Noname manuscript No.
(will be inserted by the editor)

Learning Safe Multi-Label Prediction for Weakly Labeled Data

Tong Wei · Lan-Zhe Guo
Yu-Feng Li · Wei Gao

Abstract In this paper we study multi-label learning with weakly labeled data, i.e., labels of training examples are incomplete. This includes, e.g., (i) semi-supervised multi-label learning where completely labeled examples are partially known; (ii) weak label learning where relevant labels of examples are partially known; iii) extended weak label learning where relevant and irrelevant labels of examples are partially known. Weakly labeled data commonly occur in real applications, e.g., image classification, document categorization. Previous studies often expect that learning methods with the use of weakly labeled data improve learning performance, as more data are employed. This, however, is not always the cases in reality. Using more weakly labeled data may sometimes degenerate learning performance. It is desirable to learn safe multi-label prediction that will not hurt performance when weakly labeled data is used. In this work we optimize multi-label evaluation metrics (F_1 score and Top-k precision) given that ground-truth label assignments are realized by a convex combination of basic multi-label learners. To cope with infinite number of possible ground-truth label assignments, cutting-plane strategy is adopted to iteratively generate the most helpful label assignments. The whole optimization is cast as a series of simple linear programs in an efficient manner. Extensive experiments on three weakly labeled learning tasks, namely, i) semi-supervised multi-label learning; ii) weak-label learning and iii) extended weak-label learning, show that our proposal clearly improves the safeness in comparison to many state-of-the-art methods.

Keywords multi-label learning · weakly labeled data · safe · evaluation metric

1 Introduction

In many real applications, learning objects are associated with multiple labels. For example, in image classification (Carneiro et al, 2007), one image can be associated

Tong Wei · Lan-Zhe Guo · Yu-Feng Li · Wei Gao
National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210023, China
E-mail: {weit, guolz, liyf, gaow}@lamda.nju.edu.cn

Label Assignments

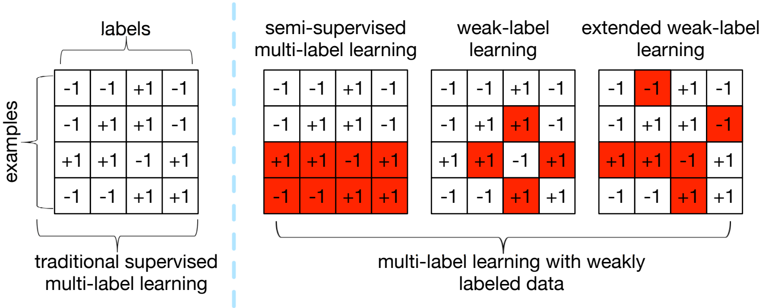

Fig. 1: Illustration for weakly labeled data. +1 and −1 represent relevant and irrelevant labels. Red cells represent missing labels. In this paper three kinds of weakly labeled data are considered, namely, semi-supervised multi-label, weak-label and extended weak-label learning.

with many concept labels such as 'sky', 'cloud', 'flower', etc; in document categorization (Srivastava and Zane-Ulman, 2005), one document could be related to multiple topics such as 'sport', 'football', 'lottery', etc. Multi-label learning (Zhang and Zhou, 2014) is now one hot research area in dealing with learning examples related to multiple labels. Due to its wide suitability, multi-label learning techniques have been adopted for many applications, and a number of multi-label learning algorithms have been developed (Tsoumakas et al, 2009; Zhang and Zhou, 2014).

Although multi-label representation provides a better characterization than singe-label one, in real applications the acquisition of labels suffers from various difficulties, and *weakly labeled data*, i.e., labels of training examples are incomplete, commonly occurs. For example, human labelers may only give labels for a few training examples. In this case, completely labeled examples are partially available and many training examples are unlabeled, which is realized as *semi-supervised multi-label learning problem* (Liu et al, 2006; Kong et al, 2013); human labelers may only give partial relevant labels for training examples. In this case, relevant labels of training examples are partially known and many relevant labels are missing, which is realized as *weak label learning problem* (Sun et al, 2010); human labelers may only give partial relevant and irrelevant labels for training examples. In this case, relevant and irrelevant labels of training examples are partially known, we refer it to *extended weak label learning problem*. Figure 1 illustrates three weakly label assignments for multi-label training data. Over the past decade, multi-label learning with weakly labeled data attracts increasing attentions and a large number of algorithms have been presented (Liu et al, 2006; Sun et al, 2010; Chen et al, 2008; Kong et al, 2013; Wang et al, 2013; Yu et al, 2014; Zhao and Guo, 2015).

In previous studies, it is often expected that multi-label learning methods with the use of weakly labeled data are better than counterpart approaches, i.e., supervised multi-label learning methods using only labeled data, as more data are employed. This, however, is not always the cases in reality. As reported in quite many empirical studies (Chen et al, 2008; Wang et al, 2013; Zhao and Guo, 2015), using more weakly labeled data may sometimes degenerate learning performance.

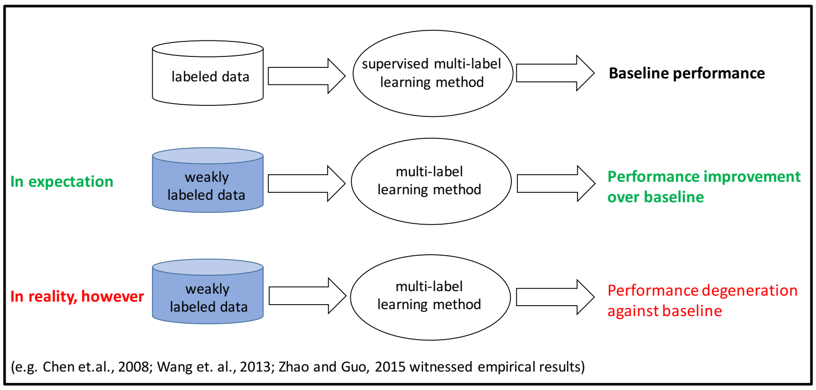

Fig. 2: Motivation of the paper. In many cases, traditional multi-label learning algorithms using weakly labeled data may degenerate learning performance, which is not in line with our expectation.

This hinder multi-label learning to play roles in more applications. It is important to have *safe* multi-label learning methods which could always improve learning performance with the use of weakly labeled data, and in the worst case scenario, they will not degenerate learning performance. Figure 2 illustrates the basic motivation of our paper.

To overcome this issue, in this work we propose SafeML (SAFE Multi-Label prediction for weakly labeled data). It directly optimizes multi-label evaluation metrics (F_1 score and Top-k precision) via formulating a distribution of ground-truth label assignments. Specifically, we assume that ground-truth label assignments are realized by a convex combination of multiple basic multi-label learners, inspired by Li et al (2017). To cope with the infinite number of possible ground-truth label assignments in optimization, cutting-plane strategy is adopted, which iteratively generates the most helpful label assignments. The optimization is then cast as a series of simple linear programs in an efficient manner. Extensive experiments on three weakly labeled tasks, namely, i) semi-supervised multi-label learning; ii) weak-label learning and iii) extended weak-label learning, show that our proposal clearly improves the safeness with the use of weakly labeled data, in comparison to many state-of-the-art methods.

The rest of the paper is organized as follows. We first introduce some related works and then present the proposed method. This is then followed by extensive experimental results, and finally we give conclusive remarks.

2 Related Work

This work is related to two branches of studies. The first one is multi-label learning approaches for weakly labeled data. As for semi-supervised multi-label learning problem, one early work is proposed by (Liu et al, 2006). They assumed that the similarity in the label space is closely related to that in the feature space, and thus employed the similarity in feature space to guide the learning of missing label assignments, which leads to a constrained nonnegative matrix factorization (CNMF) optimization. Later, Chen et al (2008), inspired by the idea of label propagation, inferred the label assignments for unlabeled data via two graphs on instance-level and label-level respectively. Similarly, Wang et al (2013) proposed to propagate from labeled data to unlabeled data via a dynamic graph. Zhao and Guo (2015) aimed to improve multi-label prediction performance by integrating label correlation and multi-label prediction in a mutually beneficial manner.

As for weak label learning problem, there are some approaches. One early work is proposed by (Sun et al, 2010). They employed label propagation idea to learn missing label assignments and controlled the quality of learned relevant labels through sparsity regularizer. Bucak et al (2011) formulated the problem via standard statistical learning framework and introduced group lasso loss function that enforced the learned relevant labels to be sparse. Chen et al (2013) first attempted to reconstruct the (unknown) complete label set from a few label assignments, and then learned a mapping from the input features to the reconstructed label set. Yu et al (2014) first initialized the label assignments via training model on the labels observed and then performed label completion based on visual similarity and label co-occurrence of learning objects (Wu et al, 2013; Zhu et al, 2010).

As for extended weak label learning problem, to our best knowledge, it has not been studied yet and this paper is the first work on this new setting. Generally, previous multi-label learning methods on weakly labeled data typically work on improving the performance based on some assumptions/conditions, no study has been proposed on using weakly labeled data safely.

