Teaching-to-Learn and Learning-to-Teach for Multi-Label Propagation

Chen Gong[†,⋆] and **Dacheng Tao**[⋆] and **Jie Yang**[†] and **Wei Liu** [♯]
[†]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University
[⋆]Centre for Quantum Computation and Intelligent Systems, University of Technology Sydney
[♯]Didi Research, Beijing, China
{goodgongchen, jieyang}@sjtu.edu.cn
dacheng.tao@uts.edu.au
weiliu@didichuxing.com

Abstract

Multi-label propagation aims to transmit the multi-label information from labeled examples to unlabeled examples based on a weighted graph. Existing methods ignore the specific propagation difficulty of different unlabeled examples and conduct the propagation in an imperfect sequence, leading to the error-prone classification of some difficult examples with uncertain labels. To address this problem, this paper associates each possible label with a "teacher", and proposes a "**M**ulti-**L**abel **T**eaching-to-**L**earn and **L**earning-to-**T**each" (ML-TLLT) algorithm, so that the entire propagation process is guided by the teachers and manipulated from simple examples to more difficult ones. In the teaching-to-learn step, the teachers select the simplest examples for the current propagation by investigating both the definitiveness of each possible label of the unlabeled examples, and the dependencies between labels revealed by the labeled examples. In the learning-to-teach step, the teachers reversely learn from the learner's feedback to properly select the simplest examples for the next propagation. Thorough empirical studies show that due to the optimized propagation sequence designed by the teachers, ML-TLLT yields generally better performance than seven state-of-the-art methods on the typical multi-label benchmark datasets.

Introduction

Multi-Label Learning (MLL) refers to the problem in which an example can be assigned a set of different labels. So far, MLL has been intensively adopted in image annotation (Wang, Huang, and Ding 2009), text categorization (Schapire and Singer 2000), social behavior learning (Tang and Liu 2009), and others.

By following the taxonomy presented in (Zhang and Zhou 2014), we classify the existing MLL algorithms into *problem transformation methods* and *algorithm adaptation methods*. Problem transformation methods cast MML into other well-studied scenarios. Representative approaches include Calibrated Label Ranking (Fürnkranz et al. 2008) which transforms MLL into a label ranking problem, and Binary Relevance (Boutell et al. 2004) which regards MLL as a series of binary classification tasks.

Algorithm adaptation methods extend the existing learning algorithms to multi-label cases. For example, (Zhang and Zhou 2007) adapt the traditional KNN classifier to multi-label KNN, (Clare and King 2001; Bi and Kwok 2011) deploy the tree model to analyze the MLL problem, and (Elisseeff and Weston 2001; Xu, Li, and Zhou 2013; Xu, Tao, and Xu 2015) develop various multi-label SVMs by introducing the ranking loss, PRO loss, and the causality between labels, respectively.

Although the above methods differ from one another, they all focus on how to exploit the label correlations to optimize learning performance. Since graph is a simple yet powerful tool to model the relationship between labels or examples, several researchers have recently introduced graph to MLL problem. Representative works include (Kong, Ng, and Zhou 2013; Chen et al. 2013; Wang, Huang, and Ding 2009; Chen et al. 2008; Jiang 2012; Kang, Jin, and Sukthankar 2006; Zha et al. 2008; Wang, Tu, and Tsotsos 2013). However, above graph-based propagation methods often suffer from unsatisfactory performance due to the unexpected noise (*e,g,* outliers) in the sample space and the huge label search space (the size is 2^q where q is the number of possible labels). To make matters worse, they propagate the labels to unlabeled examples in an unfavorable sequence without considering their individual propagation difficulty or reliability. For example, (Wang, Huang, and Ding 2009; Chen et al. 2008; Kong, Ng, and Zhou 2013) treat all the unlabeled examples equally and conduct a one-shot label propagation by minimizing the designed energy function. (Jiang 2012; Wang, Tu, and Tsotsos 2013) iteratively transfer the label information to unlabeled examples as long as these examples are directly linked to the labeled examples on a graph.

Inspired by (Gong et al. 2015), we address the above problem by proposing a novel iterative multi-label propagation scheme called "**M**ulti-**L**abel **T**eaching-to-**L**earn and **L**earning-to-**T**each" (ML-TLLT), to explicitly manipulate the propagation sequence, so that the unlabeled examples are logically propagated from simple to difficult. This is beneficial to improving propagation quality because the previously attained simple knowledge eases the learning burden for the subsequent difficult examples. Generally, an example is simple if its labels can be confidently decided. Suppose we have $n=l+u$ examples $\mathcal{X}=\{\mathbf{x}_1,\cdots,\mathbf{x}_l,\mathbf{x}_{l+1},\cdots,\mathbf{x}_{l+u}\}$, where the first l elements constitute the labeled set $\mathcal{L}$ and the remaining u examples form the unlabeled set $\mathcal{U}$. Each element $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^d$ is associated with a set of q possible labels

encoded in a label vector $\mathbf{Y}_i = (\mathbf{Y}_{i1}, \cdots, \mathbf{Y}_{iq}) \in \{0,1\}^q$, where $\mathbf{Y}_{ir} = 1$ $(r = 1, \cdots, q)$ means that $\mathbf{x}_i$ has the label r, and 0 otherwise. Our target is to iteratively propagate the labels $\mathbf{Y}_1, \cdots, \mathbf{Y}_l$ from $\mathcal{L}$ to $\mathcal{U}$ based on the established KNN graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ (see Fig.1(a)). Here $\mathcal{V}$ is the node set corresponding to the total n examples, and $\mathcal{E}$ is the edge set representing the similarities between these nodes.

We associate each of the q labels with a "teacher" (see Fig.1(b)). In the teaching-to-learn step of a single propagation, the r-th ($r = 1, \cdots, q$) teachers estimate the difficulty of $\mathbf{x}_i \in \mathcal{U}$ by evaluating the r-th label definitiveness $M_r(\mathbf{x}_i)$ from their own viewpoints. The correlations between labels are also considered by the teachers and then encoded in the variables $Q_{rp} \in [0,1]$ $(r, p = 1, \cdots, q)$. Based on the label-specific definitiveness and the pairwise label correlations, the simplest examples are determined by individual teachers (recorded by the selection matrix $\mathbf{S}^{(1)}, \cdots, \mathbf{S}^{(q)}$), after which the overall simplest examples agreed by all the teachers are placed into the curriculum set $\mathcal{S}^*$. The state-of-the-art TRAnsductive Multi-label (TRAM) algorithm (Kong, Ng, and Zhou 2013) is adopted as a "learner" to reliably propagate the labels to the designated curriculum $\mathcal{S}^*$ (see Fig.1(c)). In the learning-to-teach step, the learner delivers learning feedback to the teachers to assist them in deciding the subsequent suitable curriculum. The above ML-TLLT process iterates with the labeled set and unlabeled set respectively updated by $\mathcal{L} := \mathcal{L} \cup \mathcal{S}^*$ and $\mathcal{U} := \mathcal{U} - \mathcal{S}^*$, and terminates when $\mathcal{U} = \varnothing$. As a result, all the original unlabeled examples are assigned reliable labels $\mathbf{Y}_{l+1}, \cdots, \mathbf{Y}_{l+u}$.

