基于标记密度分类间隔面的组类属属性学习

王一宾[①②] 裴根生[①] 程玉胜*[①②]

[①](安庆师范大学计算机与信息学院 安庆 246011)

[②](安徽省高校智能感知与计算重点实验室 安庆 246011)

摘 要：类属属性学习避免相同属性预测全部标记，是一种提取各标记独有属性进行分类的一种框架，在多标记学习中得到广泛的应用。而针对标记维度较大、标记分布密度不平衡等问题，已有的基于类属属性的多标记学习算法普遍时间消耗大、分类精度低。为提高多标记分类性能，该文提出一种基于标记密度分类间隔面的组类属属性学习(GLSFL-LDCM)方法。首先，使用余弦相似度构建标记相关性矩阵，通过谱聚类将标记分组以提取各标记组的类属属性，减少计算全部标记类属属性的时间消耗。然后，计算各标记密度以更新标记空间矩阵，将标记密度信息加入原标记中，扩大正负标记的间隔，通过标记密度分类间隔面的方法有效解决标记分布密度不平衡问题。最后，通过将组类属属性和标记密度矩阵输入极限学习机以得到最终分类模型。对比实验充分验证了该文所提算法的可行性与稳定性。

关键词：多标记分类；标记密度；组类属属性；极限学习机；分类间隔面

中图分类号：TP391 **文献标识码**：A **文章编号**：1009-5896(2020)05-1179-09

DOI: 10.11999/JEIT190343

Group-Label-Specific Features Learning Based on Label-Density Classification Margin

WANG Yibin[①②] PEI Gensheng[①] CHENG Yusheng[①②]

[①](*School of Computer and Information, Anqing Normal University, Anqing* 246011, *China*)

[②](*The University Key Laboratory of Intelligent Perception and Computing of Anhui Province, Anqing* 246011, *China*)

Abstract: The label-specific features learning avoids the same features prediction for all class labels, it is a kind of framework for extracting the specific features of each label for classification, so it is widely used in multi-label learning. For the problems of large label dimension and unbalanced label distribution density, the existing multi-label learning algorithm based on label-specific features has larger time consumption and lower classification accuracy. In order to improve the performance of classification, a Group-Label-Specific Features Learning method based on Label-Density Classification Margin (GLSFL-LDCM) is proposed. Firstly, the cosine similarity is used to construct the label correlation matrix, and the class labels are grouped by spectral clustering to extract the label-specific features of each label group to reduce the time consumption for calculating the label-specific features of all class labels. Then, the density of each label is calculated to update the label space matrix, the label-density information is added to the original label space. The classification margin between the positive and negative labels is expanded, thus the imbalance label distribution density problem is effectively solved by the method of label-density classification margin. Finally, the final classification model is obtained by inputting the group-label-specific features and the label-density matrix into the extreme learning machine. The comparison experiment results verify fully the feasibility and stability of the proposed algorithm.

Key words: Multi-label classification; Label-density; Group-label-specific features; Extreme learning machine; Classification margin

收稿日期：2019-05-18；改回日期：2019-09-30；网络出版：2020-01-29

*通信作者： 程玉胜 chengyshaq@163.com

基金项目：安徽省自然科学基金

Foundation Item: The Natural Science Foundation of Anhui Province

1 引言

多标记学习更加契合现实世界对象存在的多义性，并成功应用于图像识别[1–3]、文本分类[4,5]、生物学习[5,6]和情感分析[7,8]等领域。但目前多标记学习仍存在3点主要挑战。其一，在单标记学习中各示例类标记互斥，而对于多标记学习则各标记之间存在某些相关性。其二，通常多标记学习中数据特征具有高维性，而高维数据会导致“维数灾难”。其三，在多标记学习中某个标记可能只与其独有的特征关联。

对于多标记分类，学者们提出了大量学习算法[9,10]。虽然此类多标记学习算法取得了不错的效果，但标记预测由相同的属性决定并不合理。例如，图像识别任务中，“蓝天”、“白云”等标记与颜色属性关系密切，而局部纹理属性则并不敏感；在区分病患是否患有糖尿病时，血常规检测中血糖指标具有决定区分度，但却无法区别该病患的性别。而这类与标记相关程度大、判别能力强的属性称之为类属属性[11]。为提高属性选择的有效性和高效性，学者们已提出了多种基于类属属性的多标记学习算法[12–17]。其中部分算法[12–14]在抽取各标记类属属性时时间消耗较大，主要原因在于这些算法需将全部标记的类属属性通过聚类方式提取，但当标记数较多时该类算法提取类属属性所需的时间则会大幅增加。为此，文献[15–17]则采用嵌入式模型，通过L1范数将模型的系数矩阵稀疏化，以此来进行类属属性学习，有效降低了时间消耗。但此类嵌入式类属属性学习算法在分类过程中并未实际使用类属属性信息，需利用稀疏化后的系数矩阵来提取真实类属属性。

