded as the difference between two convex functions $\mathcal{C}_1 = \frac{1}{2}\|\mathbf{P} - \mathbf{Q}\|_F^2$ and $\mathcal{C}_2 = \frac{\beta}{2}\mathrm{tr}(\mathbf{P}^\top\mathbf{L}\mathbf{P})$. In each iteration, $\mathcal{C}_2$ is replaced by its first order Taylor approximation $\tilde{\mathcal{C}}_2$, and problem (8) can be approximated by $\mathcal{C}_1 - \tilde{\mathcal{C}}_2$, which becomes a convex problem. Theoretical analyses show that CCCP converges to a local minima [Sriperumbudur and Lanckriet, 2009].

For our problem, we denote by $\mathbf{P}^{(i)}$ the updated value of $\mathbf{P}$ at the i-th iteration, and linearize $\mathcal{C}_2$ at $\mathbf{P}^{(i)}$ by its Taylor approximation:

$$\begin{aligned}\tilde{\mathcal{C}}_2 &= \frac{\beta}{2}(\mathrm{tr}(\mathbf{P}^{(i)\top}\mathbf{L}\mathbf{P}^{(i)}) + 2(\mathrm{tr}(\mathbf{P}^\top\mathbf{L}\mathbf{P}^{(i)} - \mathbf{P}^{(i)\top}\mathbf{L}\mathbf{P}^{(i)})) \\ &= \beta\mathrm{tr}(\mathbf{P}^\top\mathbf{L}\mathbf{P}^{(i)}) - \frac{\beta}{2}\mathrm{tr}(\mathbf{P}^{(i)\top}\mathbf{L}\mathbf{P}^{(i)})\end{aligned} \tag{9}$$

Algorithm 1 The SDIM Algorithm

Inputs:
$\mathcal{D}$: the PL training set $\{(\mathbf{X}, \mathbf{Y})\}$
λ, β: the regularization parameters
$\mathbf{x}$: the unseen test instance
Output:
y: the predicted label for the test instance $\mathbf{x}$

1: construct the indicating matrix $\mathbf{R} = [r_{ij}]_{m\times m}$ by (1);
2: construct the kernel matrix $\mathbf{K} = [\kappa(\mathbf{x}_i, \mathbf{x}_j)]_{m\times m}$ using Gaussian kernel function;
3: initialize $\mathbf{P}$ by (13);
4: **repeat**
5: update $\mathbf{A} = [a_{ij}]_{m\times l}$ by (7);
6: update $\mathbf{Q} = \mathbf{K}\mathbf{A}$;
7: update $\mathbf{P}$ by solving (12) with a general QP procedure;
8: **until** convergence or the maximum number of iterations.
9: return the predicted label y according to (14).

Since the variable is $\mathbf{P}$, the last term in (9) can be removed, and the approximated objective function for updating $\mathbf{P}$ is:

$$\mathbf{P}^{(i+1)} = \arg\min_{\mathbf{P}} \frac{1}{2}\|\mathbf{P} - \mathbf{Q}\|_F^2 - \beta \mathrm{tr}(\mathbf{P}^\top \mathbf{L}\mathbf{P}^{(i)}) \quad (10)$$
$$\text{s.t. } \mathbf{P} \in \mathbf{\Delta}$$

Here, we will show that problem (10) is actually a standard Quadratic Programming (QP) problem. Let us vectorize $\mathbf{P} \in \mathbb{R}^{m\times l}$, $\mathbf{Q} \in \mathbb{R}^{m\times l}$, and $\mathbf{Y} \in \mathbb{R}^{m\times l}$ to $\hat{\mathbf{p}} \in \mathbb{R}^{ml}$, $\hat{\mathbf{q}} \in \mathbb{R}^{ml}$, and $\hat{\mathbf{y}} \in \mathbb{R}^{ml}$. To copy with the equality constraint using $\hat{\mathbf{p}}$, we specially pick up the indices of $\hat{\mathbf{p}}$ by defining a set $\mathcal{Z} = \{\mathcal{Z}_0, \mathcal{Z}_1, \cdots, \mathcal{Z}_{m-1}\}$ as follows:

