The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Partial Multi-Label Learning

Ming-Kun Xie, Sheng-Jun Huang[*]
College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics
Collaborative Innovation Center of Novel Software Technology and Industrialization
Nanjing 211106, China
{mkxie,huangsj}@nuaa.edu.cn

Abstract

It is expensive and difficult to precisely annotate objects with multiple labels. Instead, in many real tasks, annotators may roughly assign each object with a set of candidate labels. The candidate set contains at least one but unknown number of ground-truth labels, and is usually adulterated with some irrelevant labels. In this paper, we formalize such problems as a new learning framework called partial multi-label learning (PML). To solve the PML problem, a confidence value is maintained for each candidate label to estimate how likely it is a ground-truth label of the instance. On one hand, the relevance ordering of labels on each instance is optimized by minimizing a rank loss weighted by the confidences; on the other hand, the confidence values are optimized by further exploiting structure information in feature and label spaces. Experimental results on various datasets show that the proposed approach is effective for solving PML problems.

Introduction

Multi-label learning (MLL) deals with the problem where each object is assigned with multiple class labels simultaneously (Zhang and Zhou 2014). For example, an image may be annotated with labels *sea*, *sunset*, and *beach*. The task of multi-label learning is to train a classification model that can predict all the relevant labels for unseen instances.

In traditional multi-label studies, a common assumption is that each training instance has been precisely annotated with all of its relevant labels. However, in many applications, this assumption hardly holds because the precise annotation is usually difficult and costly. Instead, annotators may roughly assign each instance a set of candidate labels. In addition to the relevant labels, the candidate set also contains some irrelevant labels. For example, in Figure 1, the image is annotated with a set of candidate labels, which may be the union set of annotations from multiple noisy annotators under the crowdsourcing setting. While the annotation cost is significantly reduced by partial labeling, the learning task becomes much more challenging because the ground-truth labels are mixed with some irrelevant labels, and the number of ground-truth labels is even unknown.

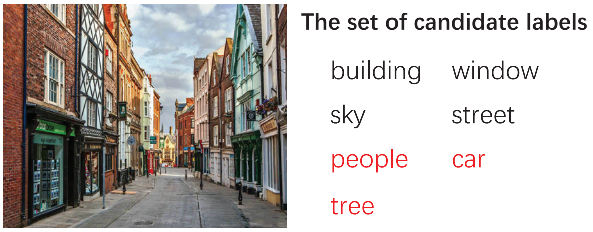


Figure 1: An example of partial multi-label learning. The image is partially labeled by noisy annotators in crowdsourcing. Among the candidate labels, building, window, sky and street are ground-truth labels while people, car and tree are irrelevant labels.

We formalize this learning problem as a new framework called partial multi-label learning (PML). More specifically, PML tries to learn a multi-label model from partially labeled training examples, where each instance is annotated with a set of candidate labels, indicating the following supervised information: a) the candidate set may consist of both relevant and irrelevant labels; b) the number of relevant labels in the candidate set is at least one but unknown; c) labels not in the candidate set are irrelevant to the instance.

PML is a novel learning framework with significant difference from existing settings. There are some studies trying to exploit weak supervision for multi-label learning. For example, semi-supervised MLL(Wang and Tsotsos 2016; Wu et al. 2015; Belkin, Niyogi, and Sindhwani 2006) trains the model based on both unlabeled and precisely labeled examples; MLL with weak label allows missing labels (Sun, Zhang, and Zhou 2010; Bucak, Jin, and Jain 2011; Zhao and Guo 2015). However, these approaches do not consider partial labeling with candidate label sets, and cannot be applied to PML problems. Partial label learning(Cour, Sapp, and Taskar 2011; Szummer and Jaakkola 2001) is similar to PML, but is designed for single-label case, where there is always one ground-truth label in the candidate set. We will discuss the differences between PML and related studies in

[*]This research was supported by JiangsuSF (BK20150754) and NSFC (61503182, 61732006).

more detail in the next section.

Partial multi-label learning degenerates into standard multi-label learning if the ground-truth labels can be identified from the candidate set. Unfortunately, this task is rather challenging or even impossible. Instead, we assume that each candidate label has a confidence of being the ground-truth label, and alternatively optimize the classification model and the confidence values. Specifically, to achieve multi-label classification, we optimize the relevance ordering of label pairs to rank relevant labels before irrelevant labels based on the ground-truth confidences. To optimize the ground-truth confidence of candidate labels, in addition to rank loss minimization, we offer two options to further exploit either the local structure of the feature space or the label correlations. The tasks are formulated into a unified objective function, and can be efficiently solved by alternating optimization of quadratic programming and linear programming. Our empirical study on datasets from diverse domains demonstrates the effectiveness of the proposed approach.

The main contributions are summarized as follows.

- A new learning framework PML is proposed to learn multi-label models from partially labeled data. PML defines a practical learning task and is significantly different from existing multi-label learning settings.
- Two effective algorithms PML-*lc* and PML-*fp* are proposed for solving PML problems. They offer options to optimize the ground-truth confidences of candidate labels by exploiting the structure information from either feature or label space.
- Experiments on various datasets validate the effectiveness of the proposed approaches.

The rest of the paper is organized as follows. We start by a brief review of related works. Then we formulate the problem and propose the algorithm. Next, experimental results are reported, followed by the conclusion.

Related Work

There is a rich body of literature on multi-label learning. The most straightforward approach for multi-label learning is to decompose the task into a set of binary classification problems (Joachims 1998; Boutell et al. 2004). Such methods treat each label independently, and ignore the correlation among labels, which is crucial to multi-label learning (Zhang and Zhou 2014). Later, many studies try to exploit the label correlations. Some of them focus on pairwise correlation (Fürnkranz et al. 2008; Elisseeff and Weston 2001), while some others consider higher order correlation among all labels (Tsoumakas, Katakis, and Vlahavas 2011; Read et al. 2011). Multi-label learning has been successfully applied to various tasks, e.g., image classification (Cabral et al. 2011; Wang et al. 2016; Wu et al. 2015), text categorization (Rubin et al. 2012) and gene function prediction (Elisseeff and Weston 2001).