The second branch of studies is safe machine learning techniques for weakly labeled data, which are now generally focused on semi-supervised learning scenario. S4VM (Safe Semi-Supervised SVM) (Li and Zhou, 2015) is one early work to build safe semi-supervised SVMs. They optimized the worst-case performance gain given a set of candidate low-density separators, showing that the proposed S4VM is provably safe given that low-density assumption (Chapelle et al, 2009) holds. UMVP (Li et al, 2016) concerns to build a generic safe SSC framework for variants of performance measures, e.g., AUC, F_1 score, Top-k precision. Krijthe and Loog (2015) developed a robust semi-supervised classifier, which learns a projection of a supervised least square classifier from all possible semi-supervised least square classifiers. Most recently, Li et al (2017) explicitly considers to maximize the performance gain and learns a safe prediction from multiple semi-supervised regressors, which is not worse than a direct supervised learner with only labeled data. However, all these works focus on binary classification or regression cases, which are not sufficient to cope with multi-label learning problems (will be verified in our empirical studies), as they fail to take rich label correlations into account.

3 Proposed SafeML Method

In this section, we first present some backgrounds of multi-label learning, including problem notations and popular evaluation metrics for multi-label learning. We then present problem formulation for safe multi-label learning with weakly labeled data, followed by its optimization and analysis.

Table 1: Summary of Notation

Notation	Meaning
N	number of instances
L	number of labels
d	number of features
$\mathbf{x} \in \mathbb{R}^d$	instance feature vector
$\mathbf{X} = [\mathbf{x}_1; \ldots; \mathbf{x}_n] \in \mathbb{R}^{n \times d}$	instance feature matrix representation
$\mathbf{y} \in \{-1, 1\}^L$	label vector of multi-label data
$\mathbf{Y} \in \{-1, 1\}^{N \times L}$	label matrix of multi-label data
$\bar{\mathbf{Y}} \in \{-1, 0, 1\}^{N \times L}$	label matrix of weakly labeled data, where '0' means missing label
b	number of base learners
$\{\mathbf{P}_i\}_{i=1}^b \in \{-1, 1\}^{N \times L}$	pseudo label matrices generated by base learners
$\mathbf{v} = [v_1, v_2, \ldots, v_b]$	weight vector of base learners
$\hat{\mathbf{Y}} \in \{-1, 1\}^{N \times L}$	our predictive label matrix

3.1 Background

Notation In traditional supervised multi-label learning, the training data set is represented as $\{(\mathbf{x}_1, \mathbf{y}_1), \cdots, (\mathbf{x}_N, \mathbf{y}_N)\}$, where $\mathbf{x}_i \in \mathbb{R}^d$ is the feature vector of the i-th instance, and $\mathbf{y}_i \in \{-1, 1\}^L$ is the corresponding label vector. N and L are the number of instances and labels, respectively. The feature matrix is denoted as $\mathbf{X} = [\mathbf{x}_1; \cdots; \mathbf{x}_N] \in \mathbb{R}^{N \times d}$ and the label matrix $\mathbf{Y} = [\mathbf{y}_1; \cdots; \mathbf{y}_N] \in \{-1, 1\}^{N \times L}$. If instance $\mathbf{x}_i$ is associated to the j-th label, then $\mathbf{Y}_{ij} = 1$; otherwise, $\mathbf{Y}_{ij} = -1$. Given $\mathbf{X}$ and $\mathbf{Y}$, the goal of multi-label learning is to learn a hypothesis $f : \mathbb{R}^d \to \{-1, 1\}^L$ that accurately predicts the label vector for a given instance.

However, when weakly labeled data occurs, the label assignments in $\mathbf{Y}$ is not complete and some parts of the label assignments in $\mathbf{Y}$ are missing. In this case, what we have is an incomplete label matrix $\bar{\mathbf{Y}} \in \{-1, 0, +1\}^{N \times L}$ where '0' indicates the cases that the corresponding label assignments are missing.

As previously mentioned, our goal in the paper is to derive safe multi-label prediction for weakly labeled data. Specifically, given $\mathbf{Y}_0$ be the predictive label matrix based on direct supervised multi-label learning algorithms, e.g., binary relevance (Read et al, 2011), we would like to learn a safe multi-label prediction $\hat{\mathbf{Y}}$ from $\{\mathbf{X}, \bar{\mathbf{Y}}\}$ such that $\hat{\mathbf{Y}}$ is often better than $\mathbf{Y}_0$ w.r.t. *multi-label evaluation metrics*. In the following, we introduce two popular multi-label evaluation metrics.

Multi-label Evaluation Metrics The first one is F_1 score. F_1 score is a widely used evaluation for multi-label learning, which trades off precision and recall (Zhang and Zhou, 2014). It takes both precision and recall into consideration with equal importance. Traditional F_1 score is computed for binary classification problem. When F_1 meets multi-label learning, it can be obtained in the following two modes.

– $MacroF_1$:

$$MacroF_1 = \frac{1}{L} \sum_{j=1}^{L} F_1(TP_j, FP_j, TN_j, FN_j) \tag{1}$$

– $MicroF_1$:

$$MicroF_1 = F_1(\sum_{j=1}^{L} TP_j, \sum_{j=1}^{L} FP_j, \sum_{j=1}^{L} TN_j, \sum_{j=1}^{L} FN_j) \tag{2}$$

where TP_j, FP_j, TN_j, FN_j represent the number of *true positive*, *false positive*, *true negative*, and *false negative* test examples with respect to label assignments of the j-th label, and

$$F_1(TP, FP, TN, FN) = \frac{2TP}{2TP + FN + FP}.$$

As can be seen, $MacroF_1$ characterizes the average of F_1 scores over all the labels, while $MicroF_1$ characterizes the F_1 score w.r.t. the sum of TP, FP, TN, FN over all the labels. They both characterize the tradeoff between precision and recall, from different aspects.

The second one is Top-k precision. Top-k precision is also popularly used in multi-label learning applications (Zhang and Zhou, 2014), especially for those in information retrieval or search areas. In Top-k precision, only a few top predictions of an instance will be considered. For each instance $\mathbf{x}_i$, the Top-k precision is defined for a predicted score vector $\hat{\mathbf{y}}_i \in \mathcal{R}^L$ and ground truth label vector $\mathbf{y}_i \in \{-1, 1\}^L$ as

$$Pre@k(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \frac{1}{k} \sum_{l \in \mathrm{rank}_k(\hat{\mathbf{y}}_i)} (\mathbf{y}_{i,l} + 1)/2 \tag{3}$$

where $\mathrm{rank}_k(\hat{\mathbf{y}}_i)$ returns the indices of k largest value in $\hat{\mathbf{y}}_i$ ranked in descending order. Therefore, the Top-k precision for a set of training instances is derived as

$$Pre@k(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{k} \sum_{l \in \mathrm{rank}_k(\hat{\mathbf{y}}_i)} (\mathbf{y}_{i,l} + 1)/2 \tag{4}$$

3.2 Problem Formulation

We now describe our prediction problem, and formulate it as a zero-sum game between two players: a predictor and an adversary which is similar to the method mentioned in (Balsubramani and Freund, 2015). In this game, the predictor is the first player, who plays $\hat{\mathbf{Y}}$, a label matrix for training instances $\{\mathbf{x}_i\}_{i=1}^{N}$. The adversary then plays $\mathbf{Y}$, setting the ground-truth label matrix $\mathbf{Y} \in \{-1, 1\}^{N \times L}$ under the constraints that $\mathbf{Y}$ could be reconstructed by a set of base learners. The predictor's goal is to maximize (and the adversary is to minimize) the *expected learning performance on the test data*. The SafeML method formulates this as the following maximin optimization framework:

$$\max_{\hat{\mathbf{Y}}} \min_{\mathbf{Y} \in \Omega} \; perf(\hat{\mathbf{Y}}, \mathbf{Y}) \tag{5}$$

$$\text{s.t.} \;\; \Omega = \left\{ \mathbf{Y} \middle| \mathbf{Y} = \sum_{i=1}^{b} v_i \mathbf{P}_i \right\}$$

where *perf* represents the target performance measure (e.g., F_1 score, Top-k precision) and $\{\mathbf{P}_1, \ldots, \mathbf{P}_b\}$ are pseudo label matrices generated by base learners, $\mathbf{v} = [v_1, \ldots, v_b]$ captures the relative importance of the b base learners. Without loss of generality, we assume that $\mathbf{v}$ is in the simplex $\mathcal{M} = \{\mathbf{v} \big| \sum_{i=1}^{b} v_i = 1, v_i \geq 0\}$. Eq.(5) leads to robust and accurate multi-label predictions, as it maximizes the learning performance w.r.t. ground-truth label assignment, meanwhile considers the risk that ground-truth label matrix is uncertain and from a distribution. In the sequel we present the optimization of Eq.(5) w.r.t. multi-label evaluation metrics, i.e., F_1 score and Top-k precision.

