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

Feature-Induced Labeling Information Enrichment for Multi-Label Learning

Qian-Wen Zhang,[†] Yun Zhong,[♯] Min-Ling Zhang[§,*]

[†,♯] Tencent Smart Platform & Products Department, Chengdu 610041, China

[†,§] School of Computer Science and Engineering, Southeast University, Nanjing 210096, China

[§] Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China

[§] Collaborative Innovation Center of Wireless Communications Technology, China

{cowenzhang,zeuzhong}@tencent.com, zhangqw@seu.edu.cn, zhangml@seu.edu.cn* (corresponding author)

Abstract

In multi-label learning, each training example is represented by a single instance (feature vector) while associated with multiple class labels simultaneously. The task is to learn a predictive model from the training examples which can assign *a set of* proper labels for the unseen instance. Most existing approaches make use of multi-label training examples by exploiting their labeling information in a crisp manner, i.e. one class label is either fully *relevant* or *irrelevant* to the instance. In this paper, a novel multi-label learning approach is proposed which aims to enrich the labeling information by leveraging the structural information in feature space. Firstly, the underlying structure of feature space is characterized by conducting sparse reconstruction among the training examples. Secondly, the reconstruction information is conveyed from feature space to label space so as to enrich the original categorical labels into numerical ones. Thirdly, the multi-label predictive model is induced by learning from training examples with enriched labeling information. Extensive experiments on fifteen benchmark data sets clearly validate the effectiveness of the proposed feature-induced strategy for enhancing labeling information of multi-label examples.

Introduction

Multi-label learning is one of the major learning frameworks to deal with real-world objects with rich semantics, where each example is represented by a single instance (feature vector) while associated with multiple class labels simultaneously (Zhang and Zhou 2014; Gibaja and Ventura 2015; Zhou and Zhang 2017). Formally speaking, let $\mathcal{X} = \mathbb{R}^d$ be the d-dimensional feature space and $\mathcal{Y} = \{y_1, y_2, \ldots, y_q\}$ be the label space with q class labels. Given the multi-label training set $\mathcal{D} = \{(\boldsymbol{x}_i, Y_i) \mid 1 \leq i \leq p\}$, where $\boldsymbol{x}_i \in \mathcal{X}$ is the d-dimensional feature vector $(x_{i1}, x_{i2}, \ldots, x_{id})^\top$ and $Y_i \subseteq \mathcal{Y}$ is the set of relevant labels associated with $\boldsymbol{x}_i$, the task of multi-label learning is to learn a predictive model $h : \mathcal{X} \to 2^{\mathcal{Y}}$ from $\mathcal{D}$ which can assign a set of proper labels for the unseen instance.

The accessible labeling information of multi-label training example is categorical, i.e. each class label y is either regarded to be *relevant* ($y \in Y_i$) or *irrelevant* ($y \notin Y_i$) for instance $\boldsymbol{x}_i$. Accordingly, most existing approaches learn from multi-label training examples by exploiting their labeling information in a crisp manner, e.g. decomposing multi-label examples into multiple binary examples, considering the co-occurring patterns of pairwise labels, enforcing ranking between relevant and irrelevant labels, among others (Zhang and Zhou 2014).

Nonetheless, recent studies show that categorical labeling information is actually a simplification of the rich semantics encoded by multi-label training examples (Li, Zhang, and Geng 2015; Zhang and Wu 2015; Hou, Geng, and Zhang 2016). For instance, a multi-scenery image may exhibit different region size for each scenery, a multi-category document may have different topical importance for each category, and a multi-functionality gene may have different expression level for each functionality, etc. Therefore, it is a natural choice to enrich the labeling information of multi-label training examples so as to induce multi-label predictive model with strong generalization performance.

In light of the above observation, a novel multi-label learning approach named MLFE, i.e. *Multi-label Learning with Feature-induced labeling information Enrichment*, is proposed. The basic strategy of MLFE is to enrich the labeling information of multi-label examples by leveraging the structural information in the feature space. Specifically, the underlying structure of feature space is characterized by the sparse reconstruction relationships among training examples. After that, the reconstruction information is utilized to guide the enrichment process of turning categorical labeling information into numerical labeling information. Then, the desired multi-label predictive model is learned from training examples with enriched labeling information based on tailored multivariate regression techniques. Experimental studies across a wide range of benchmark data sets show that MLFE achieves highly competitive performance against other state-of-the-art multi-label learning approaches.

The rest of this paper is organized as follows. Firstly, technical details of the proposed approach are introduced. Secondly, related work on multi-label learning is briefly discussed. Thirdly, experimental results of comparative studies are reported. Finally, we conclude this paper.

The MLFE Approach

The learning procedure of MLFE consists of three steps, including structural information discovery, labeling informa-

tion enrichment, and predictive model induction. Technical details of these steps are scrutinized as follows.

Structural Information Discovery

To characterize the underlying structure of feature space, MLFE works by constructing a weighted directed graph $G = (V, E, \mathbf{W})$ where the vertex set $V = \{\boldsymbol{x}_i \mid 1 \le i \le p\}$ corresponds to the set of training instances. Accordingly, the set of directed edges $E = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid w_{ij} \neq 0, 1 \le i \neq j \le p\}$ connect any pair of instances $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ with nonzero weight.

Intuitively, the weight matrix $\mathbf{W} = [w_{ij}]_{p\times p}$ encodes the relationships among all training examples, where w_{ij} reflects the influence of $\boldsymbol{x}_i$ over $\boldsymbol{x}_j$. In this paper, the weight matrix is instantiated by modeling the relationship between one example and all the other examples via sparse reconstruction. In this way, the relationship among all training examples is exploited in a global and concise way. For each instance $\boldsymbol{x}_i$, MLFE aims to reconstruct $\boldsymbol{x}_i$ from all the other instances in the training set, i.e. $V \setminus \{\boldsymbol{x}_i\}$. Let $\mathbf{A}_i = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{i-1}, \boldsymbol{x}_{i+1}, \ldots, \boldsymbol{x}_p]$ denote the $d \times (p-1)$ matrix formed by concatenating all training instances other than $\boldsymbol{x}_i$, and $\boldsymbol{v}_i = [w_{1i}, \ldots, w_{i-1,i}, w_{i+1,i}, \ldots, w_{pi}]^\top$ denote the $(p-1)$-dimensional reconstruction coefficients. Under canonical sparse representation, the coefficient vector $\boldsymbol{v}_i$ is learned by solving the following optimization problem:

$$\min_{\boldsymbol{v}_i} \ \|\mathbf{A}_i\boldsymbol{v}_i - \boldsymbol{x}_i\|_2^2 + \lambda\|\boldsymbol{v}_i\|_1 \tag{1}$$