There are some studies trying to learn multi-label models from weak supervised information. Some approaches try to train the classification model based on both unlabeled and precisely labeled examples. For instance, label propagation based methods are developed for semi-supervised multi-label learning in (Wang and Tsotsos 2016; Kong, Ng, and Zhou 2013); a simultaneous large-margin and subspace learning approach is proposed in (Guo and Schuurmans 2012); and a low-rank mapping based method is introduced in (Jing et al. 2015). Some other approaches focus on the case where some relevant labels are missing. For example, Sun et. al. (2010) propose to study multi-label learning with weak labels based on low-density assumption; Bucak et. al. (2011) propose a ranking based method for multi-label learning with incomplete class assignments; and Yu et. al. (2015) develop a large scale method for multi-label learning with missing labels. There are also some methods trying to learn from both clean and noisy data (Veit et al. 2017). However, these methods do not consider partial labeling with candidate label sets, and cannot solve PML problems.

Partial label learning is similar to our PML problem but is specifically for single-label tasks. It assumes that there is always exactly one ground-truth among the candidate set. Most partial label learning methods employ the strategy of disambiguation, i.e., trying to recover the ground-truth label from the candidate label set. One disambiguation strategy is to assume certain parametric model $F(x, y; \theta)$ and ground-truth label is regarded as latent variable. Here, the latent variable is iteratively refined by optimizing certain objectives, such as the maximum likelihood criterion(Grandvalet and Bengio 2004; Jin and Ghahramani 2002; Liu and Dietterich 2012), or the maximum margin criterion (Yu and Zhang 2017). Another way towards disambiguation is to assume equal importance of each candidate label and then make prediction by averaging their modeling outputs. For parametric models, the averaged output from all candidate labels is distinguished from the outputs from candidate labels (Cour, Sapp, and Taskar 2011). For non-parametric models, the predicted label for unseen instance is determined by averaging the candidate labeling information from its neighboring examples in the PL training set (Hüllermeier and Beringer 2006; Zhang and Yu 2015). The learnability of partial label learning has been studied in (Liu and Dietterich 2014). Compared to partial label learning, PML is much more challenging because the number of ground-truth labels in the candidate set is unknown, which makes disambiguation inapplicable.

The PML Approach

Problem Formulation

Let $\mathcal{X} = \mathcal{R}^d$ denote the input space with d-dimensions features and $\mathcal{Y} = \{y_1, y_2, ..., y_q\}$ be a finite set consisting of q possible class labels. $D = \{(\mathbf{x}_1, \hat{Y}_1), (\mathbf{x}_2, \hat{Y}_2), ..., (\mathbf{x}_m, \hat{Y}_m)\}$ is a training set with m partially labeled instances, where $\mathbf{x}_i \in \mathcal{X}$ is the feature vector and $\hat{Y}_i \subseteq \mathcal{Y}$ is the candidate label set of the i-th example. We further denote by $\bar{Y}_i = \mathcal{Y} \setminus \hat{Y}_i$ the non-candidate label set and Y_i the ground-truth label set for instance $\mathbf{x}_i$. Note that $Y_i \subseteq \hat{Y}_i$, and is unknown, while in traditional multi-label learning, $Y_i = \hat{Y}_i$. We also want to emphasize that even $|Y_i|$ is unknown, which makes partial multi-label learning much more challenging than the single-label case. Because when $|Y_i| = 1$, one could expect to recover the only ground-truth label from $\hat{Y}_i$ by selecting the most likely one. The goal of partial multi-label learning is to train a classifier $h : \mathcal{X} \to 2^{\mathcal{Y}}$ based on the partial multi-label training set D. In most multi-label studies, instead of directly outputting the classifier h, the learning system will produce a real-valued function of the form $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{R}$, which predicts larger values for relevant labels than irrelevant ones. Here we introduce a linear classifier for each label y_k in the form of $f_k(\mathbf{x}_i) = \langle \mathbf{w}_k,\ \mathbf{x}_i \rangle + b_k$, where $\mathbf{w}_k \in \mathcal{R}^d$ and $b_k \in \mathcal{R}$ are the weight vector and bias of the classifier.

Algorithm Detail

To solve PML problems, one straightforward baseline method is to simply take all labels in the candidate set as relevant labels and then directly apply standard multi-label algorithms for model training. Obviously, such methods will be misled by the irrelevant labels in the candidate set. To overcome this problem, we assume that each candidate label has a confidence of being ground-truth label. Formally, we denote by $P_{ik} \in [0, 1]$ the confidence of label y_k being a ground-truth label of instance $\mathbf{x}_i$. Note that if y_k is a non-candidate label, i.e., $y_k \in \bar{Y}_i$, then it is for sure irrelevant to $\mathbf{x}_i$, and thus $P_{ik} = 0$. In contrast, if y_k is a candidate label, then its confidence P_{ik} is unknown and to be learned. For convenience, we further introduce $P = [P_{ik}]_{m\times q}$ to denote the confidence matrix for the whole training set.

In the following part of this subsection, we will firstly introduce the multi-label classification model which incorporates label ranking with the confidence matrix, then present two strategies for optimizing the confidence matrix, and at last summarize the whole procedure of the algorithm.

To achieve multi-label classification, the relevance ordering of label pairs is optimized to rank relevant labels before irrelevant ones on each instance, and then a proper number of top ranked labels are selected as relevant ones. Specifically, we consider the relevance ordering of two kinds of label pairs: 1) the inter-set label pair with one label from the candidate set $\hat{Y}_i$ and the other one from the non-candidate set $\bar{Y}_i$; 2) the intra-set label pair with two labels both from the candidate set $\hat{Y}_i$. For an inter-set label pair $(y_k, y_l) \in \hat{Y}_i \times \bar{Y}_i$, y_k should be ranked before y_l on instance $\mathbf{x}_i$ with a confidence of P_{ik}. The ranks over the whole training set can be optimized by minimizing the ranking loss (Elisseeff and Weston 2001) weighted by the confidences:

$$\begin{aligned} \min_{W,\mathbf{b}} \quad & \sum_{i=1}^{m} \frac{1}{|\hat{Y}_i| \cdot |\bar{Y}_i|} \sum_{(y_k,y_l)\in\hat{Y}_i\times\bar{Y}_i} P_{ik} \cdot \xi_{ikl} \qquad (1) \\ s.t. \quad & \langle \mathbf{w}_k - \mathbf{w}_l, \mathbf{x}_i \rangle + b_k - b_l \geq 1 - \xi_{ikl} \\ & \xi_{ikl} \geq 0 \quad (1 \leq i \leq m,\ (y_k, y_l) \in \hat{Y} \times \bar{Y}), \end{aligned}$$

where ξ_{ikl} is the slack variable of the ranking loss. Obviously, a candidate label with higher ground-truth confidence will be ranked before irrelevant labels with more emphasis. As an extreme case, when $P_{ik} = 0$, it implies that y_k is irrelevant to $\mathbf{x}_i$, and will not contribute any ranking loss.