3.3 Optimize Eq.(5) with F_1 Score

When F_1 score is considered, given $\mathbf{Y}$ and $\hat{\mathbf{Y}}$, we have

$$\sum_{j=1}^{L} TP_j = \sum_{j=1}^{L} \sum_{i=1}^{N} \mathbb{I}(\mathbf{Y}_{i,j} = 1 \wedge \hat{\mathbf{Y}}_{i,j} = 1) \tag{6}$$

$$\sum_{j=1}^{L} FP_j = \sum_{j=1}^{L} \sum_{i=1}^{N} \mathbb{I}(\mathbf{Y}_{i,j} \neq 1 \wedge \hat{\mathbf{Y}}_{i,j} = 1) \tag{7}$$

$$\sum_{j=1}^{L} TN_j = \sum_{j=1}^{L} \sum_{i=1}^{N} \mathbb{I}(\mathbf{Y}_{i,j} \neq 1 \wedge \hat{\mathbf{Y}}_{i,j} \neq 1) \tag{8}$$

$$\sum_{j=1}^{L} FN_j = \sum_{j=1}^{L} \sum_{i=1}^{N} \mathbb{I}(\mathbf{Y}_{i,j} = 1 \wedge \hat{\mathbf{Y}}_{i,j} \neq 1) \tag{9}$$

Eq.(6) shows that $\sum_{j=1}^{L} TP_j$ equals to $\mathrm{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2})\right)$. From Eq. (6, 7, 9), we notice that $2TP + FN + FP$ is equal to the number of $+1$ in $\mathbf{Y}$ and $\hat{\mathbf{Y}}$. Thus Eq.(5) can be rewritten as following:

$$\max_{\hat{\mathbf{Y}}} \min_{\mathbf{Y} \in \Omega} \quad \frac{\mathrm{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2})\right)}{\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) + \sum_{i,j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j} = 1)} \tag{10}$$

where $\mathbb{I}(\cdot)$ is the indicator function that returns 1 when the argument holds and 0 otherwise. $\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1)$ is the number of $+1$ in $\mathbf{Y}$ and $\sum_{i,j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j} = 1)$ is the number of $+1$ in $\hat{\mathbf{Y}}$.

To simplify this problem, we consider that the ratio of relevant labels for ground-truth label assignments are approximately closely related to a constant, i.e., $\left|\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) - \gamma_0\right| \leq \epsilon$ and we set γ_0 according to the average number of $+1$ on training data. Therefore, the denominator of the object function in Eq.

(10) can be approximated as a constant and thus Eq. (10) can be written as

$$\begin{aligned}
&\max_{\hat{\mathbf{Y}}} \min_{\mathbf{Y}\in\Omega} \ \operatorname{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2}) \right) \\
&\qquad \text{s.t.} \ \Big| \sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j}=1) - \gamma_0 \Big| \le \epsilon, i=1\cdots N, \ j=1\cdots L \\
&\qquad\qquad \Big| \sum_{i,j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j}=1) - \gamma_0 \Big| \le \epsilon, i=1\cdots N, \ j=1\cdots L
\end{aligned} \tag{11}$$

Consequently, Eq. (11) can be rewritten as the following version:

$$\begin{aligned}
&\max_{\hat{\mathbf{Y}},\theta} \ \theta \\
&\text{s.t.} \ \theta \le \operatorname{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2}) \right), \forall\, \mathbf{Y}\in\Omega \\
&\quad \Big| \sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j}=1) - \gamma_0 \Big| \le \epsilon, \ i=1\cdots N, \ j=1\cdots L \\
&\quad \Big| \sum_{i,j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j}=1) - \gamma_0 \Big| \le \epsilon, \ i=1\cdots N, \ j=1\cdots L
\end{aligned} \tag{12}$$

Note that there can be an exponential number of constraints in Eq. (12), and so a direct optimization is computationally intractable. Hence we employ the cutting-plane algorithm to solve this problem. Instead of using all the constraints in Ω to construct the optimization problem in Eq. (12), we only use an active set of constraints, which contains a limited number of constraints in Ω. Cutting-plane algorithm iteratively adds a cutting plane to shrink the feasible region. Specifically, let $\mathcal{C}$ be an active constraint set. With a fixed $\hat{\mathbf{Y}}$, the cutting-plane algorithm needs to find the most violated constraint for the current $\hat{\mathbf{Y}}$ by solving

$$\mathbf{Y}_{\text{new}} = \arg\min_{\mathbf{Y}\in\Omega} \ \operatorname{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2}) \right), \ \text{s.t.} \ \Big| \sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j}=1) - \gamma_0 \Big| \le \epsilon \tag{13}$$

and add $\mathbf{Y}_{\text{new}}$ to the active constraint set $\mathcal{C}$. To simplify this equation, we construct vector $\mathbf{z}_{1\times b}$, where $\mathbf{z}_i = \operatorname{tr}(\mathbf{P}_i^\top \frac{\hat{\mathbf{Y}}+1}{2})$ and then $\operatorname{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2}) \right)$ equals to $\mathbf{v}\mathbf{z}^\top$. Similarly, construct matrix $\bar{\mathbf{Z}}_{b\times N}$, where $\bar{\mathbf{Z}}_i = (\mathbf{P}_i \mathbf{1}_{L\times 1})^\top$ and $\mathbf{1}_{L\times 1}$ is a column vector with all L values set to be 1, then $\mathbf{v}\bar{\mathbf{Z}}\mathbf{1}_{N\times 1}$ equals to the number of $+1$ in $\mathbf{Y}$. Hence, the problem can be rewritten as

$$\begin{aligned}
&\min_{\mathbf{v}\in\mathcal{M}} \ \mathbf{v}\mathbf{z}^\top \\
&\text{s.t.} \ \Big| \mathbf{v}\bar{\mathbf{Z}}\mathbf{1}_{N\times 1} - \gamma_0 \Big| \le \epsilon
\end{aligned} \tag{14}$$

Eq. (14) is a simple linear programming that is readily to solve globally and efficiently. Given active constraint set $\mathcal{C}$, which is a subset of Ω, we can replace the

Algorithm 1 Cutting-plane algorithm for Eq. (12)

Input: label matrices $\{\mathbf{P}_i\}_{i=1}^b$ and parameter γ_0
Output: predictive label matrix $\hat{\mathbf{Y}}$

```
Initialize Y_0 = (1/b) Σ_{i=1}^b P_i, working set C = {Y_0} and t = 1
while not converge do
    Obtain Ŷ_t by solving Eq. (15)
    Obtain v by solving Eq. (14)
    Obtain Y_new according to Y_new = Σ_{i=1}^b v_i P_i
    Set C = C ⋃ Y_new; t = t + 1
end while
return Ŷ = Ŷ_t
```

Ω in Eq. (12) with $\mathcal{C}$ and obtain

$$\begin{aligned} &\max_{\hat{\mathbf{Y}},\theta} \ \theta \\ &\text{s.t. } \theta \le \operatorname{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top (\frac{\mathbf{Y}+1}{2}) \right), \ \forall \mathbf{Y} \in \mathcal{C} \\ &\left| \sum_{i,j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j} = 1) - \gamma_0 \right| \le \epsilon, \ i = 1 \cdots N, \ j = 1 \cdots L \end{aligned} \tag{15}$$

Both the objective function and constraints in Eq. (15) are linear in $\mathbf{Y}$ and θ. Hence, we can solve the Eq. (15) with a linear programming efficiently.

Algorithm 1 summarizes the pseudo code of the cutting plane algorithm. In most cases of our experiment, the algorithm converged within a maximum number of iterations (100 iterations in our experiments). The update of $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ (i.e., Eq. (14) and Eq.(15)) are solved by a convex and simple linear programming problem, Eq. (12) is then addressed efficiently.

3.4 Optimize Eq.(5) with Top-k Precision

According to Eq. (4), given $\mathbf{Y}$ and $\hat{\mathbf{Y}}$, Top-k precision can be formulated as

$$Pre@k(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{Nk} \sum_{i=1}^{N} \sum_{j=1}^{L} \mathbb{I}(\mathbf{Y}_{ij} = 1) \mathbb{I}(\pi_j^{\hat{\mathbf{Y}}_i} > L - k) \tag{16}$$

where $\pi^{\hat{\mathbf{Y}}_i}$ is the ranking vector of $\hat{\mathbf{Y}}_i$, where $\pi_p^{\hat{\mathbf{Y}}_i} > \pi_q^{\hat{\mathbf{Y}}_i}$ if $\hat{\mathbf{Y}}_{ip} > \hat{\mathbf{Y}}_{iq}$(with ties broken arbitrarily). Similarly, considering that the ratio of relevant labels for ground-truth label assignments are approximately closely related to a constant, i.e., $\left| \sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) - \gamma_0 \right| \le \epsilon$ and each instance is constrained to be associated with exactly k positive labels, then the optimization objective becomes

$$\begin{aligned} \max_{\hat{\mathbf{Y}}} \min_{\mathbf{Y}\in\Omega} \quad & Pre@k(\mathbf{Y}, \hat{\mathbf{Y}}) \\ \text{s.t.} \quad & \Big|\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) - \gamma_0\Big| \le \epsilon,\ i = 1\cdots N,\ j = 1 \cdots L \\ & \sum_{j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j} = 1) = k,\ i = 1 \cdots N \end{aligned} \tag{17}$$