$$j \in \mathcal{Z}_i, \text{ if } j\%m = i, \ \forall j \in [ml] \quad (11)$$

Thus problem (10) can be equivalent to:

$$\hat{\mathbf{p}}^{(i+1)} = \arg\min_{\hat{\mathbf{p}}} \frac{1}{2}\hat{\mathbf{p}}\mathbf{H}\hat{\mathbf{p}} + \mathbf{f}^\top \hat{\mathbf{p}} \quad (12)$$
$$\text{s.t. } \sum\nolimits_{j\in\mathcal{Z}_i} \hat{\mathbf{p}}_j = 1, \ \forall \mathcal{Z}_i \in \mathcal{Z}$$
$$\mathbf{0}_{ml} \le \hat{\mathbf{p}} \le \hat{\mathbf{y}}$$

where $\mathbf{H} = \mathbf{I}_{ml\times ml}$, $\mathbf{f} = -\hat{\mathbf{q}} - \beta\mathbf{G}\hat{\mathbf{p}}^{(i)}$, and $\mathbf{G} = \mathbf{I}_{m\times m} \otimes \mathbf{L}$ where $\otimes$ is the Kronecker product. In this way, any off-the-shelf QP toolbox can be used to solve problem (12). By iteratively solving a sequence of QP problems, we can eventually get local optimal $\hat{\mathbf{p}}$. Finally, by reshaping $\hat{\mathbf{p}} \in \mathbb{R}^{ml}$ to $\mathbf{P} \in \mathbb{R}^{m\times l}$, we can obtain the eventually updated partial label confidence matrix $\mathbf{P}$.

Since the variables $\mathbf{A}$ and $\mathbf{P}$ are alternatively updated, we can simply always update $\mathbf{P}$ for one step (for one QP problem) after updating $\mathbf{A}$. Thus the iterative process of updating $\mathbf{P}$ is kept, while the optimization efficiency is improved.

For the 0-th iteration, $\mathbf{P}^{(0)} = [p_{ij}^{(0)}]_{m\times l}$ is initialized as:

$$p_{ij}^{(0)} = \begin{cases} \frac{1}{|S_i|}, & \text{if } j \in S_i \\ 0, & \text{if } j \notin S_i \end{cases} \quad (13)$$

After the completion of the whole optimization process, SDIM gives the predicted label y of the test instance $\mathbf{x}$ by:

$$y = \arg\max_{j\in[l]} \sum_{i=1}^{m} a_{ij}\kappa(\mathbf{x}, \mathbf{x}_i) \quad (14)$$

The pseudo code of SDIM is presented in Algorithm 1.

5 Experiments

In this section, we conduct extensive experiments on artficial and real-world datasets to demonstrate the effectiveness of our proposed approach.

5.1 Comparing Algorithms

We compare our approach with six state-of-the-art partial label learning approaches, each configured with suggested hyperparameters in accordance with the respective literature:

- PLKNN [Hüllermeier and Beringer, 2006]: a kNN approach following the average-based strategy [default configuration: $k \in \{5, 6, \cdots, 10\}$];
- CLPL [Cour *et al.*, 2011]: a convex approach following the average-based strategy [default configuration: SVM with squared hinge loss];
- IPAL [Zhang and Yu, 2015]: an approach following the identification-based strategy that leverages the structural information in feature space [default configuration: $\alpha \in \{0, 0.1, \cdots, 1\}, k \in \{5, 6, \cdots, 10\}$];
- PL-SVM [Nguyen and Caruana, 2008]: a maximum margin approach following the identification-based strategy [default configuration: $\lambda \in \{10^{-3}, \cdots, 10^3\}$];
- PLALOC [Wu and Zhang, 2018]: a disambiguation-free approach that adapts the binary decomposition [default configuration: $\mu = 10$]
- ECOC [Zhang *et al.*, 2017]: a disambiguation-free approach based on the coding-decoding procedure [default configuration: $L = \log_2(l)$]

