Discriminative Multi-label Model Reuse for Multi-label Learning

Yi Zhang, Zhecheng Zhang, Yinlong Zhu, Lei Zhang, and Chongjun Wang(✉)

National Key Laboratory for Novel Software Technology at Nanjing University, Department of Computer Science and Technology,Nanjing University, Nanjing, China
{njuzhangy,zzc,zhuyinlong}@smail.nju.edu.cn,
{zhangl,chjwang}@nju.edu.cn

Abstract. Traditional Chinese Medicine (TCM) with diagnosis scales is a holistic way for diagnosing Parkinson's Disease, where symptoms can be represented as multiple labels. To solve this problem, multi-label learning provides a framework for handling such task and has exhibited excellent performance. Besides, it is a challenging issue of how to effectively utilize label correlations in multi-label learning. In this paper, we propose a novel algorithm named Discriminative Multi-label Model Reuse (DMLMR) for multi-label learning, which exploits label correlations with model reuse, instance distribution adaptation and label distribution adaptation. Experiments on real-world dataset of Parkinson's disease demonstrate the superiority of DMLMR for diagnosing PD. To prove the effectiveness of the proposed DMLMR, extensive experiments on four benchmark multi-label datasets show that DMLMR significantly outperforms other state-of-the-art multi-label learning algorithms.

Keywords: Parkinson's disease · Multi-label learning · Label correlations · Model reuse · Distribution adaptation

1 Introduction

Tradition Chinese Medicine (TCM) is a new way for PD [13]. For one thing, *TCM scales* includes tongue phase as well as four traditional methods of diagnosis: observation, listening, interrogation and pulse-taking. For another, *syndrome types* of PD in TCM can be divided into following 5 categories: (1) stirring wind due to phlegma-heat, (2) stirring wind due to blood heat, (3) deficiency of both qi and blood, (4) insufficiency of the liver and kidney, (5) deficiency of both yin and yang. Moreover, each TCM syndrome type can be subdivided into primary and secondary *syndrome types*.

TCM scholars are supposed to collect disease information of patients, and categorize a patient into one or more *syndrome types* based on TCM theory and rich experience. This diagnostic process requires doctors equipped with extensive experience of *Syndrome Differentiation* at the time of treatment. Due to

X. Wang et al. (Eds.): APWeb-WAIM 2020, LNCS 12317, pp. 725–739, 2020.
https://doi.org/10.1007/978-3-030-60259-8_53

the essential characteristic of TCM, *TCM scales* appear to be overwhelmingly dependent on personal experience of doctors. The problems of diagnosing PD in TCM lie in two aspects: specialists of PD are in short supply and diagnostic levels of doctors are inconsistent. Consequently, the diagnosis of PD might be subjective, which violates the original intention of effectiveness. Therefore, it is desired to design a semi-automatic mechanism for diagnosing PD in TCM.

In this paper, we formalize the problem of diagnosing Parkinson's disease in TCM into a multi-label learning problem, where we treat *TCM scales* as features and treat *syndrome types* as multiple labels. In multi-label learning [21], each instance can be represented by multiple labels simultaneously. For example, an image may be annotated with both sea and beach. The task of multi-label learning is to learn a classification model which can predict all the relevant labels for unseen instances. Nowadays, multi-label learning has been applied to various application scenarios, such as text classification [9], image annotation [11], video annotation [14], social networks [17], music emotion categorization [18]. In addition, the exploration of label correlations has been accepted as a key component of effective multi-label learning approaches [6,23].

The main contributions of this paper include:

- Real-world Parkinson's disease diagnosis in Traditional Chines Medicine is investigated and assessed.
- We formalize the problem of diagnosing PD in TCM as a multi-label learning problem, by treating *TCM scales* as features while treating *syndrome types* as multiple labels. Meanwhile, we apply multi-label classification technology to diagnose PD in TCM.
- We propose a novel Discriminative Multi-label Model Reuse (DMLMR) algorithm to deal with multi-label learning problem, which perform excellently in handling diagnosis of Parkinson's disease in TCM. Extensive experiments on four benchmark multi-label datasets show that DMLMR algorithm significantly outperforms the state-of-the-art multi-label learning algorithms.

The remainder of the paper is organized as follows. Section 2 briefly reviews some related work of multi-label learning. Section 3 presents formulation of the problem and our proposed DMLMR algorithm. Section 4 reports the experimental results, followed by the conclusion in Sect. 5.

2 Related Work

Generally, multi-label learning algorithms can be categorized into following three strategies based on the order of label correlations considered by the system.

First-order strategy copes with multi-label learning problem in a label-by-label manner. Binary Relevance (BR) [1] takes each label independently and decomposes it into multiple binary classification tasks. However, BR neglects the relationship among labels.

Second-order strategy introduces pairwise relations among multiple labels, such as the ranking between the relevant and irrelevant labels [5]. Calibrated

Label Ranking (CLR) [4] firstly transforms the multi-label learning problem into label ranking problem by introducing the pairwise comparison. Recently, LLSF [7] performs joint label-specific feature selection and take the label correlation matrix as prior knowledge.

High-order strategy builds more complex relations among labels for multi-label learning. Classifier Chain (CC) [15] transforms the multi-label classification problem into a chain of binary classification problems, where the quality is dependent on the label order in the chain. Ensemble Classifier Chains (ECC) [16] constructs multiple CCs by using different random label orders. Multi-modal Classifier Chains (MCC) [22] release the reliance of label order by combining predicted labels as a new modality. Multi-label k-nearest neighbour (MLkNN) [20] builds a Bayesian model by using the k-nearest neighbour method to obtain the prior and likelihood. In addition, there are also some high-order approaches that exploit label correlations on the hypothesis space. For example, a boosting approach Multi-label Hypothesis Reuse (MLHR) [8] is proposed to exploit label correlations with a hypothesis reuse mechanism. Latent Semantic Aware Multi-view Multi-label Learning (LSA-MML) [19] implicitly encodes the label correlations by the common representation based on the uncovering latent semantic bases and the relations among them. Considering the potential association between paired labels, Dual-Set Multi-Label Learning (DSML) [10] exploits pairwise inter-set label relationships for assisting multi-label learning. Most of the existing approaches take label correlations as prior knowledge, which may not correctly characterize the real relationships among labels. And then, Collaboration based Multi-Label Learning (CAMEL) [3] is proposed to learn the label correlations via sparse reconstruction in the label space.

3 Methodology

This section mainly gives the detail description of Discriminative Multi-label Model Reuse (DMLMR) algorithm after a preliminary notation explanation.

3.1 Preliminaries and Problem Formulation

Before describing the problem formulation, we begin with some notations and preliminaries.

Let $\mathcal{X} = \mathbb{R}^d$ denote the d dimensional feature space, and $\mathcal{Y} = \{-1, 1\}^L$ denote the label space with L labels.