For an intra-set label pair $(y_k, y_l) \in \hat{Y}_i \times \hat{Y}_i$, the label with higher ground-truth confidence should be ranked before the other one. Similarly, we can have the following optimization problem to minimize the ranking loss of intra-set label pairs over the whole training set:

$$\begin{aligned} \min_{W,\mathbf{b}} \quad & \sum_{i=1}^{m} \frac{2}{|\hat{Y}_i| \cdot |\hat{Y}_i|} \sum_{(y_k,y_l)\in\hat{Y}_i\times\hat{Y}_i} \tilde{P}_{ikl} \cdot \xi_{ikl} \qquad (2) \\ s.t. \quad & \langle \mathbf{w}_k - \mathbf{w}_l, \mathbf{x}_i \rangle + b_k - b_l \geq 1 - \xi_{ikl} \\ & \xi_{ikl} \geq 0 \quad (1 \leq i \leq m, (y_k, y_l) \in \hat{Y} \times \hat{Y}), \end{aligned}$$

where $\tilde{P}_{ikl} = \max(0, P_{ik} - P_{il})$ measures how confident y_k should be ranked before y_l for instance $\mathbf{x}_i$.

Noticing that for any $y_l \in \bar{Y}_i$, we have $P_{il} = 0$, and thus $\tilde{P}_{ikl} = P_{ik} - P_{il} = P_{ik}$. So we can incorporate Eq. 1 and Eq. 2 to consider the ranking loss for both inter-set and intra-set label pairs in a unified objective function:

$$\begin{aligned} \min_{W,\mathbf{b}} \quad & \sum_{k=1}^{q} ||\mathbf{w}_k||^2 + C_1 \sum_{i=1}^{m} \frac{1}{\gamma_i} \sum_{y_k,y_l\in\mathcal{Y}} \tilde{P}_{ikl} \cdot \xi_{ikl} \qquad (3) \\ s.t. \quad & \langle \mathbf{w}_k - \mathbf{w}_l, \mathbf{x}_i \rangle + b_k - b_l \geq 1 - \xi_{ikl} \\ & \xi_{ikl} \geq 0 \quad (1 \leq i \leq m,\ y_k, y_l \in \mathcal{Y}), \end{aligned}$$

where $\tilde{P}_{ikl} = \max(0, P_{ik} - P_{il})$, and $\gamma_i = |\hat{Y}_i| \cdot |\bar{Y}_i| + |\hat{Y}_i|^2/2$ is a constant for normalization on instance $\mathbf{x}_i$. The first term is a regularizer to control the model complexity, and C_1 is a trade-off parameter.

In the above discussions, we assume that the confidence matrix P is given. However, the elements in P corresponding to the candidate labels are unknown. Next, we will show how to optimize the ground-truth confidences for candidate labels of each instance. First of all, the confidences values are expected to be consistent with the model predictions. However, optimizing P solely based on the model prediction may suffer from overfitting given that the model itself is trained according to P. We thus expect to exploit the structure information from data to further guide the optimization of the confidence matrix P. Here, we present two options to regularize the confidence learning.

The first option is to learn P based on label correlations. Specifically, we assume a label correlation matrix $S = [S]_{q\times q}$, where the element S_{kl} denote the correlation between y_k and y_l, and a larger value implies a stronger correlation. It is expected that two labels with strong correlation should share similar confidence values. We thus try to maximize the term $\sum_{k\in\hat{Y}} S_{k\cdot}(P_{ik} \cdot P_{i\cdot})^\top$, where $S_{k\cdot}$ and $P_{i\cdot}$ denote the k/i-th row of S and P, respectively. While there are multiple ways to calculate S, we simply employ the co-occurrence rate of two labels as their correlation (Diplaris et al. 2005). By further incorporating the model predictions, we have the following optimization problem for learning the confidence matrix P:

$$\begin{aligned}\min_{P}\quad & C_1\sum_{i=1}^{m}\frac{1}{\gamma_i}\sum_{y_k,y_l\in\mathcal{Y}}\tilde{P}_{ikl}\cdot\xi_{ikl}\\ & -C_2\sum_{i=1}^{m}\sum_{k\in\hat{Y}_i}S_{k\cdot}(P_{ik}\cdot P_{i\cdot})^\top\\ s.t.\quad & \textstyle\sum_{y_k\in\hat{Y}_i}P_{ik}\geq 1\quad (1\leq i\leq m)\\ & 0\leq P_{ik}\leq 1\quad (y_k\in\hat{Y}_i,\ 1\leq i\leq m)\\ & P_{ik}=0\quad (y_k\in\bar{Y}_i,\ 1\leq i\leq m),\end{aligned}\tag{4}$$

where the first constraint corresponds to the fact there is at least one ground-truth label in multi-label learning. Noticing that the model $\mathbf{w}_k$ and b_k is fixed, we have $\xi_{ikl}=1-\langle\mathbf{w}_k-\mathbf{w}_l,\mathbf{x}_i\rangle-b_k+b_l$ in Eq. 4. Then by combing Eq. 4 with Eq. 3, we have the final objective function for partial multi-label learning with label correlations (PML-*lc* for short):