For our approach, λ is searched in $\{0.001, 0.005, \cdots, 0.5\}$ and β is searched in $\{0.00001, 0.00005, 0.0001, \cdots, 0.1\}$. For all the approaches, parameters are selected by five-fold cross-validation on the training set. For each dataset, we perform ten-fold cross-validation, and report the resulting mean prediction accuracies and the standard deviations.

5.2 Experiments on Controlled UCI Datasets

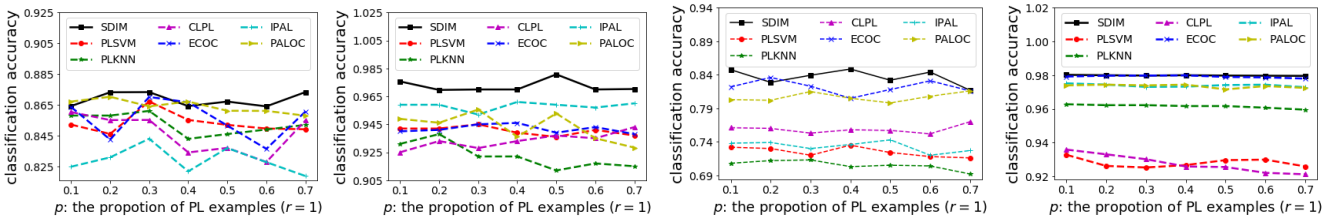
Dataset	ecoli	dermatology	vehicle	usps
Examples	336	366	846	9298
Features	7	34	18	256
Labels	8	6	4	10

Table 1: Characteristics of the controlled UCI datasets.

Table 1 reports the characteristics of four UCI datasets used in our experiments. Following the widely-used controlling protocol [Zhang and Yu, 2015; Zhang *et al.*, 2016;

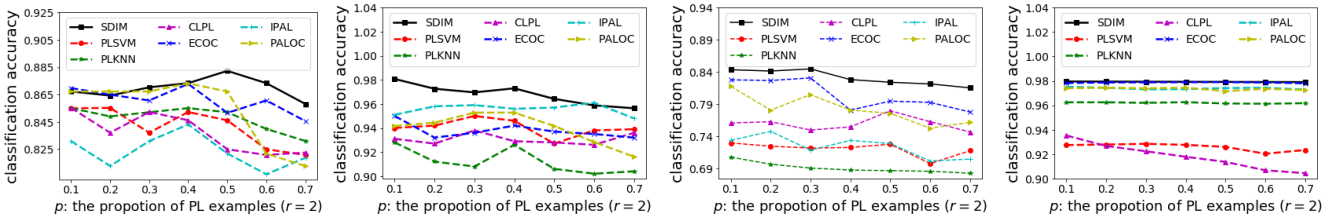
Dataset	Examples	Features	Labels	Avg. CLs	Task Domain
Lost	1122	108	16	2.23	*automatic face naming* [Panis and Lanitis, 2014]
MSRCv2	1758	48	23	3.16	*object classification* [Liu and Dietterich, 2012]
BirdSong	4998	38	13	2.18	*bird song classification* [Briggs *et al.*, 2012]
Soccer Player	17472	279	171	2.09	*automatic face naming* [Zeng *et al.*, 2013]
Yahoo! News	22991	163	219	1.91	*automatic face naming* [Guillaumin *et al.*, 2010]
FG-NET	1002	262	78	7.48	*facial age estimation* [Panis and Lanitis, 2014]

Table 2: Characteristics of real-world partial label datasets.