Here, the first and second terms in Eq.(1) control the linear reconstruction error via L2 norm and the sparsity of the coefficients via L1 norm respectively. The relative importance of each term is balanced by the tradeoff parameter λ. To solve Eq.(1), MLFE adapts the popular ADMM (Alternating Direction Method of Multiplier) techniques (Bertsekas and Tsitsiklis 1989; Ghadimi et al. 2015) which reformulate the above optimization problem into the following equivalent form:

$$\begin{aligned} \min_{\{\boldsymbol{v}_i, \boldsymbol{z}_i\}} \ & \frac{1}{2}\|\mathbf{A}_i\boldsymbol{v}_i - \boldsymbol{x}_i\|_2^2 + \lambda\|\boldsymbol{z}_i\|_1 \\ \text{s.t.} \ & \boldsymbol{v}_i - \boldsymbol{z}_i = \mathbf{0} \end{aligned} \tag{2}$$

Following the ADMM procedure, the constrained optimization problem in Eq.(2) can be solved as a series of unconstrained minimization problems with augmented Lagrangian function:

$$\begin{aligned} L(\boldsymbol{v}_i, \boldsymbol{z}_i, \boldsymbol{\mu}_i) = & \frac{1}{2}\|\mathbf{A}_i\boldsymbol{v}_i - \boldsymbol{x}_i\|_2^2 + \\ & \lambda\|\boldsymbol{z}_i\|_1 + \boldsymbol{\mu}_i^\top(\boldsymbol{v}_i - \boldsymbol{z}_i) + \frac{\rho}{2}\|\boldsymbol{v}_i - \boldsymbol{z}_i\|_2^2 \end{aligned} \tag{3}$$

Here, ρ is the penalty parameter and $\boldsymbol{\mu}_i$ is the estimate of Lagrange multiplier. A sequential minimization of the variables $\boldsymbol{v}_i$, $\boldsymbol{z}_i$ and $\boldsymbol{\mu}_i$ can be conducted by the scaled ADMM iterations:

$$\begin{aligned} \boldsymbol{v}_i^{(k+1)} &= (\mathbf{A}_i^\top\mathbf{A}_i + \rho\mathbf{I})^{-1}(\mathbf{A}_i^\top\boldsymbol{x}_i + \rho\boldsymbol{z}_i^{(k)} - \boldsymbol{\mu}_i^{(k)}) \\ \boldsymbol{z}_i^{(k+1)} &= (\boldsymbol{v}_i^{(k+1)} + \boldsymbol{\mu}_i^{(k)}/\rho - \lambda/\rho)_+ - \\ &\qquad (-(\boldsymbol{v}_i^{(k+1)} + \boldsymbol{\mu}_i^{(k)}/\rho) - \lambda/\rho)_+ \\ \boldsymbol{\mu}_i^{(k+1)} &= \boldsymbol{\mu}_i^{(k)} + \rho(\boldsymbol{v}_i^{(k+1)} - \boldsymbol{z}_i^{(k+1)}) \end{aligned} \tag{4}$$

By solving the sparse reconstruction problem of Eq.(2) with ADMM techniques for each instance $\boldsymbol{x}_i$ $(1 \le i \le p)$, the weight matrix $\mathbf{W}$ can be instantiated with $\boldsymbol{v}_i$ $(1 \le i \le p)$ and zero diagonal elements. Note that in most cases $w_{ij} \neq w_{ji}$, as the influence of $\boldsymbol{x}_i$ in reconstructing $\boldsymbol{x}_j$ is generally different to the influence of $\boldsymbol{x}_j$ in reconstructing $\boldsymbol{x}_i$.

Labeling Information Enrichment

For each multi-label training example $(\boldsymbol{x}_i, Y_i)$, its labeling information can be represented by a categorical (binary) vector $\boldsymbol{t}_i = (t_{i1}, t_{i2}, \ldots, t_{iq})^\top$ where $t_{ik} = 1$ if $y_k \in Y_i$, and $t_{ik} = -1$ otherwise. The goal of MLFE is trying to transform the binary labeling vector $\boldsymbol{t}_i \in \{1, -1\}^q$ into a numerical labeling vector $\boldsymbol{u}_i = (u_{i1}, u_{i2}, \ldots, u_{iq})^\top \in \mathbb{R}^q$ which encodes richer semantics for predictive model induction.

Considering that the weight matrix $\mathbf{W}$ characterizes the structural information among training examples in the feature space, the reconstruction error over the training set corresponds to $E(\mathbf{W}) = \sum_{i=1}^p \|\boldsymbol{x}_i - \sum_{j=1}^p w_{ji}\boldsymbol{x}_j\|_2^2$. Accordingly, suppose that the structural relationship specified in the feature space also holds in the output space, i.e. the influence of $\boldsymbol{x}_i$ over $\boldsymbol{x}_j$ is also conveyed to $\boldsymbol{u}_i$ over $\boldsymbol{u}_j$. Therefore, the goodness of the numerical labeling vectors can be measured by the reconstruction errors in the label space: $\Phi(\mathbf{U}) = \sum_{i=1}^p \|\boldsymbol{u}_i - \sum_{j=1}^p w_{ji}\boldsymbol{u}_j\|_2^2$, where $\mathbf{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_p]$.

Specifically, the enriched labeling information is generated via leveraging the structural information encoded in $\mathbf{W}$ by solving the following optimization problem:

$$\begin{aligned} \min_{\mathbf{U}} \ & \sum_{i=1}^p \|\boldsymbol{u}_i - \sum_{j=1}^p w_{ji}\boldsymbol{u}_j\|_2^2 \\ \text{s.t.} \ & c_1 \le t_{ij}u_{ij} \le c_2 \ \ (1 \le i \le p, 1 \le j \le q) \end{aligned} \tag{5}$$

Here, the constraint in Eq.(5) ensures that the numerical label possesses the same sign with the binary label and takes value with reasonable magnitude. Obviously, Eq.(5) corresponds to a standard quadratic programming (QP) problem which can be efficiently solved by any off-the-shelf QP toolbox.

Predictive Model Induction

Given the enriched labeling information $\mathbf{U}$, the original multi-label training set can be transformed into its enriched version $\tilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \boldsymbol{u}_i) \mid 1 \le i \le p\}$. As the response variables $\boldsymbol{u}_i$ for each transformed multi-label training example $(\boldsymbol{x}_i, \boldsymbol{u}_i)$ are real-valued, it is natural to induce the predictive model by employing *multi-output regression* techniques. Among various ways towards implementing multi-output regression, we choose to adapt the multi-regression support vector machines (MSVR) (Chung et al. 2015; Sánchez-Fernández et al. 2004; Tuia et al. 2011) which is capable of incorporating kernel trick for nonlinear modeling.