Given the training dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$ with N instances, the task of multi-label learning is to learn a mapping function $\boldsymbol{H} : \mathcal{X} \to \mathcal{Y}$, which maps from feature space to label space. The i-th instance $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ contains a feature vector $\boldsymbol{x}_i = [x_1, x_2, \ldots, x_d] \in \mathcal{X}$ and a label vector $\boldsymbol{y}_i = [y^1, y^2, \cdots, y^L] \in \mathcal{Y}$, where $y^k = 1$ indicating $\boldsymbol{x}_i$ is associated with the k-th label, $y^k = -1$ otherwise. $\mathcal{T} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^M$ denotes testing dataset. In addition, $\boldsymbol{H}(\cdot) = [H^1(\cdot), H^2(\cdot), \ldots, H^L(\cdot)]$ can be used to predict labels for unseen instances in $\mathcal{T}$, where $H^k(\cdot)$ denotes the classifier of the k-th label.

For simplicity, we denote $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N]^T \in \mathbb{R}^{N \times d}$ as the instance matrix, and $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_N]^T \in \mathbb{R}^{N \times L}$ as the label matrix. The original training dataset can be alternatively represented by $\mathcal{D} = \{(\boldsymbol{X}, \boldsymbol{Y})\}$.

With analysis in Sect. 1, the problem of diagnosing Parkinson's disease can be modeled as multi-label learning problem.

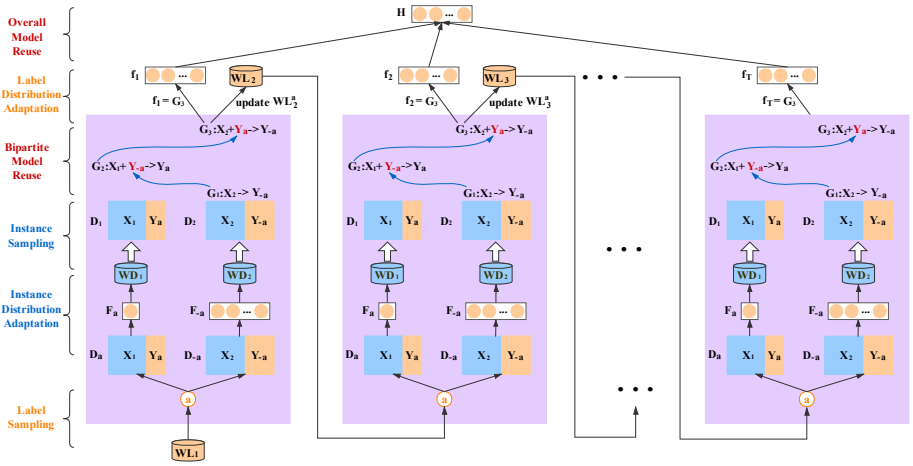


Fig. 1. The overall flowchart of DMLMR algorithm. Cylinder shadowed with orange denotes label distribution, while cylinder shadowed with blue denotes instance distribution.

3.2 Discriminative Multi-label Model Reuse

In this subsection, we introduce Discriminative Multi-label Model Reuse (DMLMR) algorithm in detail. The pseudo code of DMLMR is presented in Algorithm 1.

At first, we train on the original dataset $\mathcal{D}$ with a base multi-label algorithm (here we adopt CC algorithm) and get $\boldsymbol{F}(\cdot) = [F^1(\cdot), \cdots, F^k(\cdot), \cdots, F^L(\cdot)]$, where $F^k(\cdot)$ represents the original classifier for the k-th label. $\boldsymbol{\tau} = [\tau_1, \cdots, \tau_T]$ denotes chain of selected labels, where T denotes number of boosting round. DMLMR maintains one label distribution $\boldsymbol{WL}_t = [WL_t^1, \cdots, WL_t^k, \cdots, WL_t^L]$, where WL_t^k is the weight of the k-th label at t-th boosting round. Initially, $\boldsymbol{\tau} = \emptyset$ and $WL_1^k = \frac{1}{L}$, which means $\boldsymbol{WL}_1 = [\frac{1}{L}, \cdots, \frac{1}{L}]$.

Figure 1 illustrates an overview of our proposed DMLMR algorithm. At t-th boosting round, there are following 5 steps.

Label Sampling. We sample one label a according to the label distribution $\boldsymbol{WL}_t$, where $a \in \{1, 2, \cdots, L\}$. And then we update $\boldsymbol{\tau}$ by concatenating $\boldsymbol{\tau}$ and a, i.e., $\boldsymbol{\tau} = [\boldsymbol{\tau}, a]$.

Instance Distribution Adaptation. After getting one sampled label a, we transform the original dataset $\mathcal{D}$ into two dataset $\mathcal{D}_a = \{(\boldsymbol{X}, \boldsymbol{Y}_a)\}$ and $\boldsymbol{D}_{-a} = \{(\boldsymbol{X}, \boldsymbol{Y}_{-a})\}$.

Algorithm 1. The DMLMR algorithm

Input:
$\mathcal{D} = \{(\boldsymbol{X}, \boldsymbol{Y})\}$: original training dataset
λ_{intra}: intra-set reweight parameter
λ_{inter}: inter-set reweight parameter
T: number of boosting round
Output:
$\boldsymbol{H}(\cdot)$: classifiers of all labels
Initialize: $\boldsymbol{\tau} = \emptyset$, $\boldsymbol{WL}_1 = [\frac{1}{L}, \cdots, \frac{1}{L}]$
Train on $\mathcal{D}$
for $t = 1 : T$ **do**
  Sample one label a according to $\boldsymbol{WL}_t$
  Update $\boldsymbol{\tau} = [\boldsymbol{\tau}, a]$
  Compute $\boldsymbol{WD}_1$ and $\boldsymbol{WD}_2$ with Eq. 1
  Sample $\mathcal{D}_1$ from $\mathcal{D}$ according to $\boldsymbol{WD}_1$
  Sample $\mathcal{D}_2$ from $\mathcal{D}$ according to $\boldsymbol{WD}_2$
  Train $\boldsymbol{G}_1$, $\boldsymbol{G}_2$ and $\boldsymbol{G}_3$ with bipartite model reuse
  $\boldsymbol{f}_t(\cdot) = \boldsymbol{G}_3(\cdot)$
  Update $\boldsymbol{WL}_{t+1}$ with Eq. 6
end for
for $k = 1 : L$ **do**
  Compute $H^k(\cdot)$ with Eq. 7
end for
return $\boldsymbol{H}(\cdot)$

Here $\boldsymbol{Y}_a$ and $\boldsymbol{Y}_{-a}$ are label vectors associated with instance matrix $\boldsymbol{X}$, which is shown in Fig. 2. More specifically, $\boldsymbol{Y}_a \in \mathbb{R}^N$ denotes the a-th column vector of the matrix $\boldsymbol{Y}$ (versus $\boldsymbol{y}_i \in \mathbb{R}^L$ for the i-th row vector of $\boldsymbol{Y}$), and $\boldsymbol{Y}_{-a} = [\boldsymbol{Y}_1, \cdots, \boldsymbol{Y}_{a-1}, \boldsymbol{Y}_{a+1}, \cdots, \boldsymbol{Y}_L] \in \mathbb{R}^{N \times (L-1)}$ represents the matrix that excludes the a-th column vector of the matrix $\boldsymbol{Y}$.