Eq. (17) can be rewritten as

$$\begin{aligned} \max_{\hat{\mathbf{Y}},\theta} \quad & \theta \\ \text{s.t.} \quad & \theta \le Pre@k(\mathbf{Y}, \hat{\mathbf{Y}}), \forall\ \mathbf{Y} \in \Omega \\ & \Big|\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) - \gamma_0\Big| \le \epsilon,\ i = 1\cdots N,\ j = 1 \cdots L \\ & \sum_{j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j} = 1) = k,\ i = 1 \cdots N \end{aligned} \tag{18}$$

Instead of using all the constraints in Ω to construct the optimization problem in Eq.(18), we only use an active set of constraints, which contains a limited number of constraints in Ω. The proposed cutting-plane algorithm iteratively adds a cutting plane to shrink the feasible region. Specifically, let $\mathcal{C}$ be an active constraint set. With a fixed $\hat{\mathbf{Y}}$, the cutting-plane algorithm needs to find the most violated constraint by solving

$$\mathbf{Y}_{\text{new}} = \arg\min_{\mathbf{Y}\in\Omega} \quad Pre@k(\mathbf{Y}, \hat{\mathbf{Y}}), \quad \text{s.t.} \quad \Big|\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) - \gamma_0\Big| \le \epsilon \tag{19}$$

It can be proved that the value of $Pre@k(\mathbf{Y}, \hat{\mathbf{Y}})$ equals to $\text{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top(\frac{\mathbf{Y}+1}{2})\right)$ (Li et al, 2016). Hence, Eq. (19) can be transformed into

$$\mathbf{Y}_{\text{new}} = \arg\min_{\mathbf{Y}\in\Omega} \quad \text{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top(\frac{\mathbf{Y}+1}{2})\right), \quad \text{s.t.} \quad \Big|\sum_{i,j} \mathbb{I}(\mathbf{Y}_{i,j} = 1) - \gamma_0\Big| \le \epsilon \tag{20}$$

Similar to the case in F_1 score, the optimization problem can be rewritten as following:

$$\begin{aligned} \min_{\mathbf{v}\in\mathcal{M}} \quad & \mathbf{v}\mathbf{z}^\top \\ \text{s.t.} \quad & \Big|\mathbf{v}\bar{\mathbf{Z}}\mathbf{1}_{N\times 1} - \gamma_0\Big| \le \epsilon \end{aligned} \tag{21}$$

Eq. (21) is a simple linear programming that is readily to solve globally and efficiently. Given an active constraints set $\mathcal{C}$, which is a subset of Ω, we can replace the Ω in Eq. (18) with $\mathcal{C}$ and obtain

$$\begin{aligned} \max_{\hat{\mathbf{Y}},\theta} \quad & \theta \\ \text{s.t.} \quad & \theta \le \text{tr}\left((\frac{\hat{\mathbf{Y}}+1}{2})^\top(\frac{\mathbf{Y}+1}{2})\right), \quad \forall\ \mathbf{Y} \in \mathcal{C} \\ & \Big|\sum_{i,j} \mathbb{I}(\hat{\mathbf{Y}}_{i,j} = 1) - \gamma_0\Big| \le \epsilon,\ i = 1\cdots N,\ j = 1 \cdots L \end{aligned} \tag{22}$$

Algorithm 2 Cutting-plane algorithm for Eq. (18)

Input: label matrices $\{\mathbf{P}_i\}_{i=1}^{b}$ and parameter γ_0
Output: predictive label matrix $\hat{\mathbf{Y}}$

```
Initialize Y_0 = 1/b ∑_{i=1}^{b} P_i, working set C = {Y_0} and t = 1
while not converge do
    Obtain Ŷ_t by solving Eq. (22)
    Obtain v by solving Eq. (21)
    Obtain Y_new according to Y_new = ∑_{i=1}^{b} v_i P_i
    Set C = C ⋃ Y_new; t = t + 1
end while
return Ŷ = Ŷ_t
```

Both the objective function and constraints in Eq. (22) are linear in $\mathbf{Y}$ and θ. Hence, we can solve the Eq. (22) with a linear programming efficiently. Algorithm 2 summarizes the pseudo code of the cutting plane algorithm. The algorithm converged within a maximum number of iterations (100 iterations in our experiments). The update of $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ (i.e., Eq. (21) and Eq. (22)) is solved with convex and simple linear programming problems, Eq. (18) is addressed efficiently.

3.5 How the proposal works

Except for efficient algorithms, it is also important to study how the proposal works. In the following, we show that the performance of our proposal is closely related to the correlation of base learners.

Theorem 1 *Let* $\boldsymbol{Y}^{GT}$ *be the ground-truth label matrix and* $\hat{\mathbf{Y}}^{*}$ *be the prediction of* SafeML*, i.e., the optimal solution to Eq. (5). The performance of our proposal* $\mathrm{perf}(\hat{\mathbf{Y}}^{*}, \boldsymbol{Y}^{GT})$ *w.r.t.* F_1 *score and Top-k precision is lower bounded by* $\max\limits_{i=1,\dots,b} \min\limits_{j=1,\dots,b} \mathrm{perf}(\boldsymbol{P}_i, \boldsymbol{P}_j)$ *as long as* $\boldsymbol{Y}^{GT} \in \Omega$.

Proof Let $f(\hat{\mathbf{Y}}) = \min\limits_{\mathbf{Y}\in\Omega} \mathit{perf}(\hat{\mathbf{Y}}, \mathbf{Y})$, since $\hat{\mathbf{Y}}^{*}$ is the optimal solution to Eq. (5), the following inequality holds:

$$f(\hat{\mathbf{Y}}^{*}) \geq f(\mathbf{P}_i), \ \ i = 1, \dots, b \tag{23}$$

which implies that

$$f(\hat{\mathbf{Y}}^{*}) \geq \max_{1 \leq i \leq b} f(\mathbf{P}_i) \tag{24}$$

According to the definition of function f, for any i $(1 \leq i \leq b)$ we have

$$f(\mathbf{P}_i) = \min_{\mathbf{Y}\in\Omega} \mathit{perf}(\mathbf{P}_i, \mathbf{Y}) \tag{25}$$
$$\text{s.t. } \ \Omega = \left\{ \mathbf{Y} \middle| \mathbf{Y} = \sum_{i=1}^{b} v_i \mathbf{P}_i \right\}$$

and since the Top-k Precision, F_1 score are used as performance measures, Eq. 25 can be reduced to

$$f(\mathbf{P}_i) = \sum_{j=1}^{b} v_j \; \mathit{perf}(\mathbf{P}_i, \mathbf{P}_j) \tag{26}$$

$$\text{s.t.} \quad \sum_{i=j}^{b} v_j = 1, v_i \geq 0$$

which naturally becomes,

$$f(\mathbf{P}_i) = \min_{1 \leq j \leq b} \; \mathit{perf}(\mathbf{P}_i, \mathbf{P}_j) \tag{27}$$

$$f(\hat{\mathbf{Y}}^*) \geq \max_{1 \leq i \leq b} \min_{1 \leq j \leq b} \; \mathit{perf}(\mathbf{P}_i, \mathbf{P}_j) \tag{28}$$

because $f(\hat{\mathbf{Y}}^*) = \min_{\mathbf{Y} \in \Omega} \; \mathit{perf}(\hat{\mathbf{Y}}^*, \mathbf{Y})$ and $\mathbf{Y}^{GT} \in \Omega$, we have

$$\mathit{perf}(\mathbf{Y}^*, \mathbf{Y}^{GT}) \geq f(\hat{\mathbf{Y}}^*) \tag{29}$$

Integrating inequations (28)-(29), we then derive

$$\mathit{perf}(\mathbf{Y}^*, \mathbf{Y}^{GT}) \geq \max_{1 \leq i \leq b} \min_{1 \leq j \leq b} \; \mathit{perf}(\mathbf{P}_i, \mathbf{P}_j) \tag{30}$$

According to Theorem 1, the performance of SafeML is related to the maximin correlation of base learners. In practice, as shown in Table 2, it is often much larger than direct supervised multi-label learning with only labeled data. That is why our proposal performs effectively.

Table 2: Comparison between the lower bound performance in Theorem 1 and direct supervised multi-label learning.

Data set	the lower bound in Theorem 1		direct binary relevance SVM	
	macro F_1	micro F_1	macro F_1	micro F_1
emotions	0.774	0.855	0.539	0.592
enron	0.194	0.916	0.076	0.477
image	0.378	0.783	0.105	0.130
scene	0.739	0.866	0.422	0.458
yeast	0.501	0.908	0.318	0.620

4 Experiments

To evaluate the effectiveness of our proposal, we conduct experimental comparisons with state-of-the-art methods on a number of benchmark multi-label data sets. We report our experimental setting and results in this section.