基于以上分析，为了更快速提取多标记类属属性且在分类时考虑标记间的内部联系，本文首次提出基于标记密度分类间隔面的组类属属性学习方法(GLSFL-LDCM)，采用组类属属性代替原类属属性提取方法，避免提取全部标记的类属属性，有效降低算法的时间消耗。通过使用余弦相似度构建标记相关性矩阵，采取谱聚类将原标记分组以提取组标记类属属性，同时采取标记密度这一先验知识添加到原标记空间。将组类属属性和对应标记密度空间输入极限学习机[18,19]，得出最后的分类模型。在多个多标记基准数据集中，与多种先进的类属属性学习算法进行对比，实验结果和统计分析进一步验证了算法的有效性。

2 标记密度分类间隔面和组类属属性多标记学习

2.1 组类属属性学习

针对传统类属属性学习中，考虑各标记的类属属性造成的时间消耗大，同时未充分考虑标记间相关性。本文采用组类属属性来处理多标记属性学习和标记相关性分析。

给定多标记数据有N个示例，表示为$D=[\boldsymbol{X}, \boldsymbol{Y}]$，其中$\boldsymbol{X}$为$d$维的特征空间$\boldsymbol{R}^d$，$\boldsymbol{Y}$为$m$类标记空间，则$N$个示例数据$D=\{(\boldsymbol{x}_1, \boldsymbol{Y}_1), (\boldsymbol{x}_2, \boldsymbol{Y}_2), \cdots, (\boldsymbol{x}_N, \boldsymbol{Y}_N)\}$，其中$\boldsymbol{x}_i\in\boldsymbol{X}$为一个示例，$\boldsymbol{y}_i\in\boldsymbol{Y}$是一组标记集合$\{y_1^{(i)}, y_2^{(i)}, \cdots, y_m^{(i)}\}$，最终多标记分类的映射关系$f:\boldsymbol{X}\to 2^{\boldsymbol{Y}}$。

为符合多标记数据特点，考虑标记空间在方向上的差异，使用余弦相似度构建标记空间$\boldsymbol{Y}$的相关性矩阵$\mathbf{LC}$，提取正相关性和负相关性。余弦相似度矩阵表示为

$$\mathrm{LC}_{j,k}=\sum_{i=1}^{N} y_j^{(i)} y_k^{(i)} \Big/ \left(\sqrt{\sum_{i=1}^{N}\left(y_j^{(i)}\right)^2}\sqrt{\sum_{i=1}^{N}\left(y_k^{(i)}\right)^2}\right) \tag{1}$$

为了去除较弱标记间相关性，在式(1)中设置阈值ε加以限制，

$$\mathrm{LC}_{j,k}=\begin{cases}\mathrm{LC}_{j,k}, & |\mathrm{LC}_{j,k}|>\varepsilon\\ 0, & |\mathrm{LC}_{j,k}|\le\varepsilon\end{cases} \tag{2}$$

其中，$\mathbf{LC}$中正数表示两对标记成正相关，负数则表示两对标记成负相关，数值的绝对值越大表示相关性越大。构建Laplace矩阵$\boldsymbol{L}$

$$\boldsymbol{L}=\boldsymbol{M}-\boldsymbol{W} \tag{3}$$

其中，矩阵$\boldsymbol{M}$为度矩阵，即$M_{j,j}=\sum_k|\mathrm{LC}_{j,k}|$；矩阵$\boldsymbol{W}$为邻接矩阵，即$W_{j,k}=\{|\mathrm{LC}_{j,k}|\}_{j,k}^{m}$。由式(3)可得Laplace矩阵$\boldsymbol{L}$。将求得的$\boldsymbol{L}$矩阵计算特征向量，取最大$K$个特征值向量构成矩阵$\boldsymbol{V}=[\boldsymbol{v}_1,\boldsymbol{v}_2, \cdots, \boldsymbol{v}_K]$，再将$\boldsymbol{V}$归一化。最后使用$k$-means将$\boldsymbol{V}$聚类，则第$i$个示例的标记$y_j^{(i)}\in\{1, 2, \cdots, K\}$, j=1, 2, $\cdots$, m。

上述通过谱聚类[20]将原标记空间$\boldsymbol{Y}$分成K组，表示为$\boldsymbol{G}=[G_1,G_2, \cdots, G_K]\in\boldsymbol{Y}$。本文采用线性Lasso回归对于标记组进行类属属性提取，目标函数表示为

$$\min_{\boldsymbol{P}}\frac{1}{2}\|\boldsymbol{X}\boldsymbol{P}-\boldsymbol{G}\|_F^2+\alpha\|\boldsymbol{P}\|_1 \tag{4}$$