Our work is different from active learning (Settles 2010) because active learning needs a human labeler to label the selected examples while our method does not. Our work also differs from curriculum learning (Bengio et al. 2009; Kumar, Packer, and Koller 2010; Jiang et al. 2015; Khan, Mutlu, and Zhu 2011) in that ML-TLLT requires the interaction between a teaching committee and a learner.

Our Approach

We denote $\mathbf{W}$ as the adjacency matrix of $\mathcal{G}$ with the (i,j)-th element $\mathbf{W}_{ij} = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / (2\xi^2)\right)$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ are linked by an edge, and $\mathbf{W}_{ij} = 0$ otherwise. Here ξ is the kernel width decided as the average Euclidean distance between all pairs of examples (Kong, Ng, and Zhou 2013). Based on $\mathbf{W}$, we have the diagonal degree matrix $\mathbf{D}_{ii} = \sum_{j=1}^{n} \mathbf{W}_{ij}$ and graph Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$. We stack $\mathbf{Y}_1, \cdots, \mathbf{Y}_n$ into a label matrix $\mathbf{Y} = \left(\mathbf{Y}_1^\top, \cdots, \mathbf{Y}_n^\top\right)^\top$, and the correlation between labels r and p is then computed by $Q_{rp} = cos(\mathbf{Y}_{\mathcal{L},r}, \mathbf{Y}_{\mathcal{L},p}) = \frac{\langle \mathbf{Y}_{\mathcal{L},r}, \mathbf{Y}_{\mathcal{L},p} \rangle}{\|\mathbf{Y}_{\mathcal{L},r}\| \|\mathbf{Y}_{\mathcal{L},p}\|}$ with $\mathbf{Y}_{\mathcal{L},r} = (\mathbf{Y}_{1r}, \cdots, \mathbf{Y}_{lr})^\top$. Similarly, we also define a label score matrix $\mathbf{F} = \left(\mathbf{F}_1^\top, \cdots, \mathbf{F}_n^\top\right)^\top$ with every element $\mathbf{F}_{ir} \geq 0$ denoting the possibility of $\mathbf{x}_i$ belonging to the class r.

Teaching-to-learn Step

In each propagation, all the unlabeled examples that are directly connected to $\mathcal{L}$ are included in the candidate set $\mathcal{B}$ of size b, and the target of the teachers is to pick up the simplest curriculum examples $\mathcal{S}^* = \{\mathbf{x}_1^*, \cdots, \mathbf{x}_s^*\}$ from $\mathcal{B}$ where s is the number of selected examples. Such selection should consider both the definitiveness of individual possible label, and the correlation between the pairs of labels.

The example $\mathbf{x}_i \in \mathcal{B}$ is simple in terms of the r-th label if $\mathbf{Y}_{ir}$ is definitely 1 or 0. Let $\mathcal{C}_r$ (or $\bar{\mathcal{C}}_r$) as the set including all the labeled examples with the r-th label 1 (or 0), the definitiveness of $\mathbf{x}_i$'s r-th label is then modeled by

$$M_r(\mathbf{x}_i) = |\tilde{T}(\mathbf{x}_i, \mathcal{C}_r) - \tilde{T}(\mathbf{x}_i, \bar{\mathcal{C}}_r)|, \tag{1}$$

where $\tilde{T}(\mathbf{x}_i, \mathcal{C}_r)$ (or $\tilde{T}(\mathbf{x}_i, \bar{\mathcal{C}}_r)$) represents the average commute time between $\mathbf{x}_i$ and all the elements in the set $\mathcal{C}_r$ (or $\bar{\mathcal{C}}_r$). That is,

$$\begin{cases} \tilde{T}(\mathbf{x}_i, \mathcal{C}_r) = \dfrac{1}{|\mathcal{C}_r|} \sum\limits_{\mathbf{x}_{i'} \in \mathcal{C}_r} T(\mathbf{x}_i, \mathbf{x}_{i'}) \\ \tilde{T}(\mathbf{x}_i, \bar{\mathcal{C}}_r) = \dfrac{1}{|\bar{\mathcal{C}}_r|} \sum\limits_{\mathbf{x}_{i'} \notin \mathcal{C}_r} T(\mathbf{x}_i, \mathbf{x}_{i'}) \end{cases}. \tag{2}$$

In (2), the notation "$|\cdot|$" computes the size of the corresponding set, and $T(\mathbf{x}_i, \mathbf{x}_{i'})$ denotes the commute time (Qiu and Hancock 2007) between $\mathbf{x}_i$ and $\mathbf{x}_{i'}$, which is

$$T(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum\nolimits_{k=1}^{n} h(\lambda_k) \left(\mathbf{u}_{ki} - \mathbf{u}_{ki'}\right)^2, \tag{3}$$

where $0 = \lambda_1 \leq \cdots \leq \lambda_n$ are the eigenvalues of Laplacian matrix $\mathbf{L}$, and $\mathbf{u}_1, \cdots, \mathbf{u}_n$ are the associated eigenvectors; $\mathbf{u}_{ki}$ denotes the i-th element of $\mathbf{u}_k$; $h(\lambda_k) = 1/\lambda_k$ if $\lambda_k \neq 0$ and $h(\lambda_k) = 0$ otherwise. Commute time $T(\mathbf{x}_i, \mathbf{x}_{i'})$ describes the time cost starting from $\mathbf{x}_i$, reaching $\mathbf{x}_{i'}$, and then returning to $\mathbf{x}_i$ again, therefore it can be leveraged to describe the closeness of two examples. The larger $M_r(\mathbf{x}_i)$ is, the simpler $\mathbf{x}_i$ is in terms of the label r.

To consider the correlations between labels, we force the two teachers of labels r and p to generate similar curriculums if the two labels are highly correlated over the labeled examples (*i.e.* Q_{rp} is large). Based on above considerations, we introduce a binary example selection matrix $\mathbf{S}^{(r)} \in \{1,0\}^{b \times s}$ for the r-th label ($r = 1, \cdots, q$). The element $\mathbf{S}_{ij}^{(r)} = 1$ means that the r-th teacher considers the i-th example to be simple, and it should therefore be included as the j-th element in the curriculum set. The final selected examples agreed by all q teachers are indicated by the matrix $\mathbf{S}^*$, which has the same definition as $\mathbf{S}^{(r)}$. Therefore, the model for the selection of examples is