Table 3: Benchmark multi-label data sets

Data set	# inst	# feat	# label	# card-label
emotions	593	72	6	1.869
enron	1,702	1,001	53	3.378
image	2,000	294	5	1.236
scene	2,407	294	6	1.074
yeast	2,417	103	14	4.237
arts	5,000	462	26	1.636
bibtex	7,395	1,836	159	2.400
tmc2007	28,596	981	22	2.158
delicious	13,903	500	983	19.030

4.1 Setup

Data sets We evaluate the proposed method on nine multi-label data sets: *emotions*, *enron*, *image*, *scene*, *yeast*, *arts*, *bibtex*, *tmc*2007 and *delicious*. A summary of the statistics of data sets is shown in Table 3. #inst is the number of instance in the data set; #feat is the number of features; #label is the number of labels; #card-label is the average number of labels per example. The sample size ranges from 593 to more than 28,000. The feature dimensionality ranges from 72 to more than 1,800. The label size ranges from 5 to 983. These data sets cover a broad range of properties.

Compared Methods We compare the performance of the proposed algorithm with following methods.

– BR (**B**inary **R**elevance) (Tsoumakas et al, 2009): the baseline method. A binary SVM classifier is trained on only labeled instances for each label.
– S4VM (**S**afe **S**emi-**S**upervised **SVM**) (Li and Zhou, 2015): A binary S4VM classifier is trained on both labeled and unlabeled instances for each label.
– ML-kNN (Zhang and Zhou, 2007) is a kNN style multi-label classification algorithm which often outperforms other existing multi-label algorithms.
– ECC (**E**nsemble **C**lassifier **C**hain): state-of-the-art supervised ensemble multi-label method proposed in (Read et al, 2011).
– CNMF (semi-supervised multi-label learning by **C**onstrained **N**on-negative **M**atrix **F**actorization) (Liu et al, 2006) is a semi-supervised multi-label classification algorithm via constrained non-negative matrix factorization.
– LEML (**L**ow rank **E**mpirical risk minimization for **M**ulti-Label **L**earning) (Yu et al, 2014): recent state-of-the-art multi-label method for weakly labeled data by formulating the problem as an empirical risk minimization.
– TRAM (**TRA**sductive **M**ultilabel Classification) (Kong et al, 2013) is a transductive multi-label classification algorithm via label set propagation.
– WELL (**WE**ak **L**abel multi-**L**abel method) (Sun et al, 2010) deals with missing labels via label propagation and controls the sparsity of label assignments.

Evaluation metrics Three criteria are used to evaluate the methods: Top-k precision (performance on a few top predictions) and F_1 score (including Macro F_1 and Micro F_1). In all cases, the experimental results of test data are computed based on the complete label matrix.

Each experiment is repeated for 30 times, and the average Top-k precision, Macro F_1 and Micro F_1 score on the unlabeled data are reported. We used libsvm (Chang and Lin, 2011) as implementation for BR. For ML-kNN method, the distance metric used to find nearest neighbors is set as the Euclidean distance and the parameter k is set to 10. For ECC method, the number of base classifiers chains is set to 10. For the CNMF method, all parameters are set to the recommended ones in the paper. Parameters in LEML method are set as default value implemented by the author. For our SafeML method, the number of base learners b is set to 5 for all the experiments in this paper. The kernel type of SVM classifiers trained by all methods are set as RBF kernel on all data sets except *enron*, *bibtex* and *tmc*2007 for the number of features are large enough and standard linear SVM classifiers are trained. In the SafeML method, we generate pseudo label matrices $\mathbf{P}$ by training b base learners on labeled data for each class. In order to construct diverse base learners, a subset of labeled data is sampled randomly for each base learner. Parameter γ_0 is set to the average number of relevant labels for each example in training set multiplied by the number of testing instances.

Table 4: Macro F_1 and Micro F_1 score for the compared methods and our SafeML method with 15% labeled examples. For all methods, if the performance is significantly better/worse than the baseline BR method, the corresponding entries are bolded/boxed (paired t-tests at 95% significance level). The average performance on all data sets is listed for comparison. The win/tie/loss counts are summarized and the method with the smallest number of losses against BR is bolded.

Macro-F1 score								
Data set	BR	S4VM	ECC	ML-kNN	CNMF	LEML	TRAM	SafeML
emotions	0.539	**0.608**	**0.589**	$\boxed{0.489}$	$\boxed{0.330}$	$\boxed{0.417}$	**0.586**	**0.624**
enron	0.076	0.082	0.083	0.067	**0.092**	**0.098**	**0.123**	**0.113**
image	0.105	**0.509**	**0.280**	**0.401**	**0.271**	**0.511**	**0.532**	**0.516**
scene	0.422	**0.702**	**0.596**	**0.617**	$\boxed{0.315}$	**0.567**	**0.684**	**0.657**
yeast	0.318	**0.405**	**0.346**	$\boxed{0.307}$	$\boxed{0.257}$	$\boxed{0.183}$	**0.355**	**0.408**
arts	0.075	**0.093**	**0.107**	0.068	**0.129**	**0.131**	**0.168**	**0.136**
bibtex	0.185	**0.204**	**0.247**	$\boxed{0.031}$	0.179	$\boxed{0.112}$	**0.229**	**0.272**
tmc2007	0.443	0.452	**0.474**	$\boxed{0.220}$	$\boxed{0.138}$	$\boxed{0.274}$	$\boxed{0.384}$	**0.475**
Ave. Perf.	0.279	0.381	0.340	0.275	0.214	0.286	0.383	0.408
win/tie/loss against BR		6/2/0	**7/1/0**	2/2/4	3/1/4	4/0/4	7/0/1	**8/0/0**

Micro F_1 score								
Data set	BR	S4VM	ECC	ML-kNN	CNMF	LEML	TRAM	SafeML
emotions	0.592	**0.619**	**0.632**	$\boxed{0.535}$	$\boxed{0.332}$	$\boxed{0.412}$	**0.612**	**0.648**
enron	0.477	**0.509**	**0.529**	$\boxed{0.434}$	$\boxed{0.351}$	0.485	**0.528**	**0.538**
image	0.130	**0.506**	**0.367**	**0.425**	**0.275**	**0.509**	**0.531**	**0.521**
scene	0.458	**0.690**	**0.603**	**0.622**	$\boxed{0.315}$	**0.555**	**0.693**	**0.635**
yeast	0.620	$\boxed{0.607}$	**0.643**	$\boxed{0.604}$	$\boxed{0.299}$	$\boxed{0.256}$	**0.638**	**0.656**
arts	0.186	**0.308**	**0.331**	$\boxed{0.160}$	**0.235**	**0.317**	**0.356**	**0.365**
bibtex	0.372	**0.398**	**0.449**	$\boxed{0.147}$	0.376	$\boxed{0.237}$	$\boxed{0.229}$	**0.509**
tmc2007	0.561	0.557	**0.604**	$\boxed{0.513}$	$\boxed{0.178}$	**0.580**	**0.624**	0.562
Ave. Perf.	0.424	0.525	0.520	0.430	0.295	0.419	0.527	0.556
win/tie/loss against BR		6/1/1	**8/0/0**	2/0/6	2/1/5	4/1/3	7/0/1	**7/1/0**

Table 5: Top-k precision for the compared methods and our proposed method with 15% labeled examples.

Data set		BR	ECC	ML-kNN	CNMF	LEML	TRAM	SafeML
emotions	P@1	0.601	**0.661**	**0.643**	$\boxed{0.346}$	**0.617**	**0.671**	**0.657**
	P@3	0.465	**0.492**	**0.497**	$\boxed{0.326}$	0.470	**0.515**	**0.508**
enron	P@1	0.116	**0.682**	$\boxed{0.067}$	**0.546**	**0.702**	**0.687**	**0.646**
	P@3	0.047	**0.567**	**0.068**	**0.421**	**0.549**	**0.537**	**0.572**
image	P@1	0.577	$\boxed{0.509}$	0.581	$\boxed{0.304}$	0.583	**0.589**	**0.628**
	P@3	0.355	$\boxed{0.295}$	0.348	$\boxed{0.257}$	0.361	0.353	0.357
scene	P@1	0.624	$\boxed{0.596}$	**0.695**	$\boxed{0.400}$	$\boxed{0.607}$	**0.709**	**0.651**
	P@3	0.309	$\boxed{0.107}$	**0.335**	$\boxed{0.239}$	**0.321**	**0.342**	0.313
yeast	P@1	0.733	**0.744**	**0.745**	$\boxed{0.273}$	$\boxed{0.538}$	0.740	**0.747**
	P@3	0.703	0.696	0.697	$\boxed{0.288}$	$\boxed{0.471}$	0.696	0.711
arts	P@1	0.198	**0.392**	**0.392**	**0.286**	**0.440**	**0.430**	**0.438**
	P@3	0.103	**0.237**	**0.255**	**0.203**	**0.265**	**0.269**	**0.238**
bibtex	P@1	0.424	$\boxed{0.247}$	$\boxed{0.318}$	$\boxed{0.365}$	$\boxed{0.407}$	**0.461**	0.430
	P@3	0.286	$\boxed{0.223}$	$\boxed{0.177}$	$\boxed{0.190}$	$\boxed{0.230}$	$\boxed{0.257}$	**0.297**
tmc2007	P@1	0.657	**0.711**	0.654	$\boxed{0.307}$	**0.738**	**0.740**	**0.704**
	P@3	0.482	**0.504**	0.474	$\boxed{0.183}$	**0.533**	**0.538**	**0.506**
Ave. Perf.		0.491	0.568	0.512	0.353	0.579	0.628	0.613
		0.344	0.390	0.356	0.263	0.400	0.430	0.438
win/tie/loss against BR			5/0/3	4/2/2	2/0/6	4/1/3	**7/1/0**	**7/1/0**
			4/1/3	4/3/1	2/0/6	4/2/2	5/2/1	**5/3/0**

4.2 Results on Semi-Supervised Multi-Label Learning

For each data set, we split 15% examples randomly as labeled data and other as unlabeled data. For BR method, a binary SVM classifier is trained for each class using only labeled data. For S4VM method, we train a S4VM classifier for each class with labeled and unlabeled data together.