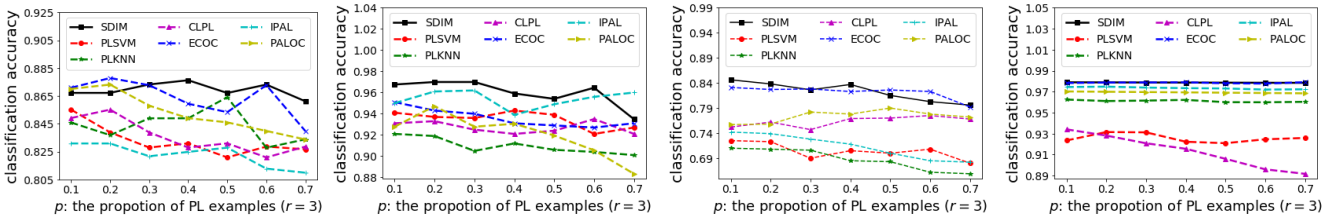
(a) ecoli (b) dermatology (c) vehicle (d) usps

Figure 1: Classification performance on controlled UCI datasets with p ranging from 0.1 to 0.7 ($r = 1$).



(a) ecoli (b) dermatology (c) vehicle (d) usps

Figure 2: Classification performance on controlled UCI datasets with p ranging from 0.1 to 0.7 ($r = 2$).



(a) ecoli (b) dermatology (c) vehicle (d) usps

Figure 3: Classification performance on controlled UCI datasets with p ranging from 0.1 to 0.7 ($r = 3$).

Tang and Zhang, 2017; Feng and An, 2018; Wu and Zhang, 2018], we generate the artificial partial label datasets by using two contorlling parameters p and r. Here, p controls the proportion of instances that have candidate labels, and r controls the number of false positive labels, in other words, $|S_i| = r+1$. All the candidate labels are randomly generated.

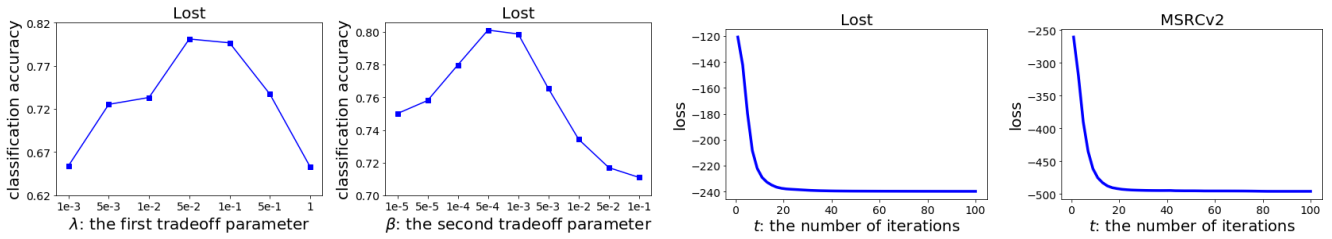
Figure 1, Figure 2, and Figure 3 report the classification accuracy of each approach as p ranges from 0.1 to 0.7 with step size 0.1, when $r = 1$, $r = 2$, and $r = 3$, respectively. As shown in these figures, SDIM outperforms other comparing algorithms in most cases. It is worth noting that candidate labels are randomly generated, there might be the case that many pairs of examples share at least one common candidate labels. In this case, Our proposed method may achieve mediocre performance. However, such case is really rare since experimental results demonstrate that our method can achieve the best performance in more than 75 cases out of the 84 cases.