Let $\varphi(\cdot) : \mathbb{R}^d \mapsto \mathbb{R}^{\mathcal{H}_\kappa}$ be the (implicit) nonlinear mapping from the original feature space to the higher-dimensional Reproducing Kernel Hilbert Space (RKHS) via kernel function $\kappa : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$. Furthermore, let $\{(\boldsymbol{\theta}_j, b_j) \mid 1 \le j \le q\}$ denote the multi-output regression model in the RKHS,

where one linear predictor $(\boldsymbol{\theta}_j, b_j)$ is assumed for each class label $y_j \in \mathcal{Y}$. MLFE induces the regression model by minimizing the following objective function:

$$\Omega(\boldsymbol{\Theta}, \boldsymbol{b}) = \frac{1}{2}\sum_{j=1}^{q}\|\boldsymbol{\theta}_j\|_2^2 + \beta_1\sum_{i=1}^{p}\Omega_1(u_i) + \beta_2\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_2(o_{ij}) + \beta_3\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_3(r_{ij}) \tag{6}$$

Here, $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_q]$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_q]^\top$ represent the regression model's weight matrix and bias vector respectively.

As shown in Eq.(6), the first term controls the complexity of the induced model. Furthermore, the second term is defined based on the ϵ-insensitive cost which measures how the model predictions fit the numerical labeling vectors:

$$\Omega_1(u_i) = \begin{cases} 0, & u_i < \epsilon \\ (u_i - \epsilon)^2, & u_i \geq \epsilon \end{cases} \tag{7}$$

Here, $u_i = ||\boldsymbol{e}_i|| = \sqrt{\boldsymbol{e}_i^\top \boldsymbol{e}_i}$ with $\boldsymbol{e}_i = \boldsymbol{u}_i - \boldsymbol{\Theta}^\top\varphi(\boldsymbol{x}_i) - \boldsymbol{b}$. Based on the ϵ-insensitive term, correlations among all class labels are exploited by considering their predictive outputs simultaneously to yield a unique input to $\Omega_1(\cdot)$. The third term is used to penalize the case where the sign of predictive output is different to that of original binary label:

$$\Omega_2(o_{ij}) = \begin{cases} 0, & o_{ij} > 0 \\ -o_{ij}, & o_{ij} \leq 0 \end{cases}, \tag{8}$$

where $o_{ij} = t_{ij}\left(\boldsymbol{\theta}_j^\top\varphi(\boldsymbol{x}_i) + b_j\right)$.

Generally, multi-label examples only assume limited number of relevant labels over the label space, i.e. $|Y_i| \ll q$ (Zhang and Zhou 2014; Gibaja and Ventura 2015). The fourth term in Eq.(6) penalizes the case where the predictive model yields large number of relevant labels for the training example:

$$\Omega_3(r_{ij}) = \begin{cases} 1, & r_{ij} > 0 \\ 0, & r_{ij} \leq 0 \end{cases}, \tag{9}$$

where $r_{ij} = \boldsymbol{\theta}_j^\top\varphi(\boldsymbol{x}_i) + b_j$.

To minimize the objective function $\Omega(\boldsymbol{\Theta}, \boldsymbol{b})$, MLFE employs the quasi-Newton iterative method named Iterative Re-Weighted Least Square (IRWLS) (Sánchez-Fernández et al. 2004; Tuia et al. 2011). In each iteration, the descending direction for model refinement is determined analytically by solving linear systems of equations. Let $\{\boldsymbol{\Theta}^{(k)}, \boldsymbol{b}^{(k)}\}$ denote the current model after k-th iteration, IRWLS works by firstly approximating $\Omega(\boldsymbol{\Theta}, \boldsymbol{b})$ based on first-order Taylor expansion over the ϵ-insensitive term $\Omega_1(u_i)$:

$$\Omega'(\boldsymbol{\Theta}, \boldsymbol{b}) = \frac{1}{2}\sum_{j=1}^{q}\|\boldsymbol{\theta}_j\|^2 \tag{10}$$

$$+\beta_1\left(\sum_{i=1}^{p}\Omega_1(u_i^{(k)}) + \left.\frac{d\Omega_1(u)}{du}\right|_{u_i^{(k)}}\frac{(\boldsymbol{e}_i^{(k)})^\top}{u_i^{(k)}}\left(\boldsymbol{e}_i - \boldsymbol{e}_i^{(k)}\right)\right)$$
$$+\beta_2\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_2(o_{ij}) + \beta_3\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_3(r_{ij})$$

Here, $\boldsymbol{e}_i^{(k)}$ and $u_i^{(k)}$ are calculated based on the current model $\{\boldsymbol{\Theta}^{(k)}, \boldsymbol{b}^{(k)}\}$. Then, a quadratic approximation to $d\Omega_1(u)/du$ is further constructed to enable identifying analytical solution to the descending direction:

$$\Omega''(\boldsymbol{\Theta}, \boldsymbol{b}) = \frac{1}{2}\sum_{j=1}^{q}\|\boldsymbol{\theta}_j\|^2 \tag{11}$$
$$+\beta_1\left(\sum_{i=1}^{p}\Omega_1(u_i^{(k)}) + \left.\frac{d\Omega_1(u)}{du}\right|_{u_i^{(k)}}\frac{u_i^2 - (u_i^{(k)})^2}{2u_i^{(k)}}\right)$$
$$+\beta_2\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_2(o_{ij}) + \beta_3\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_3(r_{ij})$$

$$= \frac{1}{2}\sum_{j=1}^{q}\|\boldsymbol{\theta}_j\|^2 + \frac{1}{2}\beta_1\sum_{i=1}^{p}\sum_{j=1}^{q}a_i u_i^2 + \beta_2\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_2(o_{ij}) + \beta_3\sum_{i=1}^{p}\sum_{j=1}^{q}\Omega_3(r_{ij}) + \tau$$

where

$$a_i = \frac{1}{u_i^{(k)}}\left.\frac{d\Omega_1(u)}{du}\right|_{u_i^{(k)}} = \begin{cases} 0, & u_i^{(k)} < \varepsilon \\ \frac{2\left(u_i^{(k)} - \varepsilon\right)}{u_i^{(k)}}, & u_i^{(k)} \geq \varepsilon \end{cases}$$

and τ is a constant term which does not depend on $\{\boldsymbol{\Theta}^{(k)}, \boldsymbol{b}^{(k)}\}$.