$$\begin{aligned} \min_{\mathbf{S}^{(1)}, \cdots, \mathbf{S}^{(q)}, \mathbf{S}^*} & \sum\nolimits_{r=1}^{q} \mathrm{tr}\left(\mathbf{S}^{(r)\top} \mathbf{M}^{(r)-1} \mathbf{S}^{(r)}\right) \\ & + \beta_0 \sum\nolimits_{r,p=1}^{q} Q_{rp} \left\| \mathbf{S}^{(r)} - \mathbf{S}^{(p)} \right\|^2 + \beta_1 \sum\nolimits_{r=1}^{q} \left\| \mathbf{S}^{(r)} - \mathbf{S}^* \right\|^2, \\ \text{s.t. } & \mathbf{S}^* \in \{1,0\}^{b \times s},\ \mathbf{S}^{*\top} \mathbf{S}^* = \mathbf{I}_{s \times s}, \\ & \mathbf{S}^{(r)} \in \{1,0\}^{b \times s},\ \mathbf{S}^{(r)\top} \mathbf{S}^{(r)} = \mathbf{I}_{s \times s},\ \text{for } r = 1, \cdots, q \end{aligned} \tag{4}$$

where $\mathbf{M}^{(r)}$ is a diagonal matrix with the diagonal elements $\mathbf{M}_{ii}^{(r)} = M_r(\mathbf{x}_i)$ for any $\mathbf{x}_i \in \mathcal{B}$. The first *definitiveness term* in the objective function investigates the definitiveness of $\mathbf{x}_i$'s all q labels and regulates the i-th row of $\mathbf{S}^{(r)}$ to zeros if $M_r(\mathbf{x}_i)$ is small. The second *label correlation term* discovers the label dependencies to make the

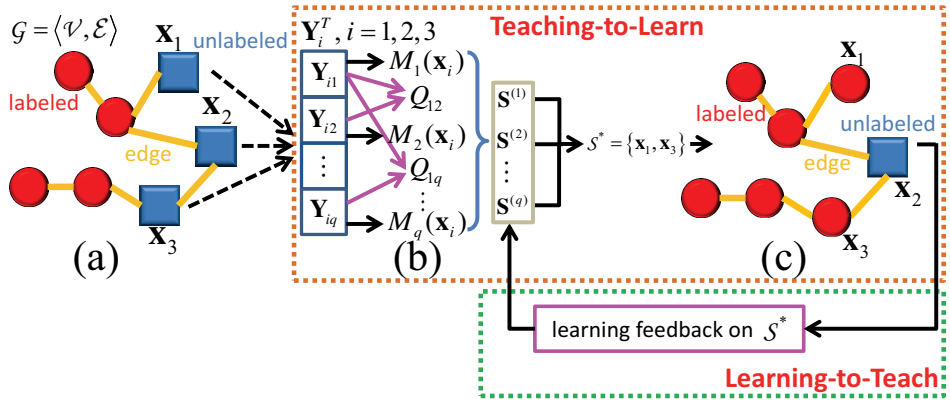

Figure 1: The framework of our algorithm. (a) illustrates the established graph, in which the red balls, blue squares and yellow lines represent the labeled examples, unlabeled examples and edges, respectively. In (b), each of the q possible labels $\mathbf{Y}_{i1},\cdots,\mathbf{Y}_{iq}$ is associated with a teacher, who evaluates the corresponding label definitiveness $M_r(\mathbf{x}_i)$ (r takes a value from $1,\cdots,q$) on all the unlabeled $\mathbf{x}_i$ ($i=1,2,3$ in this figure). By incorporating the label correlations (magenta arrows) recorded by Q_{rp} ($r,p=1\cdots,q$), the individual decisions $\mathbf{S}^{(1)},\cdots,\mathbf{S}^{(q)}$ are made and then unified to an overall simplest curriculum set $\mathcal{S}^*=\{\mathbf{x}_1,\mathbf{x}_3\}$. These curriculum examples are classified by the learner in (c). Lastly, a learning feedback on $\mathcal{S}^*$ is generated to help the teachers decide the next suitable curriculum.

highly correlated labels produce similar selection matrices. The third *consistency term* integrates the selection matrices decided by various teachers to a consistent result. The positive β_0 and β_1 are trade-off parameters. The orthogonality constraints $\mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}_{s\times s}$ ($\mathbf{I}$ denotes the identity matrix) and $\mathbf{S}^{*\top}\mathbf{S}^*=\mathbf{I}_{s\times s}$ ensure that every example is selected only once in $\mathbf{S}^{(r)}$ and $\mathbf{S}^*$.

However, the above problem (4) is NP-hard due to the discrete $\{1,0\}$-constraints on $\mathbf{S}^{(r)}$ and $\mathbf{S}^*$. Therefore, we relax these integer constraints to continuous nonnegative constraints to make (4) tractable as follows:

$$\begin{aligned}&\min_{\mathbf{S}^{(1)},\cdots,\mathbf{S}^{(q)},\mathbf{S}^*}\ \sum_{r=1}^{q}\mathrm{tr}\big(\mathbf{S}^{(r)\top}\mathbf{M}^{(r)-1}\mathbf{S}^{(r)}\big)\\&+\beta_0\sum_{r,p=1}^{q}Q_{rp}\left\|\mathbf{S}^{(r)}-\mathbf{S}^{(p)}\right\|^2+\beta_1\sum_{r=1}^{q}\left\|\mathbf{S}^{(r)}-\mathbf{S}^*\right\|^2,\quad(5)\\&\text{s.t. }\ \mathbf{S}^*\geq\mathbf{O}_{b\times s},\ \mathbf{S}^{*\top}\mathbf{S}^*=\mathbf{I}_{s\times s},\\&\quad\mathbf{S}^{(r)}\geq\mathbf{O}_{b\times s},\ \mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}_{s\times s},\ \text{for } r=1,\cdots,q\end{aligned}$$

where $\mathbf{O}$ denotes the all-zero matrix. We adopt alternating minimization to sequentially update $\mathbf{S}^{(1)},\cdots,\mathbf{S}^{(q)},\mathbf{S}^*$ with the other variables fixed, and find a local solution to (5).

Updating $\mathbf{S}^{(r)}$. To update $\mathbf{S}^{(r)}$ where r takes a value from $1,\cdots,q$, we fix $\mathbf{S}^{(r')}$ ($r'\neq r$) and $\mathbf{S}^*$, and solve the following $\mathbf{S}^{(r)}$-subproblem:

$$\begin{aligned}&\min_{\mathbf{S}^{(r)}}\ \mathrm{tr}\big(\mathbf{S}^{(r)\top}\mathbf{M}^{(r)-1}\mathbf{S}^{(r)}\big)\\&\qquad+\beta_0\sum_{p=1}^{q}Q_{rp}\left\|\mathbf{S}^{(r)}-\mathbf{S}^{(p)}\right\|^2+\beta_1\left\|\mathbf{S}^{(r)}-\mathbf{S}^*\right\|^2.\quad(6)\\&\text{s.t.}\quad\mathbf{S}^{(r)}\geq\mathbf{O}_{b\times s},\ \mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}_{s\times s}\end{aligned}$$