And then we get $\boldsymbol{F}_a(\cdot)$ and $\boldsymbol{F}_{-a}(\cdot)$, where $\boldsymbol{F}_a(\cdot) = F^a(\cdot)$ denotes the original classifier of $\mathcal{Y}_a$ and $\boldsymbol{F}_{-a} = [F^1(\cdot), \cdots, F^{a-1}(\cdot), F^{a+1}(\cdot), \cdots, F^L(\cdot)]$ denotes the original classifiers of $\mathcal{Y}_{-a}$, where $\mathcal{Y}_a = \{-1, 1\}$ denotes label space of the a-th label and $\mathcal{Y}_{-a} = \{-1, 1\}^{L-1}$ denotes label space of all the labels exclude the a-th label.

In order to exploit label correlations, we maintain two instance distributions $\boldsymbol{WD}_1$ and $\boldsymbol{WD}_2$ adapted by Eq. 1, where WD_1^i and WD_2^i are the weight for the i-th instance with respect to $\mathcal{Y}_a$ and $\mathcal{Y}_{-a}$, respectively.

$$\begin{aligned} WD_1^i &= \frac{1}{N} \cdot \lambda_{intra}^{\mathbb{I}(F_a(\boldsymbol{x}_i) \neq \boldsymbol{y}_{i,a})} \cdot \lambda_{inter}^{\mathbb{I}(F_{-a}(\boldsymbol{x}_i) \neq \boldsymbol{y}_{i,-a})} \\ WD_2^i &= \frac{1}{N} \cdot \lambda_{intra}^{\mathbb{I}(F_{-a}(\boldsymbol{x}_i) \neq \boldsymbol{y}_{i,-a})} \cdot \lambda_{inter}^{\mathbb{I}(F_a(\boldsymbol{x}_i) \neq \boldsymbol{y}_{i,a})} \end{aligned} \tag{1}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function which outputs 1 if $\cdot$ is true, 0 otherwise. Additionally, $\boldsymbol{y}_{i,a}$ denotes ground truth of a-th label associated with $\boldsymbol{x}_i$ and $\boldsymbol{y}_{i,-a}$ denotes ground truth of all the labels excludes a-th label associated with

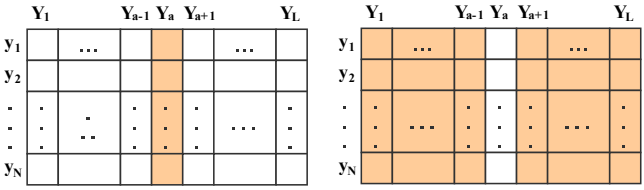


Fig. 2. Illustration of label vector $\boldsymbol{Y}_a$ and $\boldsymbol{Y}_{-a}$ in $\boldsymbol{Y}$. In the left part, matrix shadowed with orange represents $\boldsymbol{Y}_a$. In the right part, matrix shadowed with orange represents $\boldsymbol{Y}_{-a}$.

$\boldsymbol{x}_i$. λ_{intra} is the intra-set reweight parameter and λ_{inter} is the inter-set reweight parameter. Take WD_1^i as an example, item $\lambda_{intra}^{\mathbb{I}(F_a(\boldsymbol{x}_i)\neq \boldsymbol{y}_{i,a})}$ considers the mistake made by label in $\mathcal{Y}_a$, i.e, a model that has made mistake will be emphasized by assigning a higher weight. Item $\lambda_{inter}^{\mathbb{I}(F_{-a}(\boldsymbol{x}_i)\neq \boldsymbol{y}_{i,-a})}$ considers inter-set relationship between $\mathcal{Y}_a$ and $\mathcal{Y}_{-a}$, i.e., the weight of an instance on $\mathcal{Y}_a$ will be increased when misclassified on $\mathcal{Y}_{-a}$. Meaning of items in WD_2^i is similar to that in WD_1^i.

At the end of the training process, we normalize $\boldsymbol{WD}_1$ and $\boldsymbol{WD}_2$ to form a valid distribution.

Instance Sampling. We decompose the original problem into two dependent sub-problems.

And then we sample two datasets $\mathcal{D}_1 = \{(\boldsymbol{X}_1, \boldsymbol{Y}_a)\}$ and $\mathcal{D}_2 = \{(\boldsymbol{X}_2, \boldsymbol{Y}_{-a})\}$ i.i.d. according to instance distributions $\boldsymbol{WD}_1$ and $\boldsymbol{WD}_2$ respectively, where $\boldsymbol{X}_1 \in \mathbb{R}^{N\times d}$, $\boldsymbol{Y}_a \in \mathbb{R}^{N\times 1}$, $\boldsymbol{X}_2 \in \mathbb{R}^{N\times d}$, $\boldsymbol{Y}_{-a} \in \mathbb{R}^{N\times(L-1)}$.

Bipartite Model Reuse We train on two datasets $\mathcal{D}_1$ and $\mathcal{D}_2$ with model reuse and get 3 models $\boldsymbol{G}_1$, $\boldsymbol{G}_2$ and $\boldsymbol{G}_3$.

- Firstly, we train on the dataset $\mathcal{D}_2$ with basic multi-label learning algorithm (here we adopt CC algorithm), and then we get model $\boldsymbol{G}_1 : \mathcal{X} \to \mathcal{Y}_{-a}$.
- Secondly, we reuse model $\boldsymbol{G}_1$ on $\mathcal{D}_1$ and get predicted label vector $\boldsymbol{G}_1(\boldsymbol{x}_i)$. And then, we concatenate feature vector with predicted label vector, i.e, $[\boldsymbol{x}_i, \boldsymbol{G}_1(\boldsymbol{x}_i)]$. Training on dataset $\mathcal{D}_1$, we get model $\boldsymbol{G}_2 : \mathcal{X} + \mathcal{Y}_{-a} \to \mathcal{Y}_a$.
- Thirdly, we reuse model $\boldsymbol{G}_2$ on $\mathcal{D}_2$ and get predicted label vector $\boldsymbol{G}_2(\boldsymbol{x}_i)$. And then, we concatenate $\boldsymbol{x}_i$ with predicted label vector, i.e, $[\boldsymbol{x}_i, \boldsymbol{G}_2([\boldsymbol{x}_i, \boldsymbol{G}_1(\boldsymbol{x}_i)])]$. Training on dataset $\mathcal{D}_2$, we get model $\boldsymbol{G}_3 : \mathcal{X} + \mathcal{Y}_a \to \mathcal{Y}_{-a}$.