Thereafter, minimization of $\Omega''(\boldsymbol{\Theta}, \boldsymbol{b})$ can be decoupled for each class label, whose solution for $(\boldsymbol{\theta}_j, b_j)$ $(1 \leq j \leq q)$ is found by equating the corresponding gradient to zero:

$$\frac{\partial\Omega''}{\partial\boldsymbol{\theta}_j} = \boldsymbol{\theta}_j - \beta_1\sum_{i=1}^{p}a_i\varphi(\boldsymbol{x}_i)(u_{ij} - \boldsymbol{\theta}_j^\top\varphi(\boldsymbol{x}_i) - b_j) - \beta_2\sum_{i=1}^{p}\varphi(\boldsymbol{x}_i)\sigma(-o_{ij})t_{ij} = \boldsymbol{0} \tag{12}$$

$$\frac{\partial\Omega''}{\partial b_j} = -\beta_1\sum_{i=1}^{p}a_i(u_{ij} - \boldsymbol{\theta}_j^\top\varphi(\boldsymbol{x}_i) - b_j) - \beta_2\sum_{i=1}^{p}\sigma(-o_{ij})t_{ij} = \boldsymbol{0} \tag{13}$$

where $\sigma(z) = 1$ if $z > 0$, and $\sigma(z) = 0$ otherwise. Accordingly, Eqs.(12) and (13) can be expressed as a linear system of equations:

$$\begin{bmatrix} \beta_1\boldsymbol{\Phi}^\top\mathbf{D}_a\boldsymbol{\Phi} + \mathbf{I} & \beta_1\boldsymbol{\Phi}^\top\boldsymbol{a} \\ \beta_1\boldsymbol{a}^\top\boldsymbol{\Phi} & \beta_1\mathbf{1}^\top\boldsymbol{a} \end{bmatrix}\begin{bmatrix} \boldsymbol{\theta}_j \\ b_j \end{bmatrix} = \begin{bmatrix} \beta_1\boldsymbol{\Phi}^\top\mathbf{D}_a\boldsymbol{u}^j + \beta_2\boldsymbol{\Phi}^\top\mathbf{D}_j\boldsymbol{t}^j \\ \beta_1\boldsymbol{a}^\top\boldsymbol{u}^j + \beta_2(\boldsymbol{\sigma}^j)^\top\boldsymbol{t}^j \end{bmatrix} \tag{14}$$

Here, $\boldsymbol{\Phi} = [\varphi(\boldsymbol{x}_1), ..., \varphi(\boldsymbol{x}_p)]^\top$, $\mathbf{D}_a = [d_{il}]_{p\times p}$ with $d_{il} = a_i\delta_{il}$ (δ_{il} is the Kronecker's delta), $\boldsymbol{a} = [a_1, ..., a_p]^\top$, $\boldsymbol{u}^j = [u_{1j}, u_{2j}, \ldots, u_{pj}]^\top$, $\mathbf{D}_j = [d'_{il}]_{p\times p}$ with $d'_{il} = \sigma(-o_{ij})\delta_{il}$, $\boldsymbol{t}^j = [t_{1j}, t_{2j}, \ldots, t_{pj}]^\top$, $\boldsymbol{\sigma}^j =$

Table 1: Characteristics of the multi-label experimental data sets.

Data set	$\|\mathcal{S}\|$	$dim(\mathcal{S})$	$L(\mathcal{S})$	$F(\mathcal{S})$	$LCard(\mathcal{S})$	$LDen(\mathcal{S})$	$DL(\mathcal{S})$	$PDL(\mathcal{S})$	Domain
cal500	502	68	174	numeric	26.044	0.150	502	1.000	audio
emotions	593	72	6	numeric	1.868	0.311	27	0.046	audio
medical	978	1,449	45	nominal	1.245	0.028	94	0.096	text
llog	1,460	1,004	75	nominal	1.180	0.016	304	0.208	text
msra	1,868	898	19	numeric	6.315	0.332	947	0.507	image
image	2,000	294	5	numeric	1.236	0.247	20	0.010	image
scene	2.407	294	5	numeric	1.074	0.179	15	0.006	image
yeast	2.417	103	14	numeric	4.237	0.303	198	0.082	biology
slashdot	3,782	1,079	22	nominal	1.181	0.054	156	0.041	text
corel5k	5,000	499	374	nominal	3.522	0.009	3,175	0.635	image
rcv1-s1	6,000	500	101	nominal	2.880	0.029	1,028	0.171	text
rcv1-s2	6,000	500	101	nominal	2.634	0.026	954	0.159	text
rcv1-s3	6,000	500	101	nominal	2.614	0.026	939	0.156	text
rcv1-s4	6,000	500	101	nominal	2.484	0.025	816	0.136	text
rcv1-s5	6,000	500	101	nominal	2.642	0.026	946	0.158	text

$[\sigma(-o_{1j}), \sigma(-o_{2j}), \ldots, \sigma(-o_{pj})]^\top$. Then, the solution of Eq. (14) is used to form the descending direction for minimizing the objective function, and the subsequent model $\{\boldsymbol{\Theta}^{(k+1)}, \boldsymbol{b}^{(k+1)}\}$ is updated by invoking line search procedure from $\{\boldsymbol{\Theta}^{(k)}, \boldsymbol{b}^{(k)}\}$ along this direction.

According to the Representer Theorem (Schölkopf and Smola 2001), under fairly general conditions, the predictive model can be expressed as a linear combination of the training examples in the feature space, i.e., $\boldsymbol{\theta}_j = \sum_{i=1}^{p} \varphi(\boldsymbol{x}_i)\kappa_{ij} = \boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\kappa}_j$. By replacing this expression into Eq. (14), the inner product $< \varphi(\boldsymbol{x}_i), \varphi(\boldsymbol{x}_j) >$ naturally follows and then kernel trick can be applied to accommodate nonlinear predictive models.

Related Work

Significant amount of multi-label learning algorithms have been proposed in recent years, which can be roughly categorized into three groups based on the *order of label correlations* being considered (Zhang and Zhou 2014; Gibaja and Ventura 2015). For first-order approaches, multi-label learning problem is tackled in a label-by-label style ignoring the co-existence of other labels (Boutell et al. 2004; Zhang and Zhou 2007). For second-order approaches, multi-label learning problem is tackled by considering pairwise relations between labels (Elisseeff and Weston 2002; Fürnkranz et al. 2008). For high-order approaches, multi-label learning problem is tackled by considering high-order relations among label subsets or all the class labels (Ji et al. 2010; Read et al. 2011; Tsoumakas, Katakis, and Vlahavas 2011). Note that existing label correlation exploitation strategies make use of the labeling information in a crisp manner, while MLFE models the high-order correlations among class labels with enriched real-valued labeling information.

There have been some works on multi-label learning which make use of auxiliary labeling information for model induction. For instance, in (Cheng, Dembczyński, and Hüllermeier 2010), an *ordinal scale* is assumed to characterize the membership degree and each label of the training example is affiliated with an ordinal grade. In (Xu, Li, and Zhou 2013), a *full ordering* is assumed to be known to rank relevant labels of the training example. In (Geng 2016), a *multinomial distribution* is assumed to be specified over the label space to characterize the descriptive degree of each class label. Specifically, those auxiliary labeling information are explicitly given and accessible to the learning algorithm, while MLFE does not assume the availability of such explicit information.

Recently, there are also some attempts which aim to facilitate multi-label predictive model induction by manipulating the feature space. In (Zhang and Wu 2015), label-specific features are constructed by conducting clustering analysis over the positive and negative instances w.r.t. each class label. In (Li, Zhang, and Geng 2015), label propagation is conducted over the fully-connected affinity graph specified over the feature space. In (Hou, Geng, and Zhang 2016), the manifold structure of feature space is characterized by the weighted k-nearest neighbor graph defined over training examples. Different to those approaches, MLFE serves as the first attempt which enriches labeling information by exploiting the structure of feature space via sparse reconstruction.