It should be noted that (6) is a nonconvex optimization problem because of the orthogonality constraint. The feasible region falls on the Stiefel manifold, which is the set of all $m_1\times m_2$ matrices satisfying the orthogonality constraint, *i.e.* $St(m_1,m_2)=\{\mathbf{X}\in\mathbb{R}^{m_1\times m_2}:\mathbf{X}^\top\mathbf{X}=\mathbf{I}_{m_2\times m_2}\}$. Consequently, we adopt the Partial Augmented Lagrangian Multiplier (PALM) method (Bertsekas 2014) to solve the problem (6). Only the nonnegative constraint is incorporated into the objective function of the augmented Lagrangian expression, while the orthogonality constraint is explicitly retained and imposed on the subproblem for updating $\mathbf{S}^{(r)}$. By doing this, $\mathbf{S}^{(r)}$ is updated on the Stiefel manifold, which can be effectively accomplished by the curvilinear search method (Wen and Yin 2013). Therefore, the partial augmented Lagrangian function of problem (6) is

$$\begin{aligned}&L\left(\mathbf{S}^{(r)},\mathbf{\Lambda}^{(r)},\mathbf{T}^{(r)},\sigma_r\right)\\&=\mathrm{tr}\big(\mathbf{S}^{(r)\top}\mathbf{M}^{(r)-1}\mathbf{S}^{(r)}\big)+\beta_0\sum_{p=1}^{q}Q_{rp}\left\|\mathbf{S}^{(r)}-\mathbf{S}^{(p)}\right\|^2,\\&+\beta_1\left\|\mathbf{S}^{(r)}-\mathbf{S}^*\right\|^2+\mathrm{tr}\Big(\mathbf{\Lambda}^{(r)\top}\big(\mathbf{S}^{(r)}-\mathbf{T}^{(r)}\big)\Big)+\frac{\sigma_r}{2}\left\|\mathbf{S}^{(r)}-\mathbf{T}^{(r)}\right\|^2\end{aligned}\quad(7)$$

where $\mathbf{\Lambda}^{(r)}\in\mathbb{R}^{b\times s}$ is the Lagrangian multiplier, $\mathbf{T}^{(r)}\in\mathbb{R}^{b\times s}$ is an auxiliary nonnegative matrix, and $\sigma_r>0$ is the penalty coefficient. Therefore, $\mathbf{S}^{(r)}$ is updated by minimizing (7) subject to $\mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}_{s\times s}$ via the curvilinear search method (Wen and Yin 2013) (see Algorithm 1).

In Algorithm 1, $\nabla L(\mathbf{S}^{(r)})$ is the gradient of $L(\mathbf{S}^{(r)},\mathbf{\Lambda}^{(r)},\mathbf{T}^{(r)},\sigma_r)$ w.r.t. $\mathbf{S}^{(r)}$, and $L'\big(\bar{\mathbf{P}}(\tau)\big)=\mathrm{tr}\big(\nabla L(\mathbf{S}^{(r)})^\top\bar{\mathbf{P}}'(\tau)\big)$ calculates the derivate of $L(\mathbf{S}^{(r)},\mathbf{\Lambda}^{(r)},\mathbf{T}^{(r)},\sigma_r)$ w.r.t. the stepsize τ, in which $\bar{\mathbf{P}}'(\tau)=-\big(\mathbf{I}+\frac{\tau}{2}\mathbf{A}\big)^{-1}\mathbf{A}\left(\frac{\mathbf{S}^{(r)}+\bar{\mathbf{P}}(\tau)}{2}\right)$. Algorithm 1 works by finding the gradient of L in the tangent plane of the manifold at the point $\mathbf{S}^{(r)(iter)}$ (Line 4), based on which a curve is obtained on the manifold that proceeds along the projected negative gradient (Line 7). A curvilinear search is then made along the curve towards the optimal $\mathbf{S}^{(r)(iter+1)}$.

The core of Algorithm 1 for preserving the orthogonality constraint lies in the skew-symmetric matrix $\mathbf{A}$-based Cayley transformation $\left(\mathbf{I}+\frac{\tau}{2}\mathbf{A}\right)^{-1}\left(\mathbf{I}-\frac{\tau}{2}\mathbf{A}\right)$, which projects $\mathbf{S}^{(r)}$ to $\bar{\mathbf{P}}(\tau)$ to guarantee that $\bar{\mathbf{P}}(\tau)^\top\bar{\mathbf{P}}(\tau)=\mathbf{I}$ always holds. In each iteration, the optimal stepsize τ is estimated by the Barzilai-Borwein method (Fletcher 2005).

The auxiliary matrix $\mathbf{T}^{(r)}$ in (7) is to force $\mathbf{S}^{(r)}$ to be nonnegative, which is updated by the conventional rule in the augmented Lagrangian method, that is, $\mathbf{T}_{ij}^{(r)} := \max(0,\ \mathbf{S}_{ij}^{(r)}+\mathbf{\Lambda}_{ij}^{(r)}/\sigma_r)$.

The entire PALM algorithm for solving the $\mathbf{S}^{(r)}$-subproblem (6) is outlined in Algorithm 2, which is guaranteed to converge (Wen and Yin 2013).

Updating $\mathbf{S}^*$. The $\mathbf{S}^*$-subproblem is formulated as

$$\begin{aligned}\min_{\mathbf{S}^*}\ &\sum\nolimits_{r=1}^{q}\left\|\mathbf{S}^{(r)}-\mathbf{S}^*\right\|^2\\ \text{s.t. }&\mathbf{S}^*\geq\mathbf{O}_{b\times s},\ \mathbf{S}^{*\top}\mathbf{S}^*=\mathbf{I}_{s\times s}\end{aligned}. \tag{8}$$

We also use PALM to solve the $\mathbf{S}^*$-subproblem, which is the same as the updating of $\mathbf{S}^{(r)}$, therefore we omit the detailed explanation for updating $\mathbf{S}^*$ due to space limitations.

By alternately solving the $\mathbf{S}^{(r)}$-subproblem and the $\mathbf{S}^*$-subproblem, the objective value of (5) always decreases. This objective function is lower bounded by 0 since the diagonal matrices $\mathbf{M}^{(r)-1}$ $(r=1,\cdots,q)$ are positive definite. Therefore, the entire alternating minimization process is guaranteed to converge, and the overall simplest examples (*i.e.* a curriculum) agreed by all q teachers is suggested by $\mathbf{S}^*$. Based on $\mathbf{S}^*$, the solution of problem (4) can be obtained by discretizing the continuous $\mathbf{S}^*$ into binary values. Specifically, we find the largest element in $\mathbf{S}^*$, and record its row and column; then from the unrecorded columns and rows we search the largest element and mark it again. This procedure is repeated until the s largest elements have been found. The rows of these s elements indicate the selected examples for the current propagation.