It is notable that $\boldsymbol{G}_2$ reuses model $\boldsymbol{G}_1$, so $\boldsymbol{G}_3$ reuses two models $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$. Model trained on one dataset is reused on the other dataset, which provides additional help for the final classification. Furthermore, we provide theoretical analysis for bipartite model reuse. $\boldsymbol{h}_a(\cdot) = \boldsymbol{G}_2(\cdot)$ and $\boldsymbol{h}_{-a}(\cdot) = \boldsymbol{G}_3(\cdot)$ in the following analysis.

Definition 1. *Generalization error of hypothesis* $\boldsymbol{h}(\cdot)$ *mapping from* $\mathcal{X}$ *to* $\mathcal{Y}$ *based on HammingLoss:*

$$R(\boldsymbol{h}) = \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\frac{1}{L}\sum_{k=1}^{L}\mathbb{I}(\boldsymbol{h}(\boldsymbol{x}) \neq \boldsymbol{y}^k)\Big] \tag{2}$$

where $\boldsymbol{y}^k$ is the ground-truth of the k-th label.

Definition 2. *Empirical error of hypothesis* $\boldsymbol{h}(\cdot)$*:*

$$\hat{R}(\boldsymbol{h}) = \frac{1}{m}\sum_{i=1}^{m}\Big(\frac{1}{L}\sum_{k=1}^{L}\mathbb{I}(\boldsymbol{h}(\boldsymbol{x}) \neq \boldsymbol{y}^k)\Big) \tag{3}$$

Lemma 1. $R(\boldsymbol{h}) \leq max\{R(\boldsymbol{h}_a), R(\boldsymbol{h}_{-a})\}$*, where* $\boldsymbol{h}(\cdot)$ *is composed of* $\boldsymbol{h}_a(\cdot)$ *and* $\boldsymbol{h}_{-a}(\cdot)$*.*

Proof.

$$\begin{aligned}
R(\boldsymbol{h}) &= \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\frac{1}{L}\sum_{k=1}^{L}\mathbb{I}(\boldsymbol{h}(\boldsymbol{x}) \neq \boldsymbol{y}^k)\Big] \\
&= \frac{1}{L}\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\mathbb{I}(\boldsymbol{h}_a(\boldsymbol{x}) \neq \boldsymbol{y}^a)\Big] \\
&\quad + \frac{1}{L}\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\sum_{k=1,k\neq a}^{L}\mathbb{I}(\boldsymbol{h}_{-a}(\boldsymbol{x}) \neq \boldsymbol{y}^k)\Big] \\
&= \frac{1}{L}\big(R(\boldsymbol{h}_a) + (L-1)R(\boldsymbol{h}_{-a})\big) \\
&\leq \frac{1}{}Lmax\{R(\boldsymbol{h}_a), R(\boldsymbol{h}_{-a})\}(1+L-1)\} \\
&= max\{R(\boldsymbol{h}_a), R(\boldsymbol{h}_{-a})\}
\end{aligned}$$

□

Lemma 2. $R(\boldsymbol{h}_{-a}) \leq max\{R(h^k)\}_{k=1,k\neq a}^{L}$

Proof.

$$\begin{aligned}
R(\boldsymbol{h}_{-a}) &= \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\frac{1}{L-1}\sum_{k=1,k\neq a}^{L}\mathbb{I}(h^k(\boldsymbol{x}) \neq \boldsymbol{y}^k)\Big] \\
&= \frac{1}{L-1}\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\sum_{k=1,k\neq a}^{L}\mathbb{I}(h^k(\boldsymbol{x}) \neq \boldsymbol{y}^k)\Big] \\
&= \frac{1}{L-1}\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\Big[\sum_{k=1,k\neq a}^{L}R(h^k)\Big] \\
&\leq \frac{1}{L-1}(L-1)max\{R(h^k)\}_{k=1,k\neq a}^{L} = R(\boldsymbol{h}_m)
\end{aligned}$$

□

where $\boldsymbol{h}_{-a}(\cdot) = [h^1(\cdot), \cdots, h^{a-1}(\cdot), h^{a+1}(\cdot), \cdots, h^L(\cdot)]$, and for simplicity, we denote $max\{R(h^k)\}_{k=1,k\neq a}^{L}$ as $R(\boldsymbol{h}_m)$.

Theorem 31. *In mono-label case, let $H \subset \mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ be a hypothesis set. Fix $\rho > 0$. Assume there exists $r > 0$ such that $k(\boldsymbol{x}, \boldsymbol{x}) \leq r^2$ for all $\boldsymbol{x} \in \mathcal{X}$. For any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $h \in H$. [12]*

$$R(h) \leq \hat{R}_\rho(h) + 2\sqrt{\frac{r^2 \wedge^2 / \rho^2}{m}} + 3\sqrt{\frac{log(2/\delta)}{m}} \tag{4}$$

Combine Lemma 1, Lemma 2 and Theorem 31, we have:

Proof.

$$\begin{aligned} R(\boldsymbol{h}) &\leq max\{R(\boldsymbol{h}_a), R(\boldsymbol{h}_{-a}) \\ &\leq max\Big\{\hat{R}_\rho(\boldsymbol{h}_a) + 2\sqrt{\frac{r^2 \wedge^2 / \rho^2}{m}} + 3\sqrt{\frac{log(2/\delta)}{m}}, \\ &\qquad \hat{R}_\rho(\boldsymbol{h}_m) + 2\sqrt{\frac{r^2 \wedge^2 / \rho^2}{m}} + 3\sqrt{\frac{log(2/\delta)}{m}}\Big\} \\ &\leq max\{\hat{R}_\rho(\boldsymbol{h}_a), \hat{R}_\rho(\boldsymbol{h}_m)\} + 2\sqrt{\frac{r^2 \wedge^2 / \rho^2}{m}} + 3\sqrt{\frac{log(2/\delta)}{m}} \end{aligned}$$

□

The convergence rate of generalization error is standard as $O(\frac{1}{\sqrt{m}})$, which validates the effect of bipartite model reuse.

Label Distribution Adaptation. In order to select most discriminative label for bipartite model reuse, we are supposed to adapt label distribution according to the models trained by bipartite model reuse. We get prediction $\boldsymbol{f}_t(\cdot) = \boldsymbol{G}_3(\cdot)$, and $\boldsymbol{f}_t(\cdot) = [f_t^1(\cdot), \cdots, f_t^{a-1}(\cdot), f_t^{a+1}(\cdot), \cdots, f_t^L(\cdot)]$ where $f_t^k(\cdot)$ denotes the classifier of the k-th label. And then we test on dataset $\mathcal{T}$ with $\boldsymbol{f}_t(\cdot)$ and $\boldsymbol{F}_{-a}(\cdot)$ respectively. We get importance rate of the a-th label for other labels as follows:

$$\alpha_t = \frac{SubAcc(\boldsymbol{f}_t)}{SubAcc(\boldsymbol{F}_{-a})} \tag{5}$$

where $SubsetAcc_t(\boldsymbol{f}_t) = \frac{1}{M}\sum_{i=1}^{M} \mathbb{I}\big(\boldsymbol{f}_t(\boldsymbol{x}_i) = \boldsymbol{y}_{i,-a}\big)$ and $SubsetAcc_t(\boldsymbol{F}_{-a}) = \frac{1}{M}\sum_{i=1}^{M} \mathbb{I}\big(\boldsymbol{F}_{-a}(\boldsymbol{x}_i) = \boldsymbol{y}_{i,-a}\big)$.