The results measured in Macro F_1, Micro F_1 and Top-k precision are presented in Tables 4-5 and Figure 3. We can have the following observations.

- In terms of win counts, SafeML and ECC and TRAM perform the best on Macro F_1 and Micro F_1. SafeML and TRAM perform the best on Top-k precision. The other methods do not perform very well.
- In terms of average performance, SafeML obtains highly competitive performance with state-of-the-art methods on all the three multi-label evaluation metrics. SafeML obtains the best performance on Macro F_1 and Micro F_1.
- Importantly, in terms of loss counts, only SafeML does not degenerate performance significantly on three multi-label evaluation metrics, while the other methods all cause performance degeneration significantly in some cases.
- In both Macro F_1 and Micro F_1, S4VM degenerates performance seriously in some cases, pointing out that pure safe semi-supervised learning does not lead to safe multi-label predictions.
- Overall SafeML obtains highly competitive performance with state-of-the-art methods, while unlike compared methods that degenerate learning performance significantly in many cases, SafeML does not significantly hurt performance.

Table 6: Micro F_1 score for the compared methods and our proposed method for weak label learning setting.

Data set	Methods	80%	40%	20%	10%	5%
emotions	BR	0.090	0.659	0.739	0.774	0.740
	WELL	**0.161**	**0.704**	**0.783**	**0.808**	**0.821**
	LEML	**0.718**	**0.721**	0.724	$\boxed{0.723}$	0.731
	SafeML	**0.348**	**0.835**	**0.870**	**0.880**	**0.873**
enron	BR	0.301	0.556	0.624	0.632	0.662
	WELL	**0.362**	**0.604**	**0.763**	**0.848**	**0.851**
	LEML	**0.537**	**0.783**	**0.839**	**0.856**	**0.867**
	SafeML	**0.517**	**0.749**	**0.782**	**0.795**	**0.797**
image	BR	0.070	0.146	0.290	0.331	0.363
	WELL	**0.121**	**0.404**	**0.583**	**0.608**	**0.661**
	LEML	**0.120**	**0.314**	**0.403**	**0.436**	**0.446**
	SafeML	**0.086**	**0.602**	**0.753**	**0.793**	**0.792**
scene	BR	0.158	0.558	0.670	0.752	0.710
	WELL	**0.221**	$\boxed{0.443}$	$\boxed{0.553}$	$\boxed{0.612}$	$\boxed{0.671}$
	LEML	**0.295**	$\boxed{0.486}$	$\boxed{0.548}$	$\boxed{0.557}$	$\boxed{0.561}$
	SafeML	**0.414**	**0.811**	**0.861**	**0.874**	**0.878**
yeast	BR	0.209	0.627	0.702	0.725	0.733
	WELL	**0.251**	$\boxed{0.436}$	$\boxed{0.487}$	$\boxed{0.504}$	$\boxed{0.516}$
	LEML	**0.519**	0.627	$\boxed{0.633}$	$\boxed{0.634}$	$\boxed{0.661}$
	SafeML	**0.535**	**0.793**	**0.835**	**0.853**	**0.862**
arts	BR	0.050	0.238	0.305	0.300	0.334
	WELL	**0.123**	**0.343**	**0.403**	**0.436**	**0.441**
	LEML	**0.174**	**0.347**	**0.404**	**0.421**	**0.430**
	SafeML	**0.115**	**0.377**	**0.441**	**0.469**	**0.465**
bibtex	BR	0.292	0.476	0.525	0.552	0.558
	WELL	$\boxed{0.278}$	0.473	**0.579**	**0.600**	**0.631**
	LEML	$\boxed{0.204}$	$\boxed{0.364}$	$\boxed{0.446}$	$\boxed{0.485}$	$\boxed{0.500}$
	SafeML	**0.629**	**0.609**	**0.695**	**0.719**	**0.724**
tmc2007	BR	0.428	0.670	0.733	0.745	0.757
	WELL	**0.475**	**0.802**	**0.838**	**0.850**	**0.853**
	LEML	$\boxed{0.242}$	$\boxed{0.551}$	$\boxed{0.610}$	$\boxed{0.630}$	$\boxed{0.638}$
	SafeML	**0.765**	**0.890**	**0.909**	**0.917**	**0.922**
Ave. Perf.	BR	0.200	0.491	0.574	0.601	0.607
	WELL	0.249	0.526	0.624	0.658	0.681
	LEML	0.351	0.524	0.576	0.593	0.604
	SafeML	0.373	0.708	0.768	0.788	0.789

4.3 Results on Weak Label Learning

For each data set, we create training data sets with varying portions of labels, ranging from 20% (i.e., 80% of the whole training label matrix is missing) to 95% (i.e., 5% of the whole training label matrix is missing). In each case, the missing labels are randomly chosen among positive examples of each class.

The results measured in Micro F_1 and Macro F_1 are presented in Tables 6-7. We can have the following observations.

– As the number of missing relevant labels decreases, all methods generally clearly improve the learning performance.

Table 7: Macro F_1 score for the compared methods and our proposed method for weak label learning setting.

Data set	Methods	80%	40%	20%	10%	5%
emotions	BR	0.093	0.630	0.687	0.735	0.687
	WELL	**0.274**	**0.802**	**0.838**	**0.850**	**0.853**
	LEML	**0.705**	**0.714**	**0.712**	$\boxed{0.715}$	**0.721**
	SafeML	**0.323**	**0.801**	**0.841**	**0.859**	**0.844**
enron	BR	0.075	0.166	0.180	0.189	0.186
	WELL	**0.174**	**0.306**	**0.338**	**0.350**	**0.366**
	LEML	**0.186**	**0.347**	**0.408**	**0.453**	**0.447**
	SafeML	**0.130**	**0.250**	**0.272**	**0.283**	**0.253**
image	BR	0.068	0.129	0.236	0.276	0.299
	WELL	**0.074**	**0.206**	**0.246**	**0.350**	**0.366**
	LEML	**0.108**	**0.265**	**0.348**	**0.405**	**0.412**
	SafeML	**0.078**	**0.591**	**0.753**	**0.789**	**0.788**
scene	BR	0.144	0.526	0.654	0.732	0.715
	WELL	**0.176**	$\boxed{0.501}$	0.646	$\boxed{0.651}$	$\boxed{0.680}$
	LEML	**0.194**	**0.532**	0.648	$\boxed{0.657}$	**0.769**
	SafeML	**0.375**	**0.813**	**0.863**	**0.876**	**0.882**
yeast	BR	0.105	0.326	0.385	0.413	0.420
	WELL	**0.257**	**0.446**	**0.484**	**0.499**	**0.511**
	LEML	**0.373**	**0.480**	**0.483**	**0.486**	**0.485**
	SafeML	**0.234**	**0.464**	**0.532**	**0.562**	**0.575**
arts	BR	0.019	0.110	0.141	0.151	0.159
	WELL	**0.066**	**0.142**	0.144	0.151	**0.178**
	LEML	**0.073**	**0.157**	**0.191**	**0.200**	**0.209**
	SafeML	**0.046**	**0.171**	**0.204**	**0.224**	**0.215**
bibtex	BR	0.128	0.326	0.384	0.412	0.388
	WELL	**0.220**	**0.391**	**0.412**	**0.452**	**0.444**
	LEML	**0.214**	$\boxed{0.295}$	**0.448**	**0.457**	**0.482**
	SafeML	**0.506**	**0.479**	**0.577**	**0.602**	**0.581**
tmc2007	BR	0.387	0.567	0.606	0.615	0.623
	WELL	**0.474**	**0.787**	**0.844**	**0.859**	**0.862**
	LEML	0.384	**0.650**	**0.714**	**0.733**	**0.740**
	SafeML	**0.639**	**0.799**	**0.824**	**0.839**	**0.848**
Ave. Perf.	BR	0.127	0.348	0.409	0.440	0.435
	WELL	0.214	0.448	0.494	0.520	0.533
	LEML	0.280	0.430	0.494	0.513	0.533
	SafeML	0.291	0.546	0.608	0.629	0.623

– Although WELL generally improves performance significantly (30 cases in Table 6 and 34 cases in Table 7), it significantly decreases the learning performance in 9 cases in Table 6 and 3 cases in Table 7, where most cases happen on few missing relevant labels. The reason may owe to the fact that the baseline BR method becomes more competitive and thus WELL turns to be risky.
– LEML also often improves the learning performance (19 cases in Table 6 and 35 cases in Table 7), however, it still significantly decreases the learning performance in 18 cases in Table 6 and 3 cases in Table 7. Under the same reason, LEML typically degenerates the performance on few missing relevant labels.
– SafeML significantly improves the learning performance in 40 cases in terms of both the Micro F_1 and Macro F_1 metrics. More importantly, it does not suffer from performance degeneration on all the 80 cases. Further more, SafeML obtains the best average performance among all the comparison methods.