5.3 Experiments on Real-World Datasets

Table 2 summaries the characteristics of the real-world partial label datasets from various task domains, which also includes the average number of candidate labels per instance (Avg. CLs). Note that the BirdSong dataset is normalized using the Z-scores by convention. The mean accuracy with standard deviation of each approach on each real-world dataset

	SDIM	PLKNN	CLPL	IPAL	PLSVM	PALOC	ECOC
Lost	0.801±0.031	0.432±0.051•	0.742±0.038•	0.678±0.053•	0.729±0.042•	0.629±0.056•	0.703±0.052•
MSRCv2	0.518±0.037	0.417±0.034•	0.413±0.041•	0.529±0.039	0.461±0.046•	0.479±0.042•	0.505±0.027
BirdSong	0.754±0.021	0.648±0.017•	0.632±0.019•	0.713±0.015•	0.663±0.032•	0.711±0.016•	0.740±0.016•
Soccer Player	0.577±0.016	0.495±0.018•	0.368±0.010•	0.541±0.016•	0.464±0.011•	0.537±0.015•	0.537±0.020•
Yahoo! News	0.663±0.013	0.483±0.011•	0.462±0.009•	0.609±0.011•	0.629±0.010•	0.625±0.005•	0.662±0.010•
FG-NET	0.076±0.019	0.039±0.018•	0.063±0.027	0.054±0.030•	0.063±0.029	0.065±0.019	0.040±0.025•
FG-NET(MAE3)	0.466±0.022	0.269±0.045•	0.458±0.022	0.362±0.034•	0.356±0.022•	0.435±0.018•	0.251±0.029•
FG-NET(MAE5)	0.621±0.024	0.438±0.053•	0.596±0.017•	0.540±0.033•	0.479±0.016•	0.609±0.043	0.354±0.038•

Table 3: Classification accuracy of each algorithm on the real-world datasets. Furthermore, •/○ indicates whether SDIM is statistically superior/inferior to the comparing algorithm (t-test at 0.05 significance level for two independent samples).



(a) Varying λ on Lost (b) Varying β on Lost (c) Convergence on Lost (d) Convergence on MSRCv2

Figure 4: Parameter sensitivity and convergence analysis of SDIM on the real-world datasets Lost and MSRCv2.

is reported in Table 3. It is worthy noting that all the approaches achieve extremely poor performance on the FG-NET dataset, because its Avg. CLs is very large. For better evaluation of such task, we resort to conventional mean absolute error (MAE). Specifically, for FG-NET (MAE3/MAE5), a test example is considered correctly classified if the MAE between the predicted age and the real age is no more than 3/5 years. The experimental results of FG-NET(MAE3) and FG-NET(MAE5) are also recorded in Table 3. From Table 3, we can observe that: 1) SDIM statistically outperforms PLKNN on all the real-world datasets; 2) Out of the 48 cases (6 comparing algorithms and 8 tasks), SDIM statistically outperforms other algorithms in 85.4% cases; 3) It is worthy noting that SDIM is never statistically inferior to any comparing algorithms.

Further Analysis

There are two tradeoff parameters for SDIM. We conduct sensitivity analysis by varying one parameter, while keeping the other fixed at the best setting. Figure 4(a) and 4(b) show the performance of SDIM w.r.t. λ and β, respectively. As can be seen from Figure 4(a), when performance of SDIM is relatively poor when λ is too small or too big. Such observation agrees with the intuition that it is important to control the model complexity for avoiding overfitting or underfitting. Note that β controls our proposed regularization term, i.e., the importance of semantic difference maximization. From Figure 4(b), we can also find that when β is very small, the importance of semantic difference maximization is hardly considered, thus SDIM achieves relatively low prediction accuracy. As β increases, the importance of semantic difference maximization will gradually be taken into consideration, hence the prediction accuracy starts to increase. Such observation clearly confirms the effectiveness of our SDIM approach. However, if β is overly large, the classification accuracy will drop. This is because the objective function focuses too much on maximizing the semantic differences while ignoring the importance of model training. It is also worth noting that when β is extremely small (even without the proposed regularization term), SDIM still achieves satisfied performance. Which means, the reduced version of SDIM (i.e., kernel regression with confidence updating) provides a strong baseline. This observation also agrees with [Feng and An, 2018; Feng and An, 2019].

Figure 4(c) and Figure 4(d) illustrate the convergence of SDIM on Lost and MSRCv2, according to the difference of the value of the objective function (3) between two successive iterations. It can be easily observed that the loss converges within a few iterations. Hence the convergence of SDIM is demonstrated.