Experiments

Experimental Setup

In this subsection, the benchmark data sets, comparing algorithms, and evaluation metrics used for experimental studies are introduced.

A total of fifteen benchmark multi-label data sets are employed for performance evaluation.[1] For each multi-label data set $\mathcal{S}$, we use $|\mathcal{S}|$, $dim(\mathcal{S})$, $L(\mathcal{S})$ and $F(\mathcal{S})$ to represent the number of examples, feature dimensionality, size of label space and feature type respectively. In addition, properties of the data set are further characterized by several multi-label statistics, including label cardinality $LCard(\mathcal{S})$, label density $LDen(\mathcal{S})$, distinct label sets $DL(\mathcal{S})$ and proportion of distinct label sets $PDL(\mathcal{S})$. Detailed definitions on these properties can be found in (Read et al. 2011).

[1]Publicly available at http://mulan.sourceforge.net/datasets.html and http://meka.sourceforge.net/#datasets

Table 2: Predictive performance of each comparing algorithm (mean ± std. deviation) on the regular-scale data sets.

Comparing algorithms	One-error ↓							
	cal500	emotions	medical	llog	msra	image	scene	yeast
MLFE	0.129±0.047	0.260±0.030	**0.113±0.041**	**0.669±0.044**	**0.040±0.008**	**0.258±0.020**	**0.149±0.023**	0.231±0.015
LIFT	**0.116±0.040**	**0.255±0.037**	0.152±0.039	0.705±0.049	0.057±0.014	0.277±0.023	0.162±0.023	**0.229±0.013**
RELIAB	0.159±0.062	0.277±0.036	0.168±0.039	0.746±0.028	0.066±0.017	0.350±0.023	0.270±0.023	0.247±0.017
ML^2	0.166±0.078	0.268±0.032	0.129±0.028	0.699±0.038	**0.040±0.013**	0.267±0.022	0.156±0.029	0.254±0.027
CLR	0.269±0.061	0.322±0.032	0.360±0.170	0.830±0.058	0.144±0.027	0.437±0.019	0.344±0.027	0.241±0.012
RAKEL	0.611±0.084	0.315±0.074	0.246±0.038	0.879±0.026	0.239±0.031	0.412±0.029	0.339±0.027	0.280±0.016
Comparing algorithms	Coverage ↓							
	cal500	emotions	medical	llog	msra	image	scene	yeast
MLFE	0.764±0.013	**0.278±0.022**	**0.031±0.012**	0.227±0.027	**0.518±0.010**	**0.163±0.014**	**0.016±0.009**	**0.452±0.010**
LIFT	0.759±0.020	0.280±0.025	0.048±0.022	0.173±0.020	0.539±0.011	0.172±0.014	0.020±0.009	0.459±0.010
RELIAB	**0.738±0.013**	0.299±0.029	0.047±0.016	**0.161±0.020**	0.541±0.011	0.199±0.012	0.107±0.011	0.457±0.004
ML^2	0.805±0.013	0.279±0.022	**0.031±0.011**	0.181±0.019	0.522±0.011	0.165±0.012	0.018±0.009	0.455±0.011
CLR	0.795±0.008	0.334±0.020	0.080±0.068	0.186±0.044	0.618±0.013	0.247±0.016	0.137±0.017	0.480±0.008
RAKEL	0.964±0.006	0.348±0.021	0.089±0.019	0.340±0.023	0.670±0.010	0.253±0.009	0.174±0.015	0.564±0.008
Comparing algorithms	Ranking loss ↓							
	cal500	emotions	medical	llog	msra	image	scene	yeast
MLFE	0.185±0.008	**0.142±0.020**	0.020±0.010	0.233±0.027	**0.118±0.006**	**0.135±0.014**	**0.052±0.010**	**0.167±0.007**
LIFT	0.182±0.004	**0.142±0.025**	0.033±0.017	0.157±0.021	0.125±0.005	0.147±0.015	0.056±0.009	0.170±0.006
RELIAB	**0.177±0.005**	0.161±0.031	0.031±0.012	0.128±0.018	0.131±0.005	0.181±0.013	0.090±0.010	0.180±0.008
ML^2	0.210±0.009	0.144±0.023	**0.019±0.008**	0.170±0.020	0.119±0.006	0.138±0.011	0.055±0.011	0.172±0.009
CLR	0.224±0.008	0.199±0.024	0.065±0.059	**0.152±0.039**	0.190±0.009	0.243±0.018	0.119±0.016	0.187±0.005
RAKEL	0.364±0.014	0.217±0.026	0.067±0.015	0.292±0.028	0.232±0.011	0.250±0.012	0.154±0.014	0.250±0.005
Comparing algorithms	Average precision ↑							
	cal500	emotions	medical	llog	msra	image	scene	yeast
MLFE	0.490±0.025	0.815±0.020	**0.914±0.024**	0.393±0.036	0.827±0.008	**0.833±0.014**	**0.917±0.014**	**0.767±0.010**
LIFT	0.503±0.015	**0.817±0.027**	0.874±0.029	**0.402±0.039**	0.830±0.008	0.819±0.015	0.909±0.013	0.762±0.008
RELIAB	**0.507±0.019**	0.797±0.028	0.869±0.028	0.393±0.034	0.818±0.009	0.776±0.013	0.840±0.014	0.744±0.011
ML^2	0.456±0.027	0.811±0.022	0.903±0.021	0.396±0.036	**0.836±0.007**	0.827±0.014	0.913±0.016	0.757±0.014
CLR	0.436±0.019	0.762±0.024	0.687±0.192	0.295±0.075	0.741±0.013	0.718±0.014	0.795±0.018	0.745±0.008
RAKEL	0.332±0.019	0.766±0.031	0.802±0.027	0.233±0.026	0.694±0.014	0.725±0.013	0.780±0.018	0.710±0.009
Comparing algorithms	Macro-averaging F1 ↑							
	cal500	emotions	medical	llog	msra	image	scene	yeast
MLFE	0.237±0.022	**0.670±0.052**	**0.720±0.073**	**0.461±0.062**	**0.553±0.015**	**0.658±0.024**	**0.819±0.026**	0.425±0.023
LIFT	0.176±0.021	0.630±0.042	0.690±0.079	0.399±0.057	0.516±0.017	0.621±0.035	0.797±0.015	0.388±0.022
RELIAB	**0.301±0.022**	0.650±0.039	0.712±0.053	0.392±0.058	0.546±0.014	0.556±0.035	0.665±0.025	0.405±0.024
ML^2	0.236±0.021	0.650±0.047	0.674±0.061	0.370±0.060	0.548±0.012	0.646±0.029	0.799±0.029	**0.443±0.025**
CLR	0.211±0.025	0.601±0.038	0.600±0.129	0.395±0.062	0.499±0.017	0.525±0.022	0.620±0.025	0.400±0.018
RAKEL	0.187±0.020	0.618±0.036	0.672±0.058	0.366±0.051	0.492±0.020	0.540±0.012	0.644±0.019	0.430±0.014
Comparing algorithms	Micro-averaging F1 ↑							
	cal500	emotions	medical	llog	msra	image	scene	yeast
MLFE	0.374±0.024	**0.684±0.043**	**0.816±0.032**	**0.211±0.042**	**0.725±0.009**	**0.657±0.021**	**0.810±0.026**	**0.649±0.014**
LIFT	0.323±0.016	0.659±0.024	0.775±0.036	0.177±0.037	0.716±0.015	0.622±0.031	0.787±0.015	0.645±0.011
RELIAB	**0.483±0.012**	0.647±0.036	0.725±0.029	0.181±0.037	0.714±0.007	0.560±0.030	0.654±0.025	0.637±0.011
ML^2	0.358±0.027	0.661±0.039	0.782±0.026	0.057±0.021	0.722±0.008	0.645±0.028	0.788±0.029	0.641±0.014
CLR	0.326±0.019	0.614±0.037	0.598±0.157	0.176±0.049	0.624±0.010	0.525±0.019	0.612±0.026	0.628±0.012
RAKEL	0.355±0.018	0.634±0.031	0.685±0.031	0.148±0.027	0.613±0.015	0.540±0.011	0.636±0.023	0.632±0.009