$$\begin{aligned}\min_{W,\mathbf{b},P}\quad & \sum_{k=1}^{q}||\mathbf{w}_k||^2+C_1\sum_{i=1}^{m}\frac{1}{\gamma_i}\sum_{y_k,y_l\in\mathcal{Y}}\tilde{P}_{ikl}\cdot\xi_{ikl}\\ & -C_2\sum_{i=1}^{m}\sum_{k\in\hat{Y}_i}S_{k\cdot}(P_{ik}\cdot P_{i\cdot})^\top\\ s.t.\quad & \langle\mathbf{w}_k-\mathbf{w}_l,\mathbf{x}_i\rangle+b_k-b_l\geq 1-\xi_{ikl}\\ & \xi_{ikl}\geq 0\quad (1\leq i\leq m,\ y_k,y_l\in\mathcal{Y}),\\ & \textstyle\sum_{y_k\in\hat{Y}_i}P_{ik}\geq 1\quad (1\leq i\leq m)\\ & 0\leq P_{ik}\leq 1\quad (y_k\in\hat{Y}_i,\ 1\leq i\leq m)\\ & P_{ik}=0\quad (y_k\in\bar{Y}_i,\ 1\leq i\leq m).\end{aligned}\tag{5}$$

The second option is to learn P based on feature prototype. Specifically, we assume a feature prototype Q_k for each label y_k, which can be regarded as a representative instance of y_k. In our implementation, we simply calculate Q_k as the average over all instances associated with y_k. Then given an instance $\mathbf{x}_i$, it is more likely to have y_k as the ground-truth label if it has larger similarity with Q_k. This observation motivates us to minimize the term $\sum_{k\in\hat{Y}}P_{ik}\cdot\|\mathbf{x}_i-Q_k\|$. Similar to Eq. 4, we have

$$\begin{aligned}\min_{P}\quad & C_1\sum_{i=1}^{m}\frac{1}{\gamma_i}\sum_{y_k,y_l\in\mathcal{Y}}\tilde{P}_{ikl}\cdot\xi_{ikl}\\ & +C_3\sum_{i=1}^{m}\sum_{y_k\in\hat{Y}_i}P_{ik}\cdot\|\mathbf{x}_i-Q_k\|\\ s.t.\quad & \textstyle\sum_{y_k\in\hat{Y}_i}P_{ik}\geq 1\quad (1\leq i\leq m)\\ & 0\leq P_{ik}\leq 1\quad (y_k\in\hat{Y}_i,\ 1\leq i\leq m)\\ & P_{ik}=0\quad (y_k\in\bar{Y}_i,\ 1\leq i\leq m),\end{aligned}\tag{6}$$

Then by combing Eq. 6 with Eq. 3, we have the final objective function for partial multi-label learning with feature prototypes (PML-*fp* for short):

Algorithm 1 The PML-fp algorithm

```
Input:
 1: Partial label training set $D=\{(\mathbf{x}_i,Y_i)|1\leq i\leq m\}$;
 2: The nonnegative trade-off parameters $C_1$ and $C_3$;
 3: Maximal number of iterations $maxIter$;
Process:
 4: Calculate the feature prototypes $Q_k$ for each label $y_k$;
 5: Initialize the confidence matrix $P$;
 6: $iter \leftarrow 1$;
 7: repeat:
 8:     Optimize $W$ and $\mathbf{b}$ with fixed $P$ by solving Eq. 3;
 9:     Optimize $P$ with fixed $W$ and $\mathbf{b}$ by solving Eq. 6;
10:     $iter \leftarrow iter + 1$
11: until convergence or $iter$ exceeds $maxIter$
12: Output trained model $\mathbf{w}_k$ and $b_k$ for $k=1,\cdots,q$
```

$$\begin{aligned}\min_{W,\mathbf{b},P}\quad & \sum_{k=1}^{q}||\mathbf{w}_k||^2+C_1\sum_{i=1}^{m}\frac{1}{\gamma_i}\sum_{y_k,y_l\in\mathcal{Y}}\tilde{P}_{ikl}\cdot\xi_{ikl}\\ & +C_3\sum_{i=1}^{m}\sum_{y_k\in\hat{Y}_i}P_{ik}\cdot\|\mathbf{x}_i-Q_k\|\\ s.t.\quad & \langle\mathbf{w}_k-\mathbf{w}_l,\mathbf{x}_i\rangle+b_k-b_l\geq 1-\xi_{ikl}\\ & \xi_{ikl}\geq 0\quad (1\leq i\leq m,\ y_k,y_l\in\mathcal{Y}),\\ & \textstyle\sum_{y_k\in\hat{Y}_i}P_{ik}\geq 1\quad (1\leq i\leq m)\\ & 0\leq P_{ik}\leq 1\quad (y_k\in\hat{Y}_i,\ 1\leq i\leq m)\\ & P_{ik}=0\quad (y_k\in\bar{Y}_i,\ 1\leq i\leq m).\end{aligned}\tag{7}$$

We summarize the key steps of PML-*fp* in Algorithm 1. Firstly, the feature prototypes are calculated for each label, and the ground-truth confidence matrix is initialized. Then the optimization problem in Eq. 7 is solved by alternating optimization. Specifically, with fixed P, the last term in Eq. 7 becomes a constant, and the optimization problem is equivalent to Eq. 3, which can be solved by quadratic programming. When optimizing P with fixed W and $\mathbf{b}$, the optimization problem is equivalent to Eq. 6, which can be solved by linear programming. The alternating optimization procedure iterates, and terminates once the objective function converges or $iter$ exceeds a maximal number predefined by users. The process of PML-*lc* is similar to Algorithm 1 except that, at line 4, label correlations are calculated, and at line 9, Eq. 4 is optimized instead of Eq. 6.

In the test phase, a ranking list of labels can be obtained based on the model predictions. Then a threshold is needed to separate relevant and irrelevant labels from the ranking list. This is a common step for label ranking based multi-label classification, and has many existing solutions (Elisseeff and Weston 2001). In our case, one can decide the threshold value as the average of maximum prediction over non-candidate labels, or specify a fixed number as the relevant label set size. We employ the latter method in our experiments for simplicity. We set it as the average number of relevant labels on the training set.

Table 1: Characteristics of the experimental data sets.