On the other hand, we will increase the weight of the a-th label if $\alpha_t > 1$, i.e, the a-th label has a positive effect to other labels with bipartite model reuse. The weight of other labels exclude the a-th label remain unchanged. And then we adapt label distribution $\boldsymbol{WL}_{t+1} = [WL_{t+1}^1, \cdots, WL_{t+1}^k, \cdots, WL_{t+1}^L]$ for next boosting round.

$$WL_{t+1}^k = WL_t^k \cdot \alpha_t^{\mathbb{I}(k=a)} \tag{6}$$

where $\mathbb{I}(\cdot)$ is an indicator function and $k = \{1, \cdots, L\}$. Similar to $\boldsymbol{WD}_1$ and $\boldsymbol{WD}_2$, we then normalize $\boldsymbol{WL}_{t+1}$.

Above all, ***Overall Model Reuse*** is adopted. As is shown in Fig. 1, we get $\boldsymbol{f}_1(\cdot), \boldsymbol{f}_2(\cdot), \cdots, \boldsymbol{f}_T(\cdot)$ after T number of boosting round. Finally we integrate all models together and get $\boldsymbol{H}(\cdot) = [H^1(\cdot), \cdots, H^k(\cdot), \cdots, H^L(\cdot)]$, where $H^k(\cdot)$ denotes final classifier of the k-th label. In the testing phase, labels are predicted for instance $\boldsymbol{x}$ according to:

$$H^k(\boldsymbol{x}) = \underset{l}{argmax} \sum_{t=1, k \neq \tau_t}^{T} \alpha_t \cdot \mathbb{I}(f_t^k(\boldsymbol{x}) = l) \tag{7}$$

where $l \in \{-1, 1\}$, $k = \{1, \cdots, L\}$.

4 Experiments

In this section, we validate the effectiveness of our proposed DMLMR algorithm on real-world dataset of Parkinson's disease and various benchmark multi-label datasets.

4.1 Dataset Description

Firstly, we manually collect real-world dataset of Parkinson's disease in Traditional Chinese Medicine (TCM). Furthermore, we will briefly present the feature and label generation procedure for Parkinson's disease diagnosis.

Both *Parkinson-P* and *Parkinson* have 91 *TCM scales* as features. However, *Parkinson-P* has 5 primary symptoms. *Parkinson* has 10 *syndrome types*: 5 primary *syndrome types* and 5 secondary *syndrome types*. More details with regard to *syndrome types* can be found in Sect. 1.

It is notable that DMLMR is designed for diagnosing Parkinson's disease, it is also a general multi-label learning algorithm. For comprehensive performance evaluation, we collect 4 benchmark multi-label datasets.

- *ML2000*: is an image dataset from [20], including 2000 images from 5 categories.
- *Scene*: has 2407 images and 6 possible labels [1].
- *Emotions*: is a set of 593 songs with 6 clusters of music emotions [16].
- *Genbase*: consists of 662 proteins with known structure families that belong in 27 labels [2].

Table 1 summarizes the detailed characteristics of all datasets, Given a multi-label dataset $\mathcal{D} = \{(\boldsymbol{X}, \boldsymbol{Y})\}$, we use $|\mathcal{D}|$, $dim(\mathcal{D})$, $L(\mathcal{D})$, $LCard(\mathcal{D})$, $LDen(\mathcal{D})$ and $F(\mathcal{D})$ to represent number of instances, feature dimension, number of possible labels, label cardinality, label density and feature type, respectively.

- $LCard(\mathcal{D}) = \frac{1}{N} \sum_{i=1}^{N} |\boldsymbol{y}_i|$ measures the average number of labels per instance.
- $LDen(\mathcal{D}) = \frac{LCard(\mathcal{D})}{L(\mathcal{D})}$ normalizes $LCard(\mathcal{D})$ by the number of possible labels.

Table 1. Characteristics of datasets.

Dataset	$\|D\|$	*dim(D)*	*L(D)*	*LCard(D)*	*LDen(D)*	*F(D)*
Parkinson-P	401	91	5	1.262	0.126	Nominal
Parkinson	401	91	10	0.798	0.160	Nominal
ML2000	2000	2000	5	1.236	0.247	Numeric
Scene	2407	294	6	1.074	0.179	Numeric
Emotions	593	72	6	1.869	0.311	Numeric
Genbase	662	1185	27	1.252	0.046	Nominal

4.2 Evaluation Metrics

To have a fair comparison, we employ five widely-used evaluation metrics, including: $HammingLoss$, $SubsetAcc$, $MacroF_1$, $MicroF_1$, $ExampleF_1$ [21].

4.3 Comparing Algorithms

We compare our proposed DMLMR algorithm with six state-of-the-art multi-label algorithms, listed as follows:

- BR [1]: first-order algorithm which transforms the multi-label learning task into multiple binary classification tasks
- CC [15]: a novel chaining method that considers the relativity between labels
- ECC [15]: state-of-the-art supervised ensemble multi-label learning method
- MLKNN [20]: is a kNN style multi-label classification algorithm, and outperforms some existing algorithms
- LLSF [7]: second-order algorithm which exploits different feature sets for the discrimination of different labels
- CAMEL [3]: a novel method to learn the label correlations via sparse reconstruction in the label space.

4.4 Experimental Results

For all these algorithms, we report the best results of the optimal parameters in terms of classification performance. 10-fold cross validation (CV) is performed on each dataset. To better characterize the comparison, we take the mean metric value as well as the standard deviation of each algorithm. Note that for all the employed multi-label evaluation metrics, their values vary within the interval [0,1]. The larger the value of them, the better the performance of the classifier for all of these evaluation metrics except $HammingLoss$.

Experimental results of our proposed DMLMR and other comparing algorithms on real-world dataset of Parkinson's disease and four benchmark multi-label datasets are listed in Table 2 and Table 3 respectively. From the results, it is obvious that DMLMR algorithm can achieve best or at least comparable performance on all datasets with different evaluation metrics, which reveals that DMLMR algorithm is a high-competitive multi-label learning algorithm.

Table 2. Performance comparison on *Parkinson-P* and *Parkinson* dataset. ↑ / ↓ indicates that the larger/smaller the better of a criterion. The best results are in bold.