6 Conclusion

This paper gives the first attempt to leverage the dissimilarity relationships from label space for dealing with label ambiguities. A novel partial label learning approach called SDIM is proposed to maximize the latent semantic differences while training the desired model simultaneously. Extensive experimental results demonstrate the effectiveness of SDIM.

Contrary to conventional approaches that leverage the similarities in feature space to disambiguate the candidate labels, SDIM resorts to the dissimilarities in label space. Hence a question naturally arises on how similarities and dissimilarities can be combined together for enhancing partial label learning performance. We leave this as future work.

Acknowledgements

This work was supported by MOE, NRF, and NTU.

References

[Belkin *et al.*, 2006] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7(11):2399–2434, 2006.

[Briggs *et al.*, 2012] Forrest Briggs, Xiaoli Z Fern, and Raviv Raich. Rank-loss support instance machines for miml instance annotation. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 534–542, 2012.

[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, Ben Sapp, and Ben Taskar. Learning from partial labels. *Journal of Machine Learning Research*, 12(5):1501–1536, 2011.

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

[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.

[Guillaumin *et al.*, 2010] Matthieu Guillaumin, Jakob Verbeek, and Cordelia Schmid. Multiple instance metric learning from automatically labeled bags of faces. *Lecture Notes in Computer Science*, 63(11):634–647, 2010.

[Hüllermeier and Beringer, 2006] Eyke Hüllermeier and Jürgen Beringer. Learning from ambiguously labeled examples. *Intelligent Data Analysis*, 10(5):419–439, 2006.

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

[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.

[Liu and Dietterich, 2014] Li-Ping Liu and Thomas Dietterich. Learnability of the superset label learning problem. In *International Conference on Machine Learning*, pages 1629–1637, 2014.

[Luo and Orabona, 2010] Jie Luo and Francesco Orabona. Learning from candidate labeling sets. In *Neural Information Processing Systems*, pages 1504–1512, 2010.

[Nguyen and Caruana, 2008] Nam Nguyen and Rich Caruana. Classification with partial labels. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 551–559, 2008.

[Panis and Lanitis, 2014] Gabriel Panis and Andreas Lanitis. An overview of research activities in facial age estimation using the fg-net aging database. In *European Conference on Computer Vision*, pages 737–750, 2014.

[Schölkopf *et al.*, 2002] Bernhard Schölkopf, Alexander J Smola, et al. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT press, 2002.

[Sriperumbudur and Lanckriet, 2009] Bharath K Sriperumbudur and Gert RG Lanckriet. On the convergence of the concave-convex procedure. In *Neural Information Processing Systems*, pages 1759–1767, 2009.

[Sriperumbudur *et al.*, 2011] Bharath K Sriperumbudur, David A Torres, and Gert RG Lanckriet. A majorization-minimization approach to the sparse generalized eigenvalue problem. *Machine Learning*, 85(1-2):3–39, 2011.

[Tang and Zhang, 2017] Cai-Zhi Tang and Min-Ling Zhang. Confidence-rated discriminative partial label learning. In *AAAI Conference on Artificial Intelligence*, pages 2611–2617, 2017.

[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 2427–2436, 2018.

[Yuille and Rangarajan, 2003] Alan L Yuille and Anand Rangarajan. The concave-convex procedure. *Neural Computation*, 15(4):915–936, 2003.

[Zeng *et al.*, 2013] Zi-Nan Zeng, Shi-Jie Xiao, Kui Jia, Tsung-Han Chan, Sheng-Hua Gao, Dong Xu, and Yi Ma. Learning by associating ambiguously labeled images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 708–715, 2013.

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

[Zhang *et al.*, 2016] Min-Ling Zhang, Bin-Bin Zhou, and Xu-Ying Liu. Partial label learning via feature-aware disambiguation. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1335–1344, 2016.

[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.