Multi-label Propagation. By employing the normalized graph Laplacian $\mathbf{H}=\mathbf{I}-\mathbf{D}^{-1}\mathbf{W}$, and starting from $\mathbf{F}_{\mathcal{L}}:=\mathbf{Y}_{\mathcal{L}}$, Kong et al. (Kong, Ng, and Zhou 2013) suggest that the label scores $\mathbf{F}_{\mathcal{S}^*,r}$ of the curriculum examples in $\mathcal{S}^*$ on the r-th $(r=1,\cdots,q)$ label can be obtained by solving

$$\mathbf{H}_{\mathcal{S}^*,\mathcal{S}^*}\mathbf{F}_{\mathcal{S}^*,r}=-\mathbf{H}_{\mathcal{S}^*,\mathcal{L}}\mathbf{F}_{\mathcal{L},r}, \tag{9}$$

where $\mathbf{H}_{\mathcal{S}^*,\mathcal{S}^*}$ and $\mathbf{H}_{\mathcal{S}^*,\mathcal{L}}$ are sub-matrices of $\mathbf{H}$ indexed by the corresponding subscripts, and $\mathbf{F}_{\mathcal{S}^*,r}$, $\mathbf{F}_{\mathcal{L},r}$ are the r-th column vectors of the label score matrices $\mathbf{F}_{\mathcal{S}^*}$ and $\mathbf{F}_{\mathcal{L}}$, respectively. Since the (i,r)-th element in $\mathbf{F}_{\mathcal{S}^*}$ conveys the possibility of $\mathbf{x}_i\in\mathcal{S}^*$ having the r-th label, (Kong, Ng, and Zhou 2013) propose setting $\mathbf{x}_i$'s label vector $\mathbf{Y}_i$ as $\mathbf{Y}_{ir}=1$ if $\mathbf{F}_{ir}$ is among the θ_i largest elements in $\mathbf{F}_i$, and $\mathbf{Y}_{ir'}=0$ otherwise. By assuming that similar examples should have a similar number of labels, they present a linear equation to find the suitable $\mathbf{\Theta}_{\mathcal{S}^*}=(\theta_1,\cdots,\theta_s)^\top$, which is

$$\mathbf{H}_{\mathcal{S}^*,\mathcal{S}^*}\mathbf{\Theta}_{\mathcal{S}^*}=-\mathbf{H}_{\mathcal{S}^*,\mathcal{L}}\mathbf{\Theta}_{\mathcal{L}}, \tag{10}$$

where $\mathbf{\Theta}_{\mathcal{L}}$ is a column vector sharing a similar definition with $\mathbf{\Theta}_{\mathcal{S}^*}$ but recording the available numbers of labels of labeled examples instead. The number of labels for $\mathbf{x}_i\in\mathcal{S}^*$ is then decided by rounding θ_i to the nearest integer.

Algorithm 1 The curvilinear search for minimizing (7)

Input: $\mathbf{S}^{(r)}$ that satisfies $\mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}$, $\varepsilon=10^{-5}$, $\tau=10^{-3}$, $\vartheta=0.2$, $\eta=0.85$, $h=1$, $\nu=L(\mathbf{S}^{(r)})$, $iter=0$
repeat
  // Compute the skew-symmetric matrix $\mathbf{A}$
  $\mathbf{A}=\nabla L(\mathbf{S}^{(r)})\cdot\mathbf{S}^{(r)\top}-\mathbf{S}^{(r)}\cdot\left(\nabla L(\mathbf{S}^{(r)})\right)^\top$;
  // Define search path $\bar{\mathbf{P}}(\tau)$ on the Stiefel manifold and find a suitable Barzilai-Borwein (BB) stepsize τ
  repeat
    $\bar{\mathbf{P}}(\tau)=\left(\mathbf{I}+\frac{\tau}{2}\mathbf{A}\right)^{-1}\left(\mathbf{I}-\frac{\tau}{2}\mathbf{A}\right)\mathbf{S}^{(r)}$;
    $\tau:=\vartheta\cdot\tau$;
    // Check BB condition
  until $L(\bar{\mathbf{P}}(\tau))\leq\nu-\tau L'(\bar{\mathbf{P}}(0))$
  // Update variables
  $\mathbf{S}^{(r)}:=\bar{\mathbf{P}}(\tau)$;
  $Q:=\eta h+1$; $\nu:=\left(\eta h\nu+L(\mathbf{S}^{(r)})\right)/h$; $iter:=iter+1$;
until $\left\|\nabla L(\mathbf{S}^{(r)})\right\|<\varepsilon$
Output: $\mathbf{S}^{(r)}$ that minimizes (7)

Algorithm 2 PALM for solving $\mathbf{S}^{(r)}$-subproblem (6)

Input: $\mathbf{M}^{(r)}$, arbitrary initial $\mathbf{S}^{(r)}$ satisfying $\mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}$, all-one matrix $\mathbf{\Lambda}^{(r)}$, β_0, β_1, $\sigma_r=1$, $\rho=1.2$, $iter=0$
repeat
  // Update $\mathbf{T}^{(r)}$
  $\mathbf{T}_{ij}^{(r)}=\max(0,\ \mathbf{S}_{ij}^{(r)}+\mathbf{\Lambda}_{ij}^{(r)}/\sigma_r)$;
  // Update $\mathbf{S}^{(r)}$ by minimizing Eq. (7) using Algorithm 1
  $\mathbf{S}^{(r)}:=\arg\min_{\mathbf{S}^{(r)\top}\mathbf{S}^{(r)}=\mathbf{I}_{s\times s}}L\left(\mathbf{S}^{(r)},\mathbf{\Lambda}^{(r)},\mathbf{T}^{(r)},\sigma_r\right)$;
  // Update variables
  $\mathbf{\Lambda}_{ij}^{(r)}:=\max(0,\mathbf{\Lambda}_{ij}^{(r)}+\sigma_r\mathbf{S}_{ij}^{(r)})$; $\sigma_r:=\min(\rho\sigma_r,10^{10})$; $iter:=iter+1$;
until Convergence
Output: $\mathbf{S}^{(r)}$ that minimizes (6)

Learning-to-teach Step

In the learning-to-teach step, teachers should also learn from the learner by absorbing feedback to adjust the curriculum generation in the next propagation. Specifically, if the learner's t-th learning performance is satisfactory, teachers may assign more examples to it for the $(t+1)$-th propagation. Otherwise, teachers should allocate fewer examples to the learner. However, the real labels of the curriculum examples are not available, so we define a learning confidence $\mathrm{Conf}(\mathcal{S}^*)$ to evaluate the learner's performance on $\mathcal{S}^*$. Intuitively, if $\mathbf{x}_i\in\mathcal{S}^*$ is assigned label r, namely $\mathbf{Y}_{ir}=1$, the corresponding label score $\mathbf{F}_{ir}$ should be as large as possible. This is because large $\mathbf{F}_{ir}$ indicates that assigning label r to $\mathbf{x}_i$ is very confident. Therefore, the propagation confidence on single $\mathbf{x}_i$ (*i.e.* $\mathrm{Conf}(\mathbf{x}_i)$) is defined by

$$\mathrm{Conf}(\mathbf{x}_i)=\min\nolimits_{r=1,\cdots,q,\ \mathbf{Y}_{ir}=1}\mathbf{F}_{ir}, \tag{11}$$

based on which we average the confidence over all $\mathbf{x}_i\in\mathcal{S}^*$, and define $\mathrm{Conf}(\mathcal{S}^*)$ as

$$\mathrm{Conf}(\mathcal{S}^*)=\frac{1}{s}\sum\nolimits_{i=1}^{s}\mathrm{Conf}(\mathbf{x}_i),\quad \mathbf{x}_i\in\mathcal{S}^*. \tag{12}$$

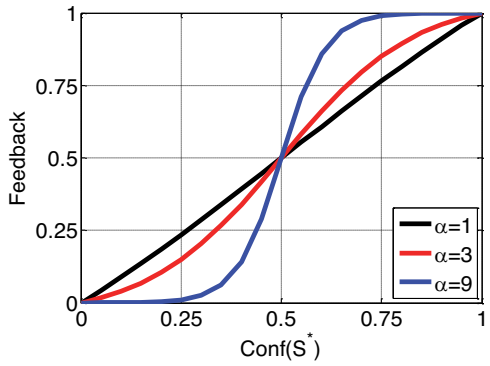

Figure 2: The curve of the learning feedback defined in (13) with different choices of α. Large α leads to a steep curve.