Evaluation Metrics	*Parkinson-P*						
	BR	CC	ECC	MLKNN	LLSF	CAMEL	DMLMR
$HammingLoss\downarrow$	0.180±0.040	0.195±0.031	0.165±0.022	0.200±0.020	0.162±0.031	0.169±0.035	**0.161±0.027**
$SubsetAcc\uparrow$	0.404±0.121	0.506±0.081	0.491±0.069	0.351±0.070	0.442±0.075	0.486±0.093	**0.559±0.067**
$MacroF_1\uparrow$	0.407±0.088	0.393±0.075	0.402±0.077	0.244±0.038	0.397±0.077	0.378±0.050	**0.410±0.050**
$MicroF_1\uparrow$	0.541±0.104	0.512±0.082	0.549±0.061	0.411±0.068	0.553±0.093	0.565±0.088	**0.592±0.066**
$ExampleF_1\uparrow$	0.490±0.113	0.509±0.083	0.498±0.065	0.351±0.070	0.483±0.089	0.528±0.091	**0.575±0.064**
Evaluation Metrics	*Parkinson*						
	BR	CC	ECC	MLKNN	LLSF	CAMEL	DMLMR
$HammingLoss\downarrow$	0.137±0.019	0.133±0.029	0.110±0.013	0.158±0.022	0.111±0.012	0.110±0.017	**0.104±0.018**
$SubsetAcc\uparrow$	0.317±0.081	0.426±0.084	0.426±0.046	0.302±0.070	0.356±0.066	0.364±0.083	**0.489±0.096**
$MacroF_1\uparrow$	0.254±0.059	0.270±0.086	0.235±0.050	0.207±0.053	0.185±0.033	0.188±0.030	**0.273±0.012**
$MicroF_1\uparrow$	0.443±0.064	0.475±0.098	0.479±0.052	0.363±0.077	0.462±0.059	0.454±0.063	**0.532±0.069**
$ExampleF_1\uparrow$	0.443±0.073	0.519±0.089	0.480±0.049	0.377±0.068	0.434±0.064	0.423±0.078	**0.550±0.075**

4.5 Influence of Parameters

More experiments are conducted on one real-world *Parkinson-P* dataset and one benchmark multi-label *Scene* dataset to explore parameter sensitivity.

Inter-set Reweight Parameter. $\boldsymbol{\lambda_{inter}}$ is used for exploring the inter-set relationship between $\mathcal{Y}_a$ and $\mathcal{Y}_{-a}$. For *Parkinson-P* dataset, we fix $\lambda_{intra} = 1.5$, $T = 3$, and then we set λ_{inter} between 1.0 and 1.5 with an interval of 0.1. For *Scene* dataset, we fix $\lambda_{intra} = 2$, $T = 3$, and then we set λ_{inter} between 1.0 and 1.5 with an interval of 0.1.

As shown in Table 4, the performance of $\lambda_{inter} > 1.0$ is better than others when $\lambda_{inter} = 1.0$ in most cases, which validates the effectiveness of exploiting inter-set label relationship. In addition, we get optimal performance when $\lambda_{inter} = 1.7$ on *Parkinson-P* dataset and $\lambda_{inter} = 1.3$ on *Scene* dataset.

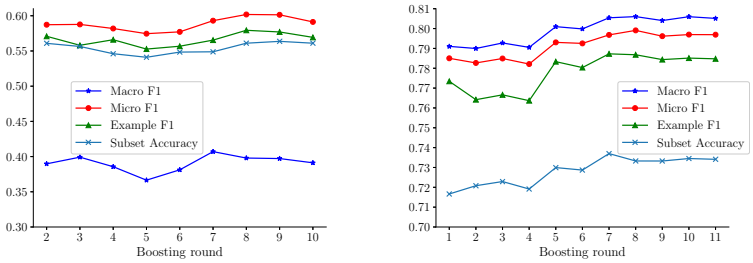
Intra-set Reweight Parameter. $\boldsymbol{\lambda_{intra}}$ is used for exploring the intra-set relationship on $\mathcal{Y}_a$ (or $\mathcal{Y}_{-a}$). Based on the above discussion of inter-set reweight parameter λ_{inter}, for *Parkinson-P* dataset, we fix $\lambda_{intra} = 1.7$, $T = 3$, and then we set λ_{inter} between 1.0 and 3 with an interval of 0.5. For *Scene* dataset, we fix $\lambda_{intra} = 1.3$, $T = 3$, and then we set λ_{inter} between 1.0 and 3 with an interval of 0.5. In Table 5, we find that $\lambda_{intra} = 1.25$ or $\lambda_{intra} = 1.5$ for *Parkinson-P* dataset may be a relatively proper setting, while $\lambda_{intra} = 2.0$ or $\lambda_{intra} = 2.5$ for *Scene* dataset.

Boosting Round$\boldsymbol{T}$. We fix $\lambda_{inter} = 1.7$, $\lambda_{intra} = 1.25$ for *Parkinson-P* dataset and fix $\lambda_{inter} = 1.3$, $\lambda_{intra} = 2.0$ for *Scene* dataset. With λ_{inter} and λ_{intra} fixed, we get the optimal results when $T = 8$ on *Parkinson-P* dataset. Similarly, we get the optimal results when $T = 7$ on *Scene* dataset.

For one thing, increasing number of boosting rounds will make classifier overly complex and may lead to overfitting. We can see from Fig. 3(a) that when boosting round $T = 10$, all evaluation metrics decline slightly, which accords with our intuition since DMLMR is an approach with a boosting framework.

Table 3. Performance comparison on four benchmark multi-label datasets. ↑ / ↓ indicates that the larger/smaller the better of a criterion. The best results are in bold.