Data set	# Instances	# Dim	# Class Labels	# Candidate Labels	Domain
emotions (Trohidis et al. 2008)	593	72	6	3,4,5	music
yeast (Elisseeff and Weston 2001)	2417	103	14	6,7,8,9,10,11,12,13	biology
CAL500 (Turnbull et al. 2008)	500	68	15	6,7,8,9,10,11,12,13	music
genbase (Diplaris et al. 2005)	662	1186	15	6,7,8,9,10,11,12,13	biology
medical (Pestian et al. 2007)	978	1449	15	6,7,8,9,10,11,12,13	text
corel5k (Duygulu et al. 2002)	5000	499	15	6,7,8,9,10,11,12,13	images
delicious (Tsoumakas et al. 2008)	14000	500	15	6,7,8,9,10,11,12,13	text

Table 2: Comparison of PML with state-of-the-art multi-label learning approaches on five evaluation criteria. The best performance and its comparable performances are bolded (statistical significance examined via pairwise t-tests at 95% significance level).

Data	# C.L	PML-*lc*	PML-*fp*	RankSVM	BSVM	ML-*k*NN	LIFT
Hamming loss (the smaller, the better)							
Emotions	4	**.247 ± .000**	.252 ± .000	.578 ± .000	.603 ± .001	.662 ± .000	.675 ± .001
Yeast	10	.215 ± .000	**.215 ± .000**	.617 ± .000	.696 ± .000	.694 ± .000	.684 ± .000
CAL500		**.314 ± .000**	**.315 ± .000**	.637 ± .000	.687 ± .000	.686 ± .000	.698 ± .000
Genbase		**.018 ± .000**	**.018 ± .000**	.783 ± .000	.849 ± .001	.874 ± .000	.902 ± .000
Medical		.069 ± .000	**.069 ± .000**	.682 ± .000	.661 ± .000	.914 ± .000	.891 ± .000
Corel5K		.151 ± .000	**.142 ± .000**	.745 ± .000	.734 ± .000	.886 ± .000	.887 ± .000
Delicious		**.289 ± .000**	**.289 ± .000**	.610 ± .000	.688 ± .000	.706 ± .000	.698 ± .000
Ranking loss (the smaller, the better)							
Emotions	4	**.202 ± .000**	.212 ± .001	.349 ± .001	.272 ± .001	.377 ± .001	.375 ± .003
Yeast	10	.189 ± .000	**.189 ± .000**	.201 ± .000	.361 ± .000	.200 ± .000	.210 ± .000
CAL500		.329 ± .000	.328 ± .000	.410 ± .000	.408 ± .000	.352 ± .000	**.321 ± .000**
Genbase		**.007 ± .000**	.008 ± .000	.014 ± .000	.059 ± .001	.031 ± .000	.106 ± .005
Medical		**.113 ± .000**	**.113 ± .000**	.188 ± .000	.214 ± .000	.268 ± .002	.162 ± .001
Corel5K		.383 ± .000	**.315 ± .000**	.422 ± .001	.420 ± .000	.334 ± .000	.316 ± .000
Delicious		**.274 ± .000**	**.274 ± .000**	.314 ± .000	.363 ± .000	.342 ± .000	.333 ± .000
One error (the smaller, the better)							
Emotions	4	**.307 ± .001**	**.322 ± .002**	.531 ± .001	.407 ± .004	.447 ± .003	.506 ± .004
Yeast	10	**.245 ± .000**	.249 ± .000	.257 ± .000	.646 ± .001	.253 ± .000	.258 ± .000
CAL500		.450 ± .005	**.446 ± .006**	.621 ± .002	.740 ± .002	.510 ± .001	.452 ± .001
Genbase		**.044 ± .000**	.045 ± .001	.115 ± .002	.189 ± .005	.050 ± .001	.295 ± .014
Medical		.426 ± .002	**.423 ± .002**	.441 ± .005	.652 ± .002	.579 ± .002	.529 ± .003
Corel5K		.782 ± .000	.714 ± .000	.822 ± .000	.855 ± .000	.749 ± .000	**.703 ± .000**
Delicious		**.399 ± .000**	**.398 ± .000**	.437 ± .000	.594 ± .000	.489 ± .000	.525 ± .000
Coverage (the smaller, the better)							
Emotions	4	**.336 ± .003**	.347 ± .007	.448 ± .004	.398 ± ..004	.489 ± .005	.492 ± .017
Yeast	10	.492 ± .002	**.488 ± .002**	.511 ± .001	.646 ± .002	.502 ± .000	.523 ± .002
CAL500		.650 ± .007	**.649 ± .007**	.674 ± .006	.677 ± .003	.684 ± .018	.661 ± .019
Genbase		**.020 ± .001**	**.021 ± .001**	.028 ± .003	.072 ± .012	.055 ± .010	.127 ± .077
Medical		**.130 ± .004**	**.130 ± .004**	.134 ± .006	.228 ± .008	.284 ± .030	.178 ± .012
Corel5K		.469 ± .002	.410 ± .001	.504 ± .016	.501 ± .002	.429 ± .005	**.409 ± .000**
Delicious		**.587 ± .000**	.587 ± .000	.606 ± .000	.648 ± .001	.659 ± .000	.645 ± .001
Average precision (the greater, the better)							
Emotions	4	**.769 ± .001**	.762 ± .001	.625 ± .000	.702 ± .001	.619 ± .001	.612 ± .003
Yeast	10	**.738 ± .000**	**.738 ± .000**	.725 ± .000	.511 ± .000	.727 ± .000	.714 ± .000
CAL500		.567 ± .000	.568 ± .000	.481 ± .000	.442 ± .000	.546 ± .000	**.581 ± .000**
Genbase		**.969 ± .000**	.967 ± .000	.928 ± .001	.859 ± .003	.939 ± .001	.766 ± .010
Medical		.695 ± .001	.697 ± .001	.685 ± .002	.510 ± .001	.526 ± .002	.609 ± .002
Corel5K		.355 ± .000	**.420 ± .000**	.321 ± .000	.303 ± .000	.396 ± .000	**.423 ± .000**
Delicious		**.611 ± .000**	**.611 ± .000**	.581 ± .000	.502 ± .000	.543 ± .000	.538 ± .000

Experiments

Settings

PML is a new learning framework, and there is no method can be directly applied to PML problems. To examine the effectiveness of the proposed algorithms PML-*lc* and PML-*fp*, we compare with multi-label learning methods, which regard all candidate labels as relevant. The following state-of-art methods are compared: RankSVM (Elisseeff and Weston

2001), ML-kNN (Zhang and Zhou 2007), BSVM (Boutell et al. 2004) and LIFT (Zhang and Wu 2015).