Table 1 summarizes detailed characteristics of the benchmark data sets, which are roughly organized in ascending order of $|\mathcal{S}|$ with eight being regular-scale (first part, $|\mathcal{S}| < 3,000$) and seven being large-scale (second part, $|\mathcal{S}| \geq 3,000$). As shown in Table 1, the fifteen experimental data sets exhibit diversified multi-label properties serving as a solid basis for thorough comparative studies.

The performance of MLFE is compared against five state-of-the-art multi-label learning algorithms, including the second-order approach CLR (Fürnkranz et al. 2008), the high-order approach RAKEL (Tsoumakas, Katakis, and Vlahavas 2011), and three feature-aware approaches LIFT (Zhang and Wu 2015), RELIAB (Li, Zhang, and Geng 2015) and ML^2 (Hou, Geng, and Zhang 2016).

For the comparing algorithms, parameter configurations suggested in the literatures are used, i.e. CLR: ensemble size $\binom{q}{2}$; RAKEL: ensemble size $2q$ with $k = 3$; LIFT: ratio parameter $r = 0.1$; RELIAB: propagation parameters τ and β chosen among $\{0.1, 0.15, \ldots, 0.5\}$ and $\{10^{-3}, 10^{-2}, \ldots, 10\}$; ML^2: balance parameter $\lambda = 1$, cost parameters C_1 and C_2 chosen among $\{1, 2, \ldots, 10\}$. For MLFE, parameters β_1, β_2 and β_3 in Eq.(6) are chosen among $\{1, 2, \ldots, 10\}$, $\{1, 10, 15\}$ and $\{1, 10\}$ respectively with cross-validation on the training set.[2]

In this paper, six widely-used multi-label metrics are employed for performance evaluation, including four example-based metrics *one-error*, *coverage*, *ranking loss*, *average*

[2]In this paper, the parameters ρ and λ in Eq.(3) are fixed to be 1 and $\frac{1}{100}\|\mathbf{A}_i^{\dagger}\boldsymbol{x}_i\|_\infty$, the parameters c_1 and c_2 in Eq.(5) are fixed to be 1 and 2. Furthermore, RBF kernel is utilized to instantiate the multi-output SVR employed by MLFE.

Table 3: Predictive performance of each comparing algorithm (mean ± std. deviation) on the large-scale data sets.

Comparing algorithms	One-error ↓						
	slashdot	corel5k	rcv1subset1	rcv1subset2	rcv1subset3	rcv1subset4	rcv1subset5
MLFE	0.372±0.020	0.646±0.021	0.413±0.013	0.463±0.014	0.472±0.023	0.453±0.020	0.445±0.015
LIFT	0.397±0.026	0.669±0.014	0.415±0.019	0.455±0.012	0.473±0.020	0.457±0.022	0.445±0.017
RELIAB	0.516±0.017	0.718±0.015	0.467±0.020	0.476±0.011	0.477±0.024	0.462±0.015	0.467±0.017
ML^2	**0.363±0.018**	**0.627±0.023**	**0.396±0.021**	**0.448±0.014**	**0.461±0.020**	**0.438±0.017**	**0.437±0.017**
CLR	0.979±0.005	0.741±0.018	0.501±0.027	0.507±0.019	0.533±0.037	0.499±0.017	0.503±0.018
RAKEL	0.615±0.020	0.872±0.014	0.623±0.023	0.592±0.017	0.598±0.018	0.592±0.013	0.595±0.021
Comparing algorithms	Coverage ↓						
	slashdot	corel5k	rcv1subset1	rcv1subset2	rcv1subset3	rcv1subset4	rcv1subset5
MLFE	0.117±0.008	**0.263±0.013**	**0.100±0.007**	**0.094±0.005**	**0.096±0.004**	**0.081±0.006**	**0.094±0.006**
LIFT	0.107±0.009	0.286±0.013	0.132±0.009	0.139±0.006	0.142±0.006	0.120±0.009	0.134±0.004
RELIAB	0.134±0.005	0.300±0.011	0.137±0.009	0.121±0.005	0.125±0.006	0.113±0.011	0.119±0.006
ML^2	**0.101±0.007**	0.288±0.012	0.110±0.008	0.105±0.007	0.109±0.005	0.088±0.007	0.108±0.007
CLR	0.258±0.009	0.287±0.015	0.112±0.008	0.105±0.006	0.114±0.024	0.095±0.007	0.107±0.006
RAKEL	0.218±0.012	0.874±0.012	0.417±0.012	0.359±0.022	0.369±0.014	0.358±0.020	0.363±0.015
Comparing algorithms	Ranking loss ↓						
	slashdot	corel5k	rcv1subset1	rcv1subset2	rcv1subset3	rcv1subset4	rcv1subset5
MLFE	0.098±0.008	**0.109±0.006**	**0.039±0.003**	**0.039±0.002**	**0.040±0.002**	**0.034±0.003**	**0.038±0.002**
LIFT	0.092±0.008	0.122±0.005	0.053±0.003	0.059±0.002	0.062±0.002	0.051±0.004	0.055±0.002
RELIAB	0.118±0.005	0.130±0.005	0.058±0.003	0.045±0.002	0.052±0.002	0.047±0.004	0.048±0.002
ML^2	**0.084±0.006**	0.163±0.008	0.042±0.003	0.043±0.003	0.046±0.003	0.037±0.003	0.043±0.003
CLR	0.245±0.010	0.131±0.008	0.048±0.002	0.046±0.002	0.054±0.020	0.044±0.002	0.046±0.003
RAKEL	0.198±0.013	0.586±0.011	0.233±0.007	0.209±0.012	0.222±0.009	0.224±0.013	0.209±0.012
Comparing algorithms	Average precision ↑						
	slashdot	corel5k	rcv1subset1	rcv1subset2	rcv1subset3	rcv1subset4	rcv1subset5
MLFE	0.715±0.015	**0.316±0.012**	0.615±0.010	0.622±0.006	0.614±0.013	0.640±0.013	0.626±0.007
LIFT	0.695±0.019	0.291±0.010	0.582±0.013	0.579±0.008	0.569±0.010	0.596±0.010	0.586±0.009
RELIAB	0.610±0.012	0.269±0.009	0.563±0.010	0.588±0.009	0.586±0.013	0.611±0.010	0.586±0.009
ML^2	**0.728±0.015**	0.315±0.013	**0.629±0.013**	**0.630±0.008**	**0.622±0.011**	**0.647±0.014**	**0.631±0.006**
CLR	0.261±0.006	0.247±0.009	0.564±0.012	0.579±0.011	0.554±0.050	0.589±0.013	0.576±0.012
RAKEL	0.516±0.012	0.120±0.007	0.391±0.009	0.429±0.010	0.423±0.010	0.431±0.011	0.421±0.012
Comparing algorithms	Macro-averaging F1 ↑						
	slashdot	corel5k	rcv1subset1	rcv1subset2	rcv1subset3	rcv1subset4	rcv1subset5
MLFE	**0.455±0.048**	**0.338±0.014**	0.282±0.024	0.263±0.023	0.243±0.021	0.290±0.036	0.265±0.024
LIFT	0.427±0.036	0.324±0.014	0.219±0.038	0.163±0.020	0.151±0.020	0.203±0.034	0.165±0.022
RELIAB	0.433±0.047	0.303±0.019	**0.332±0.026**	**0.332±0.023**	**0.333±0.022**	**0.335±0.039**	**0.332±0.012**
ML^2	0.424±0.050	0.331±0.015	0.228±0.025	0.225±0.020	0.216±0.019	0.263±0.037	0.232±0.020
CLR	0.165±0.035	0.276±0.015	0.278±0.028	0.269±0.016	0.255±0.035	0.297±0.023	0.286±0.013
RAKEL	0.363±0.033	0.257±0.013	0.266±0.029	0.237±0.024	0.243±0.023	0.256±0.020	0.255±0.016
Comparing algorithms	Micro-averaging F1 ↑						
	slashdot	corel5k	rcv1subset1	rcv1subset2	rcv1subset3	rcv1subset4	rcv1subset5
MLFE	**0.550±0.016**	0.177±0.015	0.411±0.011	0.411±0.011	0.397±0.016	0.426±0.013	0.414±0.010
LIFT	0.509±0.020	0.077±0.011	0.311±0.020	0.297±0.013	0.289±0.014	0.327±0.019	0.314±0.012
RELIAB	0.449±0.014	**0.226±0.010**	**0.417±0.007**	**0.468±0.011**	**0.431±0.013**	**0.485±0.009**	**0.466±0.009**
ML^2	0.531±0.015	0.126±0.016	0.378±0.016	0.383±0.014	0.377±0.017	0.411±0.012	0.394±0.015
CLR	0.008±0.003	0.123±0.019	0.361±0.008	0.356±0.015	0.338±0.029	0.365±0.017	0.368±0.014
RAKEL	0.362±0.014	0.135±0.009	0.341±0.008	0.337±0.008	0.335±0.012	0.349±0.010	0.350±0.010