Evaluation Metrics	*ML2000*						
	BR	CC	ECC	MLKNN	LLSF	CAMEL	DMLMR
HammingLoss ↓	0.109±0.008	0.115±0.006	0.130±0.009	0.154±0.012	0.098±0.011	**0.098±0.011**	0.101±0.013
SubsetAcc ↑	0.586±0.034	0.636±0.022	0.521±0.029	0.488±0.036	0.626±0.045	0.633±0.036	**0.660±0.040**
$MacroF_1$ ↑	0.740±0.023	0.748±0.011	0.700±0.025	0.643±0.032	0.778±0.025	0.781±0.023	**0.793±0.027**
$MicroF_1$ ↑	0.749±0.020	0.756±0.011	0.705±0.023	0.654±0.029	0.782±0.025	0.783±0.024	**0.793±0.026**
$ExampleF_1$ ↑	0.690±0.026	0.758±0.011	0.646±0.028	0.607±0.030	0.740±0.031	0.739±0.029	**0.791±0.029**
Evaluation Metrics	*Scene*						
	BR	CC	ECC	MLKNN	LLSF	CAMEL	DMLMR
HammingLoss ↓	0.104±0.009	0.105±0.010	0.094±0.005	0.089±0.008	0.106±0.006	0.076±0.006	**0.070±0.006**
SubsetAcc ↑	0.536±0.041	0.653±0.031	0.596±0.0158	0.629±0.030	0.487±0.028	0.646±0.024	**0.733±0.020**
$MacroF_1$ ↑	0.692±0.025	0.713±0.028	0.710±0.012	0.743±0.022	0.644±0.027	0.772±0.021	**0.806±0.015**
$MicroF_1$ ↑	0.688±0.027	0.703±0.029	0.705±0.016	0.739±0.021	0.643±0.027	0.763±0.019	**0.799±0.015**
$ExampleF_1$ ↑	0.627±0.034	0.705±0.030	0.639±0.014	0.710±0.024	0.536±0.034	0.695±0.027	**0.787±0.015**
Evaluation Metrics	*Emotions*						
	BR	CC	ECC	MLKNN	LLSF	CAMEL	DMLMR
HammingLoss ↓	0.216±0.019	0.216±0.029	**0.202±0.023**	0.278±0.022	0.207±0.014	0.207±0.025	0.209±0.024
SubsetAcc ↑	0.265±0.063	0.292±0.062	0.283±0.042	0.212±0.038	0.254±0.049	0.272±0.048	**0.301±0.079**
$MacroF_1$ ↑	0.618±0.035	0.614±0.060	0.627±0.039	0.496±0.033	0.616±0.033	0.615±0.058	**0.631±0.035**
$MicroF_1$ ↑	0.638±0.043	0.649±0.053	0.647±0.038	0.529±0.030	0.641±0.033	0.637±0.048	**0.662±0.035**
$ExampleF_1$ ↑	0.594±0.058	0.623±0.057	0.584±0.046	0.495±0.026	0.594±0.039	0.581±0.049	**0.628±0.035**
Evaluation Metrics	*Genbase*						
	BR	CC	ECC	MLKNN	LLSF	CAMEL	DMLMR
HammingLoss ↓	0.001±0.001	0.001±0.001	0.001±0.001	0.003±0.001	0.001±0.001	0.001±0.001	**0.001±0.001**
SubsetAcc ↑	0.970±0.034	0.976±0.024	0.971±0.016	0.943±0.028	0.982±0.015	0.979±0.019	**0.982±0.019**
$MacroF_1$ ↑	0.639±0.079	0.638±0.084	0.627±0.026	0.593±0.044	0.632±0.075	0.561±0.121	**0.677±0.051**
$MicroF_1$ ↑	0.988±0.014	0.990±0.010	0.988±0.006	0.970±0.014	0.992±0.007	0.991±0.008	**0.993±0.007**
$ExampleF_1$ ↑	0.990±0.012	0.990±0.012	0.990±0.004	0.971±0.020	0.993±0.007	0.992±0.007	**0.994±0.007**



(a) *Parkinson-P* (b) *Scene*

Fig. 3. Performance of changes made by the number of boosting rounds T on *Parkinson-P* and *Scene* dataset, with λ_{inter} and λ_{intra} fixed.

For another, classifier should have low training error and a small number of boosting rounds in order to achieve good performance. As is shown in Fig. 3, with λ_{inter} and λ_{intra} fixed, the performance of DMLMR is unstable in the initial

Table 4. Performance comparison on *Parkinson-P* and *Scene* dataset when λ_{inter} increases with λ_{intra} and T fixed. ↑ / ↓ indicates that the larger/smaller the better of a criterion. The best results are in bold.

Dataset	Evaluation Metrics	$\lambda_{intra} = 1.5, T = 3$					
		$\lambda_{inter} = 1.0$	$\lambda_{inter} = 1.1$	$\lambda_{inter} = 1.3$	$\lambda_{inter} = 1.5$	$\lambda_{inter} = 1.7$	$\lambda_{inter} = 1.9$
Parkinson-P	$HammingLoss \downarrow$	0.170±0.031	0.173±0.026	0.162±0.031	0.168±0.050	**0.161±0.027**	0.167±0.024
	$SubsetAcc \uparrow$	0.544±0.065	0.524±0.072	0.546±0.082	0.546±0.118	**0.559±0.067**	0.554±0.071
	$MacroF_1 \uparrow$	0.392±0.041	0.374±0.031	0.404±0.066	0.378±0.075	**0.410±0.050**	0.398±0.048
	$MicroF_1 \uparrow$	0.571±0.073	0.559±0.062	0.586±0.083	0.580±0.122	**0.592±0.066**	0.579±0.059
	$ExampleF_1 \uparrow$	0.559±0.067	0.539±0.066	0.565±0.084	0.566±0.116	**0.575±0.064**	0.567±0.062
Dataset	**Evaluation Metrics**	$\lambda_{intra} = 2, T = 3$					
		$\lambda_{inter} = 1.0$	$\lambda_{inter} = 1.1$	$\lambda_{inter} = 1.2$	$\lambda_{inter} = 1.3$	$\lambda_{inter} = 1.4$	$\lambda_{inter} = 1.5$
Scene	$HammingLoss \downarrow$	0.075±0.009	0.074±0.010	0.074±0.008	**0.073±0.005**	0.074±0.009	0.075±0.005
	$SubsetAcc \uparrow$	0.719±0.030	**0.724±0.035**	0.719±0.027	0.723±0.016	0.723±0.032	0.718±0.021
	$MacroF_1 \uparrow$	0.789±0.026	0.790±0.026	0.793±0.022	**0.793±0.014**	0.790±0.023	0.789±0.011
	$MicroF_1 \uparrow$	0.780±0.029	0.782±0.027	0.783±0.023	**0.785±0.015**	0.781±0.027	0.780±0.016
	$ExampleF_1 \uparrow$	0.765±0.032	0.766±0.028	0.764±0.024	**0.767±0.016**	0.765±0.031	0.763±0.017

Table 5. Performance comparison on *Parkinson-P* and *Scene* dataset when λ_{intra} increases from 1.0 to 3.0 with λ_{inter} and T fixed. ↑ / ↓ indicates that the larger/smaller the better of a criterion. The best results are in bold.