Based on $\mathrm{Conf}(\mathcal{S}^*)$, we utilize a sigmoid function to map the $\mathrm{Conf}(\mathcal{S}^*)$ to a nonlinear learning feedback, which is

$$\mathrm{Feedback} = \frac{\frac{1}{1+\exp(-2\alpha(\mathrm{Conf}(\mathcal{S}^*)-0.5))} - \frac{1}{1+\exp(\alpha)}}{\frac{1}{1+\exp(-\alpha)} + \frac{1}{1+\exp(\alpha)}}. \quad (13)$$

Eq. (13) has a number of ideal properties, such as being monotonically increasing, and $\mathrm{Feedback} = 0$, 0.5 and 1 when $\mathrm{Conf}(\mathcal{S}^*) = 0$, 0.5 and 1, respectively. The parameter α regulates the "steepness" of the curve of (13), and is set to 3 throughout this paper (see Fig.2).

By utilizing the designed feedback (13), the number of simplest examples selected for the next propagation is $s^{(t+1)} = \lceil b^{(t+1)} \times \mathrm{Feedback} \rceil$ where $b^{(t+1)}$ is the size of candidate set $\mathcal{B}$ in the $(t+1)$-th propagation, and "$\lceil \cdot \rceil$" rounds up the inside element to the nearest integer.

Experimental Results

This section first validates several critical steps in the proposed ML-TLLT, and then compares ML-TLLT with seven state-of-the-art methods on five benchmark datasets. Six evaluation metrics for MLL are adopted, including ranking loss, average precision, hamming loss, one error, coverage, and Micro F1; their definitions can be found in (Zhang and Zhou 2014). All the adopted datasets come from the MULAN[1] repository. The reported results of various algorithms on all the datasets are produced by 5-fold cross validation.

Algorithm Validation

Three critical factors help to boost the performance of ML-TLLT: 1) the simple-to-difficult propagation sequence generated by the teaching-to-learn step; 2) the label correlation term developed in (4); and 3) the feedback (13) designed in the learning-to-teach step. To verify their benefits, we first replace the teaching-to-learn step by randomly selecting the curriculum examples (termed "Random") to highlight the importance of 1), and then remove the label correlation term from (4) (termed "NoCorr") to demonstrate the contribution of 2), and lastly set the feedback (13) to a constant value 0.5 (denoted "NoFB") to show the effect of 3). The results on *Emotions* dataset presented in Table 1 clearly indicate that the performances on all the metrics decrease without any of the above three critical factors, therefore they are indispensable to ML-TLLT for achieving the improved results.

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

Table 1: The validation of key steps in our ML-TLLT model. "↑ (↓)" denotes the larger (smaller), the better for the corresponding metric. The best records are marked in bold.

	Random	NoCorr	NoFB	ML-TLLT
Ranking loss ↓	0.260±0.020	0.260±0.022	0.261±0.020	**0.255±0.019**
Average precision ↑	0.712±0.024	0.713±0.024	0.713±0.023	**0.717±0.023**
Hamming loss ↓	0.292±0.021	0.291±0.025	0.290±0.019	**0.289±0.022**
One error ↓	0.388±0.538	0.385±0.052	0.385±0.048	**0.383±0.051**
Coverage ↓	2.185±0.128	2.194±0.136	2.199±0.119	**2.172±0.109**
Micro F1 ↑	0.536±0.030	0.537±0.035	0.539±0.025	**0.540±0.031**

Comparison With Existing Methods

The employed baselines include Multi-Label KNN ["MLKNN", (Zhang and Zhou 2007)], Multi-Label SVM (Elisseeff and Weston 2001) with linear kernel ["MLSVM (Linear)"] and RBF Kernel ["MLSVM (RBF)"], Multi-Label learning on Tensor Product Graph ["MLTPG", (Jiang 2012)], Semi-supervised Multi-label learning via Sylvester Equation (Chen et al. 2008) inherited from Harmonic Functions ("SMSE-HF") and Local and Global Consistency ("SMSE-LGC"), and TRAM (Kong, Ng, and Zhou 2013) which acts as the "learner" in our algorithm.

Five datasets *Emotions*, *Yeast*, *Scene*, *Corel5K* and *Bibtex* in MULAN are leveraged to test the performance of all the methods. For fair comparison, we build the same graph for MLKNN, SMSE-HF, SMSE-LGC, TRAM and ML-TLLT on every dataset, and the number of neighbors K is set to $10, 10, 10, 35, 35$ on *Emotions*, *Yeast*, *Scene*, *Corel5K* and *Bibtex*, respectively. In ML-TLLT, the trade-off parameters β_0 and β_1 are set to 1 for all the experiments. As suggested by (Chen et al. 2008), we set $u = 1$, $v = 0.15$ in SMSE-HF, and $\beta = \gamma = 1$ in SMSE-LGC. The weighting parameter C in MLSVM (Linear) and MLSVM (RBF) is tuned to 1.

The results are shown in Table 2. We do not conduct MLSVM on *Corel5K* and *Bibtex* because MLSVM is not scalable to these two datasets. The hamming loss and Micro F1 of SMSE are also not reported because (Chen et al. 2008) do not provide an explicit solution for deciding the number of assigned labels. Table 2 suggests that ML-TLLT generally outperforms other baselines on all the datasets. Specifically, it can be observed that ML-TLLT performs better than the plain "learner" TRAM in almost all situations, therefore the effectiveness of our TLLT strategy is demonstrated.

Conclusion

This paper proposes a novel framework for multi-label propagation, termed "Multi-Label Teaching-to-Learn and Learning-to-Teach" (ML-TLLT). As a result of the interactions between teachers and learner, all the unlabeled examples are elaborately propagated from simple to difficult. They are consequently assigned trustable and accurate labels, leading to the superior performance of ML-TLLT over existing state-of-the-art methods.

Table 2: Experimental results of the compared methods on benchmark datasets. "↑ (↓)" denotes the larger (smaller), the better. The best records are marked in bold. "$\surd$ ($\times$)" indicates that TLLT is significantly better (worse) than the corresponding method.