precision, and two label-based metrics *macro-averaging F1*, *micro-averaging F1*. These evaluation metrics consider the performance of multi-label predictor from various aspects, whose values all vary between [0,1]. Concrete metric definitions can be found in (Zhang and Zhou 2014), and the *coverage* metric is normalized by the number of class labels (i.e. q). For *one-error*, *coverage* and *ranking loss*, the *smaller* the values the better the performance. For the other three metrics, the *larger* the values the better the performance. Ten-fold cross-validation is performed on the benchmark data sets, where the mean metric value as well as standard deviation are recorded for each comparing algorithm.

Experimental Results

Table 2 and 3 report the detailed experimental results of six comparing algorithm on the regular-scale and large-scale data sets respectively, where the best performance among the comparing algorithms is shown in boldface. For each evaluation metric, "↓" indicates "the smaller the better" while "↑" indicates "the larger the better".

In this paper, *Friedman test* (Demšar 2006) is used as the statistical test to analyze the relative performance among the comparing algorithms. Table 4 summarizes the Friedman statistics F_F and the corresponding critical value on each evaluation metric. For each evaluation metric, the null hypothesis of indistinguishable performance among the comparing algorithms is rejected at 0.05 significance level. Consequently, the post-hoc *Bonferroni-Dunn test* (Demšar 2006) is employed to show the relative performance among the comparing algorithms. Here, MLFE is treated as the control algorithm whose average rank difference against the comparing algorithm is calibrated with the *critical difference*

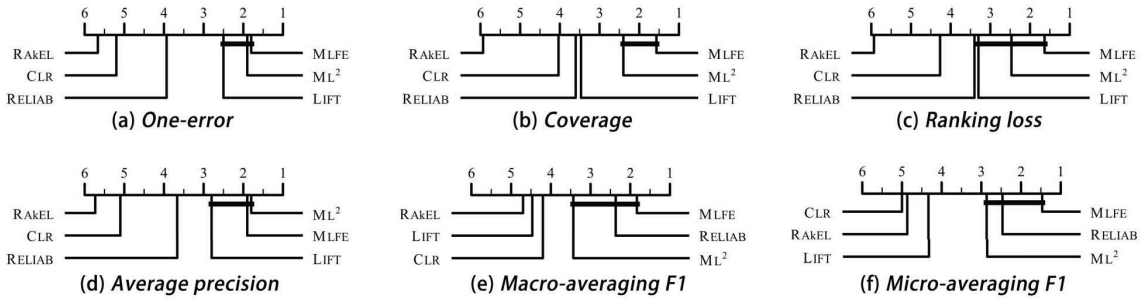

Figure 1: Comparison of MLFE (control algorithm) against five comparing algorithms with the *Bonferroni-Dunn test*. Algorithms not connected with MLFE in the CD diagram are considered to have significantly different performance from the control algorithm (CD=1.7597 at 0.05 significance level).

(CD). Accordingly, MLFE is deemed to have significantly different performance to one comparing algorithm if their average ranks differ by at least one CD (CD=1.7597 in this paper: # comparing algorithms $k = 6$, # data sets $N = 15$).