Dataset	Evaluation Metrics	$\lambda_{inter} = 1.7$, T=3					
		$\lambda_{intra} = 1.0$	$\lambda_{intra} = 1.25$	$\lambda_{intra} = 1.5$	$\lambda_{intra} = 2$	$\lambda_{intra} = 2.5$	$\lambda_{intra} = 3.0$
Parkinson-P	$HammingLoss \downarrow$	**0.159±0.026**	0.162±0.025	0.161±0.027	0.165±0.019	0.162±0.025	0.167±0.017
	$SubsetAcc \uparrow$	0.556±0.068	**0.564±0.063**	0.559±0.067	0.544±0.050	0.551±0.068	0.534±0.048
	$MacroF_1 \uparrow$	0.407±0.052	**0.410±0.062**	0.410±0.050	0.390±0.037	0.402±0.057	0.398±0.047
	$MicroF_1 \uparrow$	0.596±0.069	0.588±0.064	**0.592±0.066**	0.579±0.048	0.586±0.063	0.573±0.043
	$ExampleF_1 \uparrow$	0.576±0.070	0.572±0.064	**0.575±0.064**	0.560±0.049	0.566±0.064	0.552±0.046
Dataset	**Evaluation Metrics**	$\lambda_{inter} = 1.3$, T=3					
		$\lambda_{intra} = 1.0$	$\lambda_{intra} = 1.25$	$\lambda_{intra} = 1.5$	$\lambda_{intra} = 2.0$	$\lambda_{intra} = 2.5$	$\lambda_{intra} = 3.0$
Scene	$HammingLoss \downarrow$	0.075±0.005	0.075±0.006	0.074±0.008	**0.073±0.005**	0.073±0.008	0.074±0.006
	$SubsetAcc \uparrow$	0.721±0.025	0.721±0.022	0.721±0.030	**0.723±0.016**	0.720±0.027	0.715±0.027
	$MacroF_1 \uparrow$	0.787±0.018	0.788±0.018	0.790±0.022	0.793±0.014	**0.794±0.019**	0.789±0.017
	$MicroF_1 \uparrow$	0.779±0.017	0.780±0.018	0.782±0.023	0.785±0.015	**0.785±0.023**	0.780±0.017
	$ExampleF_1 \uparrow$	0.763±0.022	0.764±0.019	0.767±0.025	0.767±0.016	**0.768±0.024**	0.761±0.017

increasing phase of T. After that, DMLMR improves remarkably. Eventually, as the number of boosting round T increases, all curves tend to be smoother, which show convergence when $T > 6$ for *Parkinson-P* and $T > 7$ for *Scene* dataset.

5 Conclusion

Traditional Chinese Medicine (TCM) is a new way for diagnosing Parkinson's disease (PD). In this paper, we apply multi-label classification technology to diagnose PD in TCM, where we treat *TCM scales* as features and treat *syndrome*

types as multiple labels. Furthermore, we propose a novel Discriminative Multi-label Model Reuse (DMLMR) algorithm to advance diagnosing PD in TCM. DMLMR exploits label correlations by selecting discriminative label with label distribution adaptation, and then trains with model reuse. An assessment on the real-world dataset of PD shows that DMLMR obtains remarkable results in terms of various evaluation metrics, and DMLMR validates its ability of diagnosing PD in TCM. Extensive experiments on multi-label benchmark datasets show that DMLMR outperforms the state-of-the-art counterparts. In the future, how to extend to scenario with partial labels is a very interesting work.

Acknowledgment. This paper is supported by the National Key Research and Development Program of China (Grant No. 2018YFB1403400), the National Natural Science Foundation of China (Grant No. 61876080), the Key Research and Development Program of Jiangsu (Grant No. BE2019105), the Collaborative Innovation Center of Novel Software Technology and Industrialization at Nanjing University.

References

1. Boutell, M.R.: Learning multi-label scene classification. Pattern Recogn. **37**, 1757–1771 (2004)
2. Diplaris, S., Tsoumakas, G., Mitkas, P.A., Vlahavas, I.: Protein classification with multiple algorithms. In: Bozanis, P., Houstis, E.N. (eds.) PCI 2005. LNCS, vol. 3746, pp. 448–456. Springer, Heidelberg (2005). https://doi.org/10.1007/11573036_42
3. Feng, L., An, B., He, S.: Collaboration based multi-label learning. In: Thirty-Third AAAI Conference on Artificial Intelligence, pp. 3550–3557 (2019)
4. Fürnkranz, J., Hüllermeier, E., Mencía, E.L., Brinker, K.: Multilabel classification via calibrated label ranking. Mach. Learn. **73**(2), 133–153 (2008)
5. Ghamrawi, N., McCallum, A.: Collective multi-label classification. In: Proceedings of the 14th ACM international conference on Information and knowledge management, pp. 195–200. ACM (2005)
6. Gibaja, E., Ventura, S.: A tutorial on multilabel learning. ACM Comput. Surv. (CSUR) **47**(3), 1–38 (2015)
7. Huang, J., Li, G., Huang, Q., Wu, X.: Learning label specific features for multi-label classification. In: 2015 IEEE International Conference on Data Mining, pp. 181–190. IEEE (2015)
8. Huang, S.J., Yu, Y., Zhou, Z.H.: Multi-label hypothesis reuse. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 525–533. ACM (2012)
9. Kazawa, H., Izumitani, T., Taira, H., Maeda, E.: Maximal margin labeling for multi-topic text categorization. In: Advances in neural information processing systems, pp. 649–656 (2005)
10. Liu, C., Zhao, P., Huang, S.J., Jiang, Y., Zhou, Z.H.: Dual set multi-label learning. In: Thirty-Second AAAI Conference on Artificial Intelligence, (2018)
11. Luo, Y., Tao, D., Xu, C., Li, D., Xu, C.: Vector-valued multi-view semi-supervsed learning for multi-label image classification. In: Twenty-Seventh AAAI Conference on Artificial Intelligence, (2013)
12. Mohri, M., Rostamizadeh, A., Talwalkar, A.: Foundations of Machine Learning. MIT press (2018)

13. Peng, Y., Tang, C., Chen, G., Xie, J., Wang, C.: Multi-label learning by exploiting label correlations for tcm diagnosing parkinson's disease. In: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 590–594. IEEE (2017)
14. Qi, G.J., Hua, X.S., Rui, Y., Tang, J., Mei, T., Zhang, H.J.: Correlative multi-label video annotation. In: Proceedings of the 15th ACM international conference on Multimedia, pp. 17–26. ACM (2007)
15. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. Mach. Learn. **85**(3), 333 (2011)
16. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.P.: Multi-label classification of music into emotions. ISMIR. **8**, 325–330 (2008)
17. Wang, X., Sukthankar, G.: Multi-label relational neighbor classification using social context features. In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 464–472. ACM (2013)
18. Wu, B., Zhong, E., Horner, A., Yang, Q.: Music emotion recognition by multi-label multi-layer multi-instance multi-view learning. In: Proceedings of the 22nd ACM international conference on Multimedia, pp. 117–126. ACM (2014)
19. Zhang, C., Yu, Z., Hu, Q., Zhu, P., Liu, X., Wang, X.: Latent semantic aware multi-view multi-label classification. In: Thirty-Second AAAI Conference on Artificial Intelligence, (2018)
20. Zhang, M.L., Zhou, Z.H.: Ml-knn: a lazy learning approach to multi-label learning. Pattern Recogn. **40**(7), 2038–2048 (2007)
21. Zhang, M.L., Zhou, Z.H.: A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. **26**(8), 1819–1837 (2013)
22. Zhang, Y., Zeng, C., Cheng, H., Wang, C., Zhang, L.: Many could be better than all: a novel instance-oriented algorithm for multi-modal multi-label problem. In: 2019 IEEE International Conference on Multimedia and Expo (ICME), pp. 838–843. IEEE (2019)
23. Zhu, Y., Kwok, J.T., Zhou, Z.H.: Multi-label learning with global and local label correlation. IEEE Trans. Knowl. Data Eng. **30**(6), 1081–1094 (2018)