For PML-*lc*, the label correlation is extracted according to the method in (Diplaris et al. 2005). For PML-*fp*, the feature prototype is defined by averaging features of training instances associated with a specific label. For both PML-*fp* and PML-*lc*, $C1$ is fixed to 1 as default on all datasets. C_2 is selected from $\{1, 2, \cdots, 10\}$, and C_3 is selected from $\{1,$ $10,$ $100\}$ with regard to the performance on *hamming loss*. The influences of parameters are presented in the following content. For other methods, parameters are selected as suggested in the corresponding literatures.

There are different criteria for evaluating the performances of multi-label learning. In our experiments, we employ five commonly used criteria *hamming loss*, *one error*, *coverage*, *ranking loss* and *average precision*. More details about these criteria can be found in (Zhang and Zhou 2014).

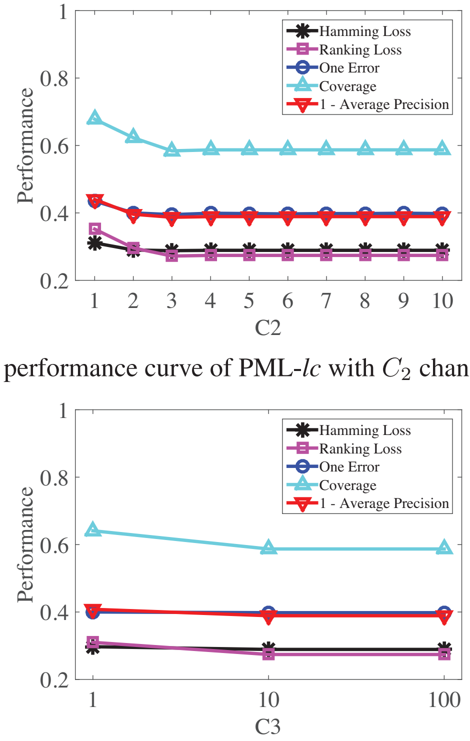
We perform the experiments on seven datasets. These data sets spanned a broad range of applications: *corel5k* for image annotation, *CAL500* and *emotions* for music classification, *yeast* for gene function prediction, *genbase* for protein classification, *medical* for text categorization and *delicious* for web categorization. For each data set, several statistics are used to depict its characteristics. Specifically, we illustrate *number of instances*, *number of classes*, *number of candidate labels* and *domain* for each data set at Table 1. Here *number of candidate labels* lists some options for the size of candidate label set. For each instance, we randomly pick some irrelevant labels to construct the candidate set with ground-truth labels. We also did some pre-processing to facilitate the partial labeling. Specifically, rare labels are filtered out to keep at most 15 labels, and instances without any relevant label are filtered out.

Results

Due to page limit, we cannot report all results with every possible size of the candidate label set. Instead, we report the detailed results for a specific candidate set size (the median of all optional sizes), and the statistical results for all optional set sizes. Specifically, the median set size is 4 for *emotions* and 10 for the other datasets.

The detailed results are presented in Table 2. When comparing the PML approach (either PML-*lc* or PML-*fp*) with other methods, our algorithms show significant superiority. They achieve the best performance in most cases. PML-*lc* and PML-*fp* are comparable with each other. Among the four compared multi-label approaches, LIFT shows some superiority, and achieves the best performance of 2 criteria on *CAL500* and 3 criteria on *Corel5K*, while loses for the other cases. It is worth noting that our PML methods simply use a linear model for each label. It is expected to achieve better performance if more powerful base models are used.

In addition, we performed experiments with all possible sizes of the candidate label set. PML-*lc* and PML-*fp* are compared with other methods on each data set with respect to each criterion. Statistical significance is examined with pairwise t-test at 95% significance level. Table 3 summarizes the win/tie/loss counts of our methods versus the other methods.



(a) performance curve of PML-*lc* with C_2 changes.

(b) Performance curve of PML-*fp* with C_3 changes.

Figure 2: Results of PML-lc and PML-fp with varying value of trade-off parameters.

The results show that our methods outperform the others with varied sizes of candidate label set. PML-*lc* and PML-*fp* are still significantly better than other approaches in most cases. One exception is on the smallest dataset *CAL500*, where LIFT outperforms our methods over 3 criteria. This is probably because that there is too few training examples to recover the structure information. PML-*lc* and PML-*fp* are comparable on most datasets except for the *Corel5K*, on which PML-*lc* is outperformed by PML-*fp* along with ML-kNN and LIFT. One possible reason that PML-*lc* performs worse on image data is that the average number of instances associated with each label is relatively small on the image data, and thus the estimated label correlation based on co-occurrence may be less accurate.

At last, we study the influence of the trade-off parameters on the performances of PML-*lc* and PML-*fp*. While C_1 is fixed to 1 as default, we plot the performance curve in Fig. 2 as the parameters C_2 and C_3 change. Specifically, Fig. 2 (a) presents the performances of PML-*lc* when C_2 changes from 1 to 10 with step size of 1, and Fig. 2 (b) presents the performances of PML-*fp* when C_3 changes among $\{1, 10, 100\}$. As we can see, in general the performance is not sensitive to the parameters.

Table 3: Win/tie/loss counts (pairwise t-test at 95% significance level) on five multi-label learning criteria: Hamming loss, Ranking loss, One error, Coverage and Average precision of PML-lc and PML-fp against each comparing algorithm with different candidate set sizes .