	Emotions	*Yeast*	*Scene*	*Corel5K*	*Bibtex*
			Ranking loss ↓		
MLSVM (Linear)	0.294 ± 0.035 $\surd$	0.167 ± 0.005	0.080 ± 0.005 $\surd$	-	-
MLSVM (RBF)	0.415 ± 0.023 $\surd$	0.195 ± 0.007 $\surd$	0.302 ± 0.020 $\surd$	-	-
MLKNN	0.262 ± 0.016	0.170 ± 0.006	0.076 ± 0.009	0.278 ± 0.003	0.137 ± 0.003 $\surd$
MLTPG	0.439 ± 0.037 $\surd$	0.239 ± 0.002 $\surd$	0.116 ± 0.008 $\surd$	0.335 ± 0.004 $\surd$	0.192 ± 0.003 $\surd$
SMSE-HF	0.262 ± 0.015	$\mathbf{0.166 \pm 0.007}$	0.080 ± 0.004 $\surd$	0.265 ± 0.004 $\times$	0.127 ± 0.004
SMSE-LGC	0.273 ± 0.026 $\surd$	0.180 ± 0.007 $\surd$	0.080 ± 0.007 $\surd$	$\mathbf{0.251 \pm 0.003}$ $\times$	0.139 ± 0.003 $\surd$
TRAM	0.263 ± 0.022 $\surd$	0.178 ± 0.008 $\surd$	0.080 ± 0.006 $\surd$	0.297 ± 0.009 $\surd$	0.133 ± 0.003 $\surd$
ML-TLLT	$\mathbf{0.255 \pm 0.019}$	0.169 ± 0.006	$\mathbf{0.073 \pm 0.007}$	0.271 ± 0.004	$\mathbf{0.126 \pm 0.003}$
			Average precision ↑		
	Emotions	*Yeast*	*Scene*	*Corel5K*	*Bibtex*
MLSVM (Linear)	0.678 ± 0.023 $\surd$	0.761 ± 0.005	0.852 ± 0.005 $\surd$	-	-
MLSVM (RBF)	0.573 ± 0.012 $\surd$	0.713 ± 0.006 $\surd$	0.606 ± 0.016 $\surd$	-	-
MLKNN	0.702 ± 0.021 $\surd$	0.762 ± 0.008	$\mathbf{0.868 \pm 0.013}$ $\times$	0.111 ± 0.005 $\surd$	$\mathbf{0.583 \pm 0.003}$ $\times$
MLTPG	0.580 ± 0.034 $\surd$	0.683 ± 0.006 $\surd$	0.831 ± 0.009 $\surd$	0.073 ± 0.001 $\surd$	0.529 ± 0.003 $\surd$
SMSE-HF	0.711 ± 0.019	$\mathbf{0.765 \pm 0.006}$	0.864 ± 0.007 $\times$	0.117 ± 0.004 $\surd$	0.576 ± 0.004
SMSE-LGC	0.707 ± 0.034 $\surd$	0.746 ± 0.005 $\surd$	0.861 ± 0.011	0.119 ± 0.003 $\surd$	0.571 ± 0.003
TRAM	0.700 ± 0.025 $\surd$	0.752 ± 0.013 $\surd$	0.858 ± 0.012	0.115 ± 0.004 $\surd$	0.561 ± 0.006 $\surd$
ML-TLLT	$\mathbf{0.717 \pm 0.023}$	$\mathbf{0.765 \pm 0.005}$	0.859 ± 0.012	$\mathbf{0.123 \pm 0.002}$	0.575 ± 0.005
			Hamming loss ↓		
	Emotions	*Yeast*	*Scene*	*Corel5K*	*Bibtex*
MLSVM (Linear)	0.289 ± 0.022	0.202 ± 0.005	0.131 ± 0.005 $\surd$	-	-
MLSVM (RBF)	0.330 ± 0.013 $\surd$	0.227 ± 0.002 $\surd$	0.171 ± 0.002 $\surd$	-	-
MLKNN	$\mathbf{0.264 \pm 0.004}$ $\times$	$\mathbf{0.194 \pm 0.003}$ $\times$	0.087 ± 0.006	$\mathbf{0.016 \pm 0.000}$ $\times$	$\mathbf{0.033 \pm 0.000}$
MLTPG	0.333 ± 0.034 $\surd$	0.300 ± 0.005 $\surd$	0.124 ± 0.006 $\surd$	$\mathbf{0.016 \pm 0.000}$ $\times$	0.044 ± 0.001 $\surd$
SMSE-HF	-	-	-	-	-
SMSE-LGC	-	-	-	-	-
TRAM	0.306 ± 0.018 $\surd$	0.211 ± 0.009 $\surd$	0.092 ± 0.007	0.030 ± 0.000	0.037 ± 0.001 $\surd$
ML-TLLT	0.289 ± 0.022	0.204 ± 0.006	$\mathbf{0.086 \pm 0.007}$	0.029 ± 0.001	$\mathbf{0.033 \pm 0.001}$
			One error ↓		
	Emotions	*Yeast*	*Scene*	*Corel5K*	*Bibtex*
MLSVM (Linear)	0.481 ± 0.041 $\surd$	0.229 ± 0.013	0.253 ± 0.004 $\surd$	-	-
MLSVM (RBF)	0.555 ± 0.066 $\surd$	0.249 ± 0.018 $\surd$	0.596 ± 0.019 $\surd$	-	-
MLKNN	0.407 ± 0.050 $\surd$	0.236 ± 0.019 $\surd$	0.222 ± 0.020	0.870 ± 0.018 $\surd$	0.085 ± 0.003
MLTPG	0.553 ± 0.068 $\surd$	0.249 ± 0.018 $\surd$	0.269 ± 0.015 $\surd$	0.931 ± 0.009 $\surd$	0.088 ± 0.004 $\surd$
SMSE-HF	0.398 ± 0.038 $\surd$	0.234 ± 0.012	0.228 ± 0.013 $\surd$	0.857 ± 0.007	0.080 ± 0.006
SMSE-LGC	0.386 ± 0.064	0.251 ± 0.019 $\surd$	0.232 ± 0.019 $\surd$	0.867 ± 0.009 $\surd$	$\mathbf{0.078 \pm 0.004}$ $\times$
TRAM	0.415 ± 0.049 $\surd$	0.268 ± 0.024 $\surd$	0.239 ± 0.022 $\surd$	0.857 ± 0.006	0.104 ± 0.004 $\surd$
ML-TLLT	$\mathbf{0.383 \pm 0.051}$	$\mathbf{0.228 \pm 0.011}$	$\mathbf{0.220 \pm 0.021}$	$\mathbf{0.853 \pm 0.006}$	0.083 ± 0.003
			Coverage ↓		
	Emotions	*Yeast*	*Scene*	*Corel5K*	*Bibtex*
MLSVM (Linear)	2.284 ± 0.202 $\surd$	6.306 ± 0.112 $\surd$	0.448 ± 0.058 $\surd$	-	-
MLSVM (RBF)	2.970 ± 0.283 $\surd$	6.608 ± 0.136 $\surd$	1.577 ± 0.099 $\surd$	-	-
MLKNN	2.266 ± 0.113 $\surd$	6.340 ± 0.137 $\surd$	0.448 ± 0.058 $\surd$	226.031 ± 2.620 $\surd$	74.259 ± 1.434 $\surd$
MLTPG	3.102 ± 0.113 $\surd$	8.072 ± 0.037 $\surd$	0.647 ± 0.044 $\surd$	246.253 ± 1.359 $\surd$	89.830 ± 1.196 $\surd$
SMSE-HF	2.208 ± 0.137 $\surd$	6.223 ± 0.152 $\times$	0.470 ± 0.030 $\surd$	215.497 ± 1.695	71.533 ± 1.957 $\surd$
SMSE-LGC	2.259 ± 0.142 $\surd$	$\mathbf{6.213 \pm 0.136}$ $\times$	0.464 ± 0.037 $\surd$	$\mathbf{209.655 \pm 0.956}$ $\times$	74.935 ± 1.441 $\surd$
TRAM	2.202 ± 0.151	6.341 ± 0.122 $\surd$	0.460 ± 0.027 $\surd$	226.254 ± 2.552 $\surd$	70.123 ± 1.840 $\surd$
ML-TLLT	$\mathbf{2.172 \pm 0.109}$	6.265 ± 0.139	$\mathbf{0.423 \pm 0.044}$	217.511 ± 2.032	$\mathbf{68.539 \pm 1.298}$
			Micro F1 ↑		
	Emotions	*Yeast*	*Scene*	*Corel5K*	*Bibtex*
MLSVM (Linear)	0.510 ± 0.047 $\surd$	0.651 ± 0.008 $\surd$	0.632 ± 0.019 $\surd$	-	-
MLSVM (RBF)	0.310 ± 0.039 $\surd$	0.596 ± 0.006 $\surd$	0.591 ± 0.013 $\surd$	-	-
MLKNN	0.453 ± 0.035 $\surd$	0.642 ± 0.010 $\surd$	0.734 ± 0.020 $\surd$	0.004 ± 0.001 $\surd$	0.460 ± 0.005 $\surd$
MLTPG	0.368 ± 0.026 $\surd$	0.520 ± 0.018 $\surd$	0.645 ± 0.015 $\surd$	0.002 ± 0.000 $\surd$	$\mathbf{0.482 \pm 0.007}$ $\times$
SMSE-HF	-	-	-	-	-
SMSE-LGC	-	-	-	-	-
TRAM	0.513 ± 0.023 $\surd$	0.650 ± 0.013 $\surd$	0.735 ± 0.021 $\surd$	0.099 ± 0.003 $\surd$	0.445 ± 0.007 $\surd$
ML-TLLT	$\mathbf{0.540 \pm 0.031}$	$\mathbf{0.663 \pm 0.007}$	$\mathbf{0.753 \pm 0.021}$	$\mathbf{0.107 \pm 0.003}$	0.476 ± 0.004