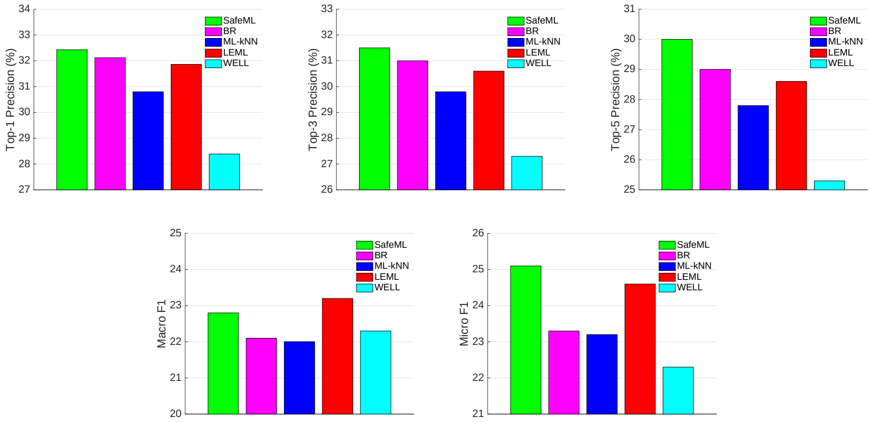

Fig. 3: Performance for the compared methods and our proposed method with 15% labeled examples.

Figure 4 shows the results of Top-k precision on three representative data sets. The results on other data sets perform similarly. SafeML performs highly competitive performance with compared methods, while unlike compared methods that degenerate learning performance significantly in many cases, SafeML does not significantly hurt performance compared with baseline BR method.

4.4 Results on Extended Weak Label Learning

For extended weak label learning, we create training data sets with varying portions of labels, ranging from 20% (i.e., 80% of the whole training label matrix is missing) to 95% (i.e., 5% of the whole training label matrix is missing). The missing labels are randomly chosen among positive and negative examples of each class.

The results measured in Micro F_1 and Macro F_1 are presented in Tables 8-9. We can have the following observations.

- WELL improves performance significantly (23 cases in Table 8 and 26 cases in Table 9), however it significantly decreases the learning performance in 8 cases in Table 8 and 6 cases in Table 9.
- LEML also often improves the learning performance (29 cases in Table 8 and 34 cases in Table 9), however, it still significantly decreases the learning performance in 7 cases in Table 8 and 5 cases in Table 9.
- SafeML significantly improves the learning performance in 39/38 cases in terms of Micro F_1 and Macro F_1, respectively. More importantly, it does not suffer from performance degeneration on all the 80 cases. Moreover, SafeML obtains the best average performance among all the comparison methods.

Figure 5 shows the results of Top-k precision on three representative data sets. Results on other data sets perform similarly. SafeML obtains highly competitive performance with compared methods, while unlike compared methods that

Table 8: Micro F_1 score for the compared methods and our proposed method for extended weak label learning setting.

Data set	Methods	80%	40%	20%	10%	5%
emotions	BR	0.634	0.679	0.697	0.681	0.648
	WELL	0.620	**0.681**	0.692	$\boxed{0.652}$	**0.664**
	LEML	**0.646**	**0.700**	**0.717**	**0.720**	**0.721**
	SafeML	**0.663**	**0.696**	0.700	**0.701**	**0.710**
enron	BR	0.510	0.545	0.556	0.548	0.534
	WELL	$\boxed{0.421}$	$\boxed{0.489}$	$\boxed{0.502}$	0.532	**0.564**
	LEML	**0.539**	**0.734**	**0.745**	**0.754**	**0.760**
	SafeML	**0.550**	**0.569**	**0.573**	**0.576**	**0.565**
image	BR	0.134	0.292	0.344	0.325	0.322
	WELL	**0.220**	**0.383**	**0.399**	**0.442**	**0.464**
	LEML	**0.484**	**0.470**	**0.464**	**0.460**	**0.464**
	SafeML	**0.531**	**0.618**	**0.631**	**0.636**	**0.639**
scene	BR	0.499	0.670	0.704	0.702	0.700
	WELL	$\boxed{0.420}$	**0.698**	0.690	**0.722**	**0.740**
	LEML	$\boxed{0.381}$	0.678	$\boxed{0.653}$	**0.742**	**0.740**
	SafeML	**0.695**	**0.736**	**0.749**	**0.749**	**0.752**
yeast	BR	0.628	0.651	0.651	0.666	0.667
	WELL	$\boxed{0.530}$	0.652	0.651	0.664	0.669
	LEML	$\boxed{0.497}$	$\boxed{0.523}$	$\boxed{0.532}$	$\boxed{0.532}$	$\boxed{0.534}$
	SafeML	**0.654**	**0.675**	**0.676**	**0.680**	**0.680**
arts	BR	0.230	0.310	0.331	0.346	0.350
	WELL	**0.273**	**0.396**	**0.428**	**0.429**	**0.434**
	LEML	**0.407**	**0.440**	**0.438**	**0.437**	**0.438**
	SafeML	**0.321**	**0.397**	**0.410**	**0.413**	**0.422**
bibtex	BR	0.403	0.523	0.548	0.558	0.548
	WELL	$\boxed{0.325}$	**0.551**	**0.552**	0.557	**0.564**
	LEML	**0.460**	**0.562**	**0.570**	**0.577**	**0.581**
	SafeML	**0.629**	**0.609**	**0.695**	**0.719**	**0.724**
tmc2007	BR	0.573	0.638	0.649	0.652	0.652
	WELL	$\boxed{0.485}$	**0.708**	**0.734**	**0.750**	**0.753**
	LEML	**0.625**	**0.641**	0.645	0.645	0.645
	SafeML	**0.765**	**0.890**	**0.909**	**0.917**	**0.922**
Ave. Perf.	BR	0.451	0.539	0.560	0.560	0.553
	WELL	0.424	0.570	0.581	0.594	0.607
	LEML	0.505	0.594	0.596	0.608	0.610
	SafeML	0.601	0.649	0.668	0.674	0.678

degenerate learning performance significantly in many cases, SafeML does not significantly hurt performance compared with baseline BR method.

4.5 Theoretical Analysis

To generate pseudo label matrices, we train b base learners, which takes $O(bdN_{train}L)$ time and N_{train} is usually far less than the size of whole data sets. At each iteration of our cutting-plane algorithm, we get $\hat{\mathbf{Y}}$ by solving Eq. (15) as a linear programming, which takes $O(N_{test}L)$ time. And then in order to find the most violated constraint for the current $\hat{\mathbf{Y}}$, we solve a simple linear programming which takes $O(b^3)$ time. In total, this takes $O(tNL)$ time, where t is the number of iter-

Table 9: Macro F_1 score for the compared methods and our proposed method for extended weak label learning setting.