Figure 1 illustrates the CD diagrams (Demšar 2006) on each evaluation metric, where the average rank of each comparing algorithm is marked along the axis (lower ranks to the right). In each subfigure, any comparing algorithm whose average rank is within one CD to that of MLFE is interconnected to each other with a thick line. Overall, the following observations can be made based on the above experimental results:

- On regular-scale data sets (Table 2), across all evaluation metrics, MLFE ranks *1st* in 68.7% cases and ranks *2nd* in 14.5% cases; On large-scale data sets (Table 3), across all evaluation metrics, MLFE ranks *1st* in 38.0% cases and ranks *2nd* in 45.2% cases.
- It is noteworthy that MLFE achieves optimal (lowest) average rank in terms of all evaluation metrics except *average precision*. Furthermore, no algorithm significantly outperforms MLFE across all evaluation metrics.
- MLFE significantly outperforms CLR and RAKEL in terms of all evaluation metrics.
- MLFE is comparable to LIFT in terms of *one-error*, *ranking loss*, *average precision*, comparable to RELIAB in terms of *macro-averaging F1*, *micro-averaging F1*, and significantly outperforms LIFT and RELIAB on all the other cases.

Further Analysis

To further investigate the usefulness of the enriched labeling information generated by MLFE, experimental studies on artificial data with ground-truth numerical labeling information are conducted. Specifically, following the same experimental scheme in (Geng 2016), an artificial multi-label data set with 2,601 examples and ground-truth labeling on three class labels is generated. The enriched labeling information generated by MLFE is normalized and its distance with the ground-truth labeling information is measured by several widely-used dissimilarity criteria.

Table 4: Friedman statistics F_F in terms of each evaluation metric and the critical value at 0.05 significance level (# comparing algorithms $k = 6$, # data sets $N = 15$).

Evaluation metric	F_F	critical value
One-error	62.0069	2.3456
Coverage	24.9330	
Ranking loss	24.4755	
Average precision	48.7117	
Macro-averaging F1	9.2395	
Micro-averaging F1	20.5828	

Table 5: Quantitative measures of the dissimilarity between generated and ground-truth labeling information (ranks on each criterion shown in brackets).

Criterion	MLFE	ML^2	KM	FCM	RANDOM
Canberra ↓	0.8064(2)	0.8442(3)	1.5887(5)	0.7851(1)	1.2085(4)
Chebyshev ↓	0.1403(1)	0.1519(2)	0.1974(3)	0.2010(4)	0.3559(5)
Clark ↓	0.6099(2)	0.6278(3)	1.1805(5)	0.5293(1)	0.7817(4)
KL ↓	0.1242(1)	0.1341(2)	0.2264(3)	0.4091(4)	0.6807(5)
Intersection ↑	0.8596(1)	0.8481(2)	0.8026(3)	0.7990(4)	0.6441(5)
Cosine ↑	0.9661(1)	0.9609(2)	0.9471(3)	0.9294(4)	0.7550(5)
Average Rank	1.333	2.333	3.666	3.000	4.666

Table 5 reports the quantitative measures of the dissimilarity between generated and ground-truth labeling information, which clearly shows the strong capability of MLFE in discovering useful enriched labeling information w.r.t. the ML^2 (Hou, Geng, and Zhang 2016), KM (k-means) (Jiang, Zhang, and Lv 2006), FCM (fuzzy c-means) (Klir and Yuan 1995), and RANDOM approaches which can also generate numerical labeling scores.

Conclusion

In this paper, a novel approach is proposed to learning from multi-label data by leveraging the structural information in feature space. The key strategy is to convey the structural information modeled by sparse reconstruction in feature space

to facilitate generating enriched labeling information in output space. The effectiveness of feature-induced labeling information enrichment is clearly validated with extensive experiments on benchmark multi-label data sets.

Acknowledgements

This work was completed while Qian-Wen Zhang is conducting her internship at Tencent Smart Platform & Products Department. The authors wish to thank the anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Science Foundation of China (61573104), the Fundamental Research Funds for the Central Universities (2242017K40140), and partially supported by the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

Bertsekas, D. P., and Tsitsiklis, J. N. 1989. *Parallel and distributed computation: Numerical methods*. Upper Saddle River, NJ: Prentice Hall.

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

Cheng, W.; Dembczyński, K.; and Hüllermeier, E. 2010. Graded multilabel classification: The ordinal case. In *Proceedings of the 27th International Conference on Machine Learning*, 223–230.

Chung, W.; Kim, J.; Lee, H.; and Kim, E. 2015. General dimensional multiple-output support vector regressions and their multiple kernel learning. *IEEE Transactions on Cybernetics* 45(11):2572–2584.

Demšar, J. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* 7(Jan):1–30.

Elisseeff, A., and Weston, J. 2002. A kernel method for multi-labelled classification. In Dietterich, T. G.; Becker, S.; and Ghahramani, Z., eds., *Advances in Neural Information Processing Systems 14*. Cambridge, MA: MIT Press. 681–687.

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

Geng, X. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering* 28(7):1734–1748.

Ghadimi, E.; Teixeira, A.; Shames, I.; and Johansson, M. 2015. Optimal parameter selection for the alternating direction method of multipliers (admm): Quadratic problems. *IEEE Transactions on Automatic Control* 60(3):644–658.

Gibaja, E., and Ventura, S. 2015. A tutorial on multilabel learning. *ACM Computing Surveys* 47(3):Article 52.

Hou, P.; Geng, X.; and Zhang, M.-L. 2016. Multi-label manifold learning. In *Proceedings of the 30th AAAI Conference on Aritificial Intelligence*, 1680–1686.

Ji, S.; Tang, L.; Yu, S.; and Ye, J. 2010. A shared-subspace learning framework for multi-label classification. *ACM Transactions on Knowledge Discovery from Data* 4(2):Article 8.

Jiang, X.; Zhang, Y.; and Lv, J. C. 2006. Fuzzy svm with a new fuzzy membership function. *Neural Computing & Applications* 15(3-4):268–276.

Klir, G., and Yuan, B. 1995. *Fuzzy sets and fuzzy logic*. Upper Saddle River, NJ: Prentice Hall.

Li, Y.-K.; Zhang, M.-L.; and Geng, X. 2015. Leveraging implicit relative labeling-importance information for effective multi-label learning. In *Proceedings of the 15th IEEE International Conference on Data Mining*, 251–260.

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

Sánchez-Fernández, M.; de Prado-Cumplido, M.; Arenas-García, J.; and Pérez-Cruz, F. 2004. SVM multiregression for nonliear channel estimation in multiple-input multiple-output systems. *IEEE Transactions on Signal Processing* 52(8):2298–2307.

Schölkopf, B., and Smola, A. J. 2001. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. Camridge, MA: MIT Press.

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

Tuia, D.; Verrelst, J.; Alonso, L.; Pérez-Cruz, F.; and Camps-Valls, G. 2011. Multioutput support vector regression for remote sensing biophysical parameter estimation. *IEEE Geoscience and Remote Sensing Letters* 8(4):804–808.

Xu, M.; Li, Y.-F.; and Zhou, Z.-H. 2013. Multi-label learning with PRO loss. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, 998–1004.

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

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

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

Zhou, Z.-H., and Zhang, M.-L. 2017. Multi-label learning. In Sammut, C., and Webb, G. I., eds., *Encyclopedia of Machine Learning and Data Mining*. Berlin: Springer. 1–8.