Data	PML-*lc* versus					PML-*fp* versus				
	PML-*fp*	RankSVM	BSVM	ML-kNN	LIFT	PML-*lc*	RankSVM	BSVM	ML-kNN	LIFT
Hamming loss (the smaller, the better)										
Emotions	2/1/0	3/0/0	3/0/0	3/0/0	3/0/0	0/1/2	3/0/0	3/0/0	3/0/0	3/0/0
Yeast	2/2/4	8/0/0	8/0/0	8/0/0	7/1/0	4/2/2	8/0/0	8/0/0	8/0/0	7/1/0
CAL500	3/3/2	8/0/0	7/0/1	7/0/1	7/0/1	2/3/3	8/0/0	7/0/1	7/1/0	7/0/1
Genbase	2/3/3	8/0/0	8/0/0	8/0/0	8/0/0	3/3/2	8/0/0	8/0/0	8/0/0	8/0/0
Medical	5/1/2	8/0/0	8/0/0	8/0/0	8/0/0	2/1/5	8/0/0	8/0/0	8/0/0	8/0/0
Corel5K	0/0/8	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0
Delicious	4/3/1	8/0/0	7/0/1	8/0/0	7/0/1	1/3/4	8/0/0	7/0/1	8/0/0	7/0/1
Ranking loss (the smaller, the better)										
Emotions	3/0/0	3/0/0	3/0/0	3/0/0	3/0/0	0/0/3	3/0/0	3/0/0	3/0/0	3/0/0
Yeast	5/2/1	8/0/0	8/0/0	7/0/1	5/2/1	1/2/5	8/0/0	8/0/0	7/0/1	4/0/4
CAL500	0/7/1	8/0/0	8/0/0	8/0/0	1/1/6	1/7/0	8/0/0	8/0/0	8/0/0	1/1/6
Genbase	2/2/4	8/0/0	8/0/0	8/0/0	8/0/0	4/2/2	8/0/0	8/0/0	8/0/0	8/0/0
Medical	1/6/1	6/1/1	8/0/0	8/0/0	6/0/2	1/6/1	6/0/2	8/0/0	8/0/0	6/0/2
Corel5K	0/0/8	8/0/0	7/0/1	0/0/8	0/0/8	8/0/0	8/0/0	8/0/0	7/0/1	4/0/4
Delicious	4/3/1	8/0/0	8/0/0	8/0/0	6/0/2	1/3/4	8/0/0	8/0/0	8/0/0	6/1/1
One error (the smaller, the better)										
Emotions	2/0/1	3/0/0	2/0/1	3/0/0	3/0/0	1/0/2	3/0/0	2/0/1	3/0/0	3/0/0
Yeast	4/1/3	8/0/0	7/0/1	6/0/2	6/1/1	3/1/4	8/0/0	8/0/0	6/0/2	7/0/1
CAL500	4/1/3	8/0/0	7/0/1	5/0/3	3/0/5	3/1/4	8/0/0	7/0/1	5/0/3	2/0/6
Genbase	3/2/3	8/0/0	8/0/0	8/0/0	8/0/0	3/2/3	8/0/0	8/0/0	8/0/0	8/0/0
Medical	6/0/2	8/0/0	8/0/0	8/0/0	6/1/1	2/0/6	8/0/0	8/0/0	8/0/0	6/1/1
Corel5K	0/0/8	8/0/0	7/0/1	2/0/6	0/0/8	8/0/0	8/0/0	7/0/1	7/0/1	5/0/3
Delicious	4/1/3	8/0/0	8/0/0	8/0/0	6/0/2	3/1/4	8/0/0	8/0/0	8/0/0	6/0/2
Coverage (the smaller, the better)										
Emotions	3/0/0	3/0/0	3/0/0	3/0/0	3/0/0	0/0/3	3/0/0	3/0/0	3/0/0	3/0/0
Yeast	5/1/2	8/0/0	8/0/0	7/0/1	5/0/3	2/1/5	8/0/0	8/0/0	6/0/2	5/0/3
CAL500	0/7/1	8/0/0	8/0/0	8/0/0	6/0/2	1/7/0	8/0/0	8/0/0	8/0/0	6/0/2
Genbase	1/5/2	8/0/0	8/0/0	8/0/0	8/0/0	2/5/1	8/0/0	8/0/0	8/0/0	8/0/0
Medical	1/4/3	7/0/1	8/0/0	8/0/0	6/0/2	3/4/1	6/0/2	8/0/0	8/0/0	6/0/2
Corel5K	0/0/8	8/0/0	7/0/1	0/0/8	0/0/8	8/0/0	8/0/0	8/0/0	7/0/1	3/0/5
Delicious	4/3/1	7/0/1	8/0/0	8/0/0	8/0/8	1/3/4	7/0/1	8/0/0	8/0/0	8/0/0
Average precision (the smaller, the better)										
Emotions	3/0/0	3/0/0	2/0/1	3/0/0	3/0/0	0/0/3	3/0/0	2/0/1	3/0/0	3/0/0
Yeast	3/2/3	8/0/0	8/0/0	7/0/1	5/1/2	3/2/3	8/0/0	8/0/0	7/0/1	5/1/2
CAL500	2/4/2	8/0/0	7/1/0	6/0/2	0/2/6	2/4/2	8/0/0	8/0/0	6/0/2	0/2/6
Genbase	2/3/3	8/0/0	8/0/0	8/0/0	8/0/0	3/3/2	8/0/0	8/0/0	8/0/0	8/0/0
Medical	4/2/2	8/0/0	8/0/0	8/0/0	6/0/2	2/2/4	8/0/0	8/0/0	8/0/0	6/0/2
Corel5K	0/0/8	8/0/0	7/0/1	0/0/8	0/0/8	8/0/0	8/0/0	7/1/0	7/0/1	4/1/3
Delicious	3/5/0	8/0/0	8/0/0	8/0/0	6/0/2	0/5/3	8/0/0	8/0/0	8/0/0	6/0/2

Conclusion

In this paper, we propose a new learning framework named partial multi-label learning (PML), where each instance is associated with a set of candidate labels. A confidence value is defined for each candidate label to estimate how likely it is a ground-truth label. By minimizing the confidence weighted ranking loss and exploiting data structure information, the classification model along with the ground-truth confidence are optimized in a unified framework. Experiments are performed on various datasets, and results validate that the proposed approaches are superior to state-of-the-art multi-label approaches. In the future, we plan to improve the PML algorithms by incorporating domain knowledge and designing more advanced classification models.