Data set	Methods	80%	40%	20%	10%	5%
emotions	BR	0.593	0.652	0.671	0.647	0.619
	WELL	$\boxed{0.520}$	$\boxed{0.591}$	0.672	**0.673**	**0.663**
	LEML	**0.644**	**0.701**	**0.710**	**0.708**	**0.712**
	SafeML	**0.646**	**0.680**	**0.683**	**0.683**	**0.678**
enron	BR	0.133	0.168	0.168	0.153	0.140
	WELL	0.120	**0.171**	0.168	**0.168**	**0.164**
	LEML	**0.320**	**0.397**	**0.417**	**0.426**	**0.433**
	SafeML	**0.153**	**0.192**	**0.202**	**0.198**	**0.192**
image	BR	0.112	0.239	0.282	0.268	0.290
	WELL	0.120	**0.381**	**0.392**	**0.452**	**0.464**
	LEML	**0.460**	**0.456**	**0.416**	**0.427**	**0.431**
	SafeML	**0.519**	**0.622**	**0.635**	**0.639**	**0.638**
scene	BR	0.461	0.662	0.697	0.698	0.681
	WELL	$\boxed{0.420}$	0.661	**0.702**	**0.702**	**0.694**
	LEML	$\boxed{0.367}$	$\boxed{0.620}$	0.700	**0.741**	**0.746**
	SafeML	**0.705**	**0.748**	**0.760**	**0.762**	**0.759**
yeast	BR	0.327	0.363	0.370	0.379	0.376
	WELL	0.330	**0.381**	**0.392**	**0.422**	**0.464**
	LEML	**0.447**	**0.474**	**0.485**	**0.484**	**0.485**
	SafeML	**0.399**	**0.439**	**0.447**	**0.449**	**0.452**
arts	BR	0.177	0.143	0.157	0.164	0.197
	WELL	$\boxed{0.120}$	**0.181**	**0.192**	**0.191**	$\boxed{0.164}$
	LEML	**0.200**	**0.214**	**0.216**	**0.217**	**0.216**
	SafeML	$\boxed{0.124}$	**0.180**	**0.193**	**0.193**	0.198
bibtex	BR	0.221	0.379	0.409	0.412	0.377
	WELL	0.220	**0.381**	0.392	**0.452**	**0.464**
	LEML	$\boxed{0.205}$	$\boxed{0.356}$	**0.535**	**0.571**	**0.580**
	SafeML	**0.506**	**0.479**	**0.577**	**0.602**	**0.581**
tmc2007	BR	0.452	0.501	0.513	0.520	0.525
	WELL	$\boxed{0.420}$	**0.581**	**0.592**	**0.625**	**0.664**
	LEML	$\boxed{0.338}$	**0.550**	**0.549**	**0.548**	**0.558**
	SafeML	**0.639**	**0.799**	**0.824**	**0.839**	**0.848**
Ave. Perf.	BR	0.310	0.388	0.408	0.405	0.401
	WELL	0.284	0.416	0.438	0.461	0.468
	LEML	0.373	0.471	0.504	0.515	0.520
	SafeML	0.461	0.517	0.540	0.546	0.543

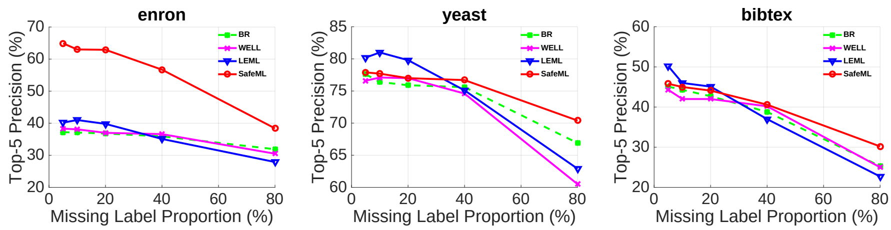

Fig. 4: Top-5 precision on weak label learning

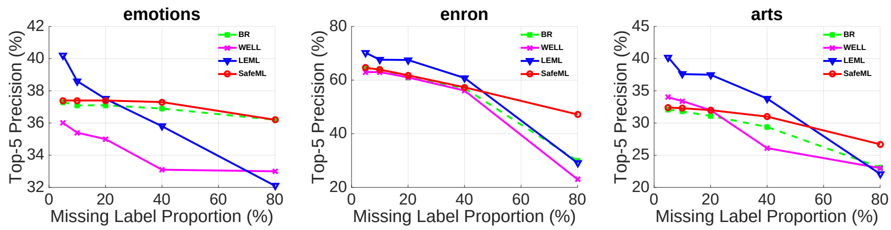

Fig. 5: Top-5 precision on extended weak label learning

ations and no more than 100 in our experiments. Besides, the convergence rate of our algorithm on two representative data sets is shown in Figure 6, from which we can see that it converges in an efficient manner. The convergence rate of our proposal on other data sets performs similarly. In summary, the proposed method is computationally efficient

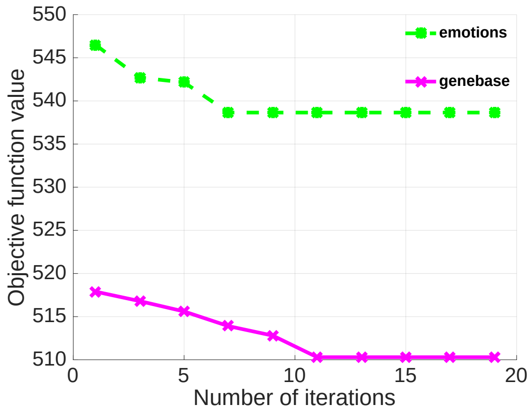

Fig. 6: The convergence rate of our proposal

5 Conclusion

Multi-label learning with weakly labeled data is commonly occurred in reality. This includes, e.g., (i) semi-supervised multi-label learning where completely labeled examples are partially known; (ii) weak label learning where relevant labels of examples are partially known; iii) extended weak label learning where relevant and irrelevant labels of examples are partially known. In this paper, we study to learn safe multi-label predictions for weakly labeled data, which means multi-label learning method does not hurt performance when using weakly labeled data. To overcome this issue, in this work we explicitly optimize multi-label evaluation metrics (F_1 score and Top-k precision) via formulating ground-truth label assignments are from a convex combination of basic multi-label learners. Although the optimization suffers from infinite number of possible ground-truth label assignments, cutting-plane strategy is adopted to iteratively generate the most helpful label assignments and consequently efficiently solve the optimization. Extensive experimental results on three weakly labeled learning tasks, namely, i) semi-supervised multi-label learning; ii) weak-label learning and iii) extended weak-label learning, show that our proposal clearly improves the safeness when using weakly labeled data in comparison to many state-of-the-art methods.

References

Balsubramani A, Freund Y (2015) Optimally combining classifiers using unlabeled data. In: Proceedings of the 28th Conference on Learning Theory, Paris, France, pp 211–225

Bucak SS, Jin R, Jain AK (2011) Multi-label learning with incomplete class assignments. In: Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 2801–2808

Carneiro G, Chan AB, Moreno PJ, Vasconcelos N (2007) Supervised learning of semantic classes for image annotation and retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(3):394–410

Chang CC, Lin CJ (2011) Libsvm: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2(3):27:1–27:27

Chapelle O, Scholkopf B, Zien A (2009) Semi-supervised learning (chapelle, o. et al., eds.; 2006). IEEE Transactions on Neural Networks 20(3):542–542

Chen G, Song Y, Wang F, Zhang C (2008) Semi-supervised multi-label learning by solving a sylvester equation. In: Proceedings of the 8th SIAM International Conference on Data Mining, SIAM, Atlanta, GA, pp 410–419

Chen M, Zheng AX, Weinberger KQ (2013) Fast image tagging. In: Proceedings of the 30th International Conference of Machine Learning, Atlanta, GA, pp 1274–1282

Kong X, Ng MK, Zhou ZH (2013) Transductive multilabel learning via label set propagation. IEEE Transactions on Knowledge and Data Engineering 25(3):704–719

Krijthe JH, Loog M (2015) Implicitly constrained semi-supervised least squares classification. In: Proceedings of the 14th International Symposium on Intelligent Data Analysis, Springer, Saint-Etienne, France, pp 158–169

Li YF, Zhou ZH (2015) Towards making unlabeled data never hurt. IEEE Transactions on Pattern Analysis and Machine Intelligence 37(1):175–188

Li YF, Kwok JT, Zhou ZH (2016) Towards safe semi-supervised learning for multivariate performance measures. In: Proceedings of 30th AAAI Conference on Artificial Intelligence, Phoenix, AZ, pp 1816–1822

Li YF, Zha HW, Zhou ZH (2017) Learning safe prediction for semi-supervised regression. In: Proceedings of the 31th AAAI Conference on Artificial Intelligence, San Francisco, CA, in press

Liu Y, Jin R, Yang L (2006) Semi-supervised multi-label learning by constrained non-negative matrix factorization. In: Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1, AAAI Press, pp 421–426

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

Srivastava AN, Zane-Ulman B (2005) Discovering recurring anomalies in text reports regarding complex space systems. In: Proceedings of the 25th IEEE Aerospace Conference, IEEE, pp 3853–3862

Sun YY, Zhang Y, Zhou ZH (2010) Multi-label learning with weak label. In: Proceedings of the 24th AAAI Conference on Artificial Intelligence, Citeseer, Atlanta, GA, pp 593–598

Tsoumakas G, Katakis I, Vlahavas I (2009) Mining multi-label data. In: Data mining and knowledge discovery handbook, Springer, pp 667–685

Wang B, Tu Z, Tsotsos JK (2013) Dynamic label propagation for semi-supervised multi-class multi-label classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp 425–432

Wu L, Jin R, Jain AK (2013) Tag completion for image retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 35(3):716–727

Yu HF, Jain P, Kar P, Dhillon IS (2014) Large-scale multi-label learning with missing labels. In: Proceedings of the 31st International Conference of Machine Learning, Beijing, China, pp 593–601

Zhang ML, Zhou ZH (2007) Ml-knn: A lazy learning approach to multi-label learning. Pattern Recognition 40(7):2038–2048

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

Zhao F, Guo Y (2015) Semi-supervised multi-label learning with incomplete labels. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, pp 4062–4068

Zhu G, Yan S, Ma Y (2010) Image tag refinement towards low-rank, content-tag prior and error sparsity. In: Proceedings of the 18th ACM International Conference on Multimedia, ACM, Firenze, Italy, pp 461–470