Acknowledgments

This research is supported by NSFC, China (No: 61572315); 973 Plan, China (No: 2015CB856004); and Australian Research Council Discovery Project (No: DP-140102164 & No: FT-130101457).

References

Bengio, Y.; Louradour, J.; Collobert, R.; and Weston, J. 2009. Curriculum learning. In *ICML*, 41–48. ACM.

Bertsekas, D. 2014. *Constrained optimization and Lagrange multiplier methods*. Academic Press.

Bi, W., and Kwok, J. 2011. Multi-label classification on tree-and DAG-structured hierarchies. In *ICML*, 17–24.

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

Chen, G.; Song, Y.; Wang, F.; and Zhang, C. 2008. Semi-supervised multi-label learning by solving a Sylvester equation. In *SDM*, 410–419. SIAM.

Chen, X.; Mu, Y.; Liu, H.; Yan, S.; Rui, Y.; and Chua, T. 2013. Large-scale multilabel propagation based on efficient sparse graph construction. *ACM Transactions on Multimedia Computing, Communications, and Applications* 10(1):6.

Clare, A., and King, R. 2001. Knowledge discovery in multi-label phenotype data. In *ECML-PKDD*, 42–53. Springer.

Elisseeff, A., and Weston, J. 2001. A kernel method for multi-labelled classification. In *NIPS*, 681–687.

Fletcher, R. 2005. On the Barzilai-Borwein method. In *Optimization and Control with Applications*. Springer. 235–256.

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

Gong, C.; Tao, D.; Liu, W.; Maybank, S.; Fang, M.; Fu, K.; and Yang, J. 2015. Saliency propagation from simple to difficult. In *CVPR*, 2531–2539.

Jiang, L.; Meng, D.; Zhao, Q.; Shan, S.; and Hauptmann, A. 2015. Self-paced curriculum learning. In *AAAI*.

Jiang, J. 2012. Multi-label learning on tensor product graph. In *AAAI*.

Kang, F.; Jin, R.; and Sukthankar, R. 2006. Correlated label propagation with application to multi-label learning. In *CVPR*, volume 2, 1719–1726. IEEE.

Khan, F.; Mutlu, B.; and Zhu, X. 2011. How do humans teach: On curriculum learning and teaching dimension. In *NIPS*, 1449–1457.

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

Kumar, M.; Packer, B.; and Koller, D. 2010. Self-paced learning for latent variable models. In *NIPS*, 1189–1197.

Qiu, H., and Hancock, E. 2007. Clustering and embedding using commute times. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 29(11):1873–1890.

Schapire, R., and Singer, Y. 2000. Boostexter: A boosting-based system for text categorization. *Machine Learning* 39(2):135–168.

Settles, B. 2010. Active learning literature survey. *Technical Report 1648, University of Wisconsin, Madison* 52(55-66):11.

Tang, L., and Liu, H. 2009. Relational learning via latent social dimensions. In *KDD*, 817–826. ACM.

Wang, H.; Huang, H.; and Ding, C. 2009. Image annotation using multi-label correlated Green's function. In *ICCV*, 2029–2034. IEEE.

Wang, B.; Tu, Z.; and Tsotsos, J. 2013. Dynamic label propagation for semi-supervised multi-class multi-label classification. In *ICCV*, 425–432. IEEE.

Wen, Z., and Yin, W. 2013. A feasible method for optimization with orthogonality constraints. *Mathematical Programming* 142(1-2):397–434.

Xu, M.; Li, Y.; and Zhou, Z. 2013. Multi-label learning with pro loss. In *AAAI*.

Xu, C.; Tao, D.; and Xu, C. 2015. Large-margin multi-label causal feature learning. In *AAAI*.

Zha, Z.; Mei, T.; Wang, J.; Wang, Z.; and Hua, X. 2008. Graph-based semi-supervised learning with multi-label. In *ICME*. IEEE.

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. *Knowledge and Data Engineering, IEEE Transactions on* 26(8):1819–1837.