References

Belkin, M.; Niyogi, P.; and Sindhwani, V. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research* 7:2399–2434.

Boutell, M. R.; Luo, J.; Shen, X.; and Brown, C. M. 2004. Learning multi-label scene classification. *Pattern Recognition* 37(9):1757–1771.

Bucak, S. S.; Jin, R.; and Jain, A. K. 2011. Multi-label learning with incomplete class assignments. In *Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition*, 2801–2808.

Cabral, R. S.; la Torre, F. D.; Costeira, J. P.; and Bernardino, A. 2011. Matrix completion for multi-label image classification. In *Advances in Neural Information Processing Systems 24*, 190–198.

Cour, T.; Sapp, B.; and Taskar, B. 2011. Learning from partial labels. *Journal of Machine Learning Research* 12:1501–1536.

Diplaris, S.; Tsoumakas, G.; Mitkas, P. A.; and Vlahavas, I. P. 2005. Protein classification with multiple algorithms. In *Proceedings of 10th Panhellenic Conference on Informatics*, 448–456.

Duygulu, P.; Barnard, K.; de Freitas, J. F. G.; and Forsyth, D. A. 2002. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of 7th European Conference on Computer Vision*, 97–112.

Elisseeff, A., and Weston, J. 2001. A kernel method for multi-labelled classification. In *International Conference on Neural Information Processing Systems*, 681–687.

Fürnkranz, J.; Hüllermeier, E.; Loza Mencía, E.; and Brinker, K. 2008. Multilabel classification via calibrated label ranking. *Machine Learning* 73(2):133–153.

Grandvalet, Y., and Bengio, Y. 2004. Learning from partial labels with minimum entropy. *Cirano Working Papers*.

Guo, Y., and Schuurmans, D. 2012. Semi-supervised multi-label classification. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 355–370.

Hüllermeier, E., and Beringer, J. 2006. Learning from ambiguously labeled examples. *Lecture Notes in Computer Science* 10(5):419–439.

Jin, R., and Ghahramani, Z. 2002. Learning with multiple labels. In *Advances in Neural Information Processing Systems*, 897–904.

Jing, L.; Yang, L.; Yu, J.; and Ng, M. K. 2015. Semi-supervised low-rank mapping learning for multi-label classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1483–1491.

Joachims, T. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of 10th European Conference on Machine Learning*, 137–142.

Kong, X.; Ng, M. K.; and Zhou, Z. 2013. Transductive multi-label learning via label set propagation. *IEEE Transactions on Knowledge and Data Engineering* 25(3):704–719.

Liu, L., and Dietterich, T. G. 2012. A conditional multinomial mixture model for superset label learning. In *Proceedings of 26th Annual Conference on Neural Information Processing Systems*, 557–565.

Liu, L., and Dietterich, T. G. 2014. Learnability of the superset label learning problem. In *Proceedings of the 31th International Conference on Machine Learning*, 1629–1637.

Pestian, J. P.; Brew, C.; Hovermale, D. J.; Johnson, N.; and Cohen, K. B. 2007. A shared task involving multi-label classification of clinical free text. In *The Workshop on Bionlp 2007: Biological, Translational, and Clinical Language Processing*, 97–104.

Read, J.; Pfahringer, B.; Holmes, G.; and Frank, E. 2011. Classifier chains for multi-label classification. *Machine Learning* 85(3):333–359.

Rubin, T. N.; Chambers, A.; Smyth, P.; and Steyvers, M. 2012. Statistical topic models for multi-label document classification. *Machine Learning* 88(1-2):157–208.

Sun, Y.; Zhang, Y.; and Zhou, Z. 2010. Multi-label learning with weak label. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*.

Szummer, M., and Jaakkola, T. S. 2001. Partially labeled classification with markov random walks. In *Proceedings of 14th Annual Conference on Neural Information Processing Systems*, 945–952.

Trohidis, K.; Tsoumakas, G.; Kalliris, G.; and Vlahavas, I. P. 2008. Multi-label classification of music into emotions. In *Proceedings of 9th International Conference on Music Information Retrieval*, 325–330.

Tsoumakas, G.; Katakis, I.; Vlahavas; and Ioannis. 2008. Effective and efficient multilabel classification in domains with large number of labels. *Mining Multidimensional Data*.

Tsoumakas, G.; Katakis, I.; and Vlahavas, I. 2011. Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering* 23(7):1079–1089.

Turnbull, D.; Barrington, L.; Torres, D. A.; and Lanckriet, G. R. G. 2008. Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio, Speech and Language Processing* 16(2):467–476.

Veit, A.; Alldrin, N.; Chechik, G.; Krasin, I.; Gupta, A.; and Belongie, S. 2017. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 839–847.

Wang, B., and Tsotsos, J. K. 2016. Dynamic label propagation for semi-supervised multi-class multi-label classification. *Pattern Recognition* 52:75–84.

Wang, J.; Yang, Y.; Mao, J.; Huang, Z.; Huang, C.; and Xu, W. 2016. Cnn-rnn: A unified framework for multi-label image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2285–2294.

Wu, F.; Wang, Z.; Zhang, Z.; Yang, Y.; Luo, J.; Zhu, W.; and Zhuang, Y. 2015. Weakly semi-supervised deep learning for multi-label image annotation. *IEEE Transactions on Big Data* 1(3):109–122.

Yu, F., and Zhang, M. 2017. Maximum margin partial label learning. *Machine Learning* 106(4):573–593.

Zhang, M., and Wu, L. 2015. Lift: Multi-label learning with label-specific features. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(1):107–120.

Zhang, M., and Yu, F. 2015. Solving the partial label learning problem: An instance-based approach. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 4048–4054.

Zhang, M., and Zhou, Z. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition* 40(7):2038–2048.

Zhang, M., and Zhou, Z. 2014. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26(8):1819–1837.

Zhao, F., and Guo, Y. 2015. Semi-supervised multi-label learning with incomplete labels. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 4062–4068.