其中，$\boldsymbol{P}$为稀疏权值矩阵，而α系数控制矩阵$\boldsymbol{P}$稀疏程度，且由于$\boldsymbol{G}$分为K组，所以$\boldsymbol{P}=[p_1, p_2, \cdots, p_K]\in\boldsymbol{R}^{d\times K}$。根据$\boldsymbol{P}$矩阵中对应位置数值进行属性选择，若为0删除该属性，若不为0则保留该属性。

由于求解$\boldsymbol{P}$矩阵即求线性Lasso优化问题存在不可导点，本文采用交替方向乘子法(Alternating Direction Method of Multipliers, ADMM)[21]求解该问题，首先将式(4)更换成全局一致性优化问题

$$\left.\begin{aligned}&\min_{\boldsymbol{P}}\frac{1}{2}\|\boldsymbol{XP}-\boldsymbol{G}\|_{\mathrm{F}}^2+\alpha\|\boldsymbol{Z}\|_1\\&\text{s.t. }\boldsymbol{P}-\boldsymbol{Z}=0\end{aligned}\right\}\tag{5}$$

则式(5)的ADMM迭代更新公式为

$$\left.\begin{aligned}\boldsymbol{P}^{t+1}&=\arg\min_{\boldsymbol{P}}\left(\frac{1}{2}\|\boldsymbol{XP}-\boldsymbol{G}\|_{\mathrm{F}}^2+(\boldsymbol{S}^{\mathrm{T}})^t(\boldsymbol{P}-\boldsymbol{Z}^t)\right.\\&\quad\left.+\frac{u}{2}\left\|\boldsymbol{P}-\boldsymbol{Z}^t\right\|_{\mathrm{F}}^2\right)\\\boldsymbol{Z}^{t+1}&=\arg\min_{\boldsymbol{Z}}\left(\alpha\|\boldsymbol{Z}\|_1+\left(\left(-\boldsymbol{S}^{\mathrm{T}}\right)^t\boldsymbol{Z}\right.\right.\\&\quad\left.\left.+\frac{u}{2}\left\|\boldsymbol{P}^{t+1}-\boldsymbol{Z}\right\|_{\mathrm{F}}^2\right)\right)\\\boldsymbol{S}^{t+1}&=\boldsymbol{S}^t+\mu\left(\boldsymbol{P}^{t+1}-\boldsymbol{Z}^{t+1}\right)\end{aligned}\right\}\tag{6}$$

其中，$\boldsymbol{S}\in\boldsymbol{R}^{d\times K}$为$d$行$K$列的稀疏矩阵，通过$\boldsymbol{S}$矩阵即可提取出$K$组类属属性。

2.2 基于标记密度的分类间隔面

对于标记分布密度不平衡问题，通过标记密度这一先验知识，改造原标记空间的二元标记向量，本文提出了标记密度分类间隔面计算方法。将标记密度信息添加到原标记空间，使得原标记+1/–1变为连续型数值标记。这种变换可以将标记的分布不平衡问题通过与分割面0之间数值距离来表示，同时这种转换可以使得分类器更加容易分类各标记，提高分类精度。对于多标记数据，N个示例的标记空间$\boldsymbol{Y}=\{\boldsymbol{Y}_1,\ \boldsymbol{Y}_2,\ \cdots,\ \boldsymbol{Y}_N\}$，$\boldsymbol{Y}_i=\{y_1{}^{(i)},\ y_2{}^{(i)},\ \cdots,\ y_m{}^{(i)}\}$，$i$=1, 2, …, N。则标记密度表示为

$$\left.\begin{aligned}\mathrm{YD}_{\mathrm{pos}}(j)&=\sum_{i=1}^{N}(y_j^{(i)}=1)\Big/\sum_{i=1}^{N}\sum_{j=1}^{m}(y_j^{(i)}=1)\\\mathrm{YD}_{\mathrm{neg}}(j)&=\sum_{i=1}^{N}(y_j^{(i)}\neq 1)\Big/\sum_{i=1}^{N}\sum_{j=1}^{m}(y_j^{(i)}\neq 1)\end{aligned}\right\}\tag{7}$$

其中，$\mathrm{YD}_{\mathrm{pos}}$为正标记密度，$\mathrm{YD}_{\mathrm{neg}}$为负标记密度。最终原标记空间转换为标记密度空间为

$$\mathrm{YD}(j)=\begin{cases}\mathrm{YD}_{\mathrm{pos}}(j)+1, & y_j=+1\\-\mathrm{YD}_{\mathrm{neg}}(j)-1, & y_j=-1\end{cases}\tag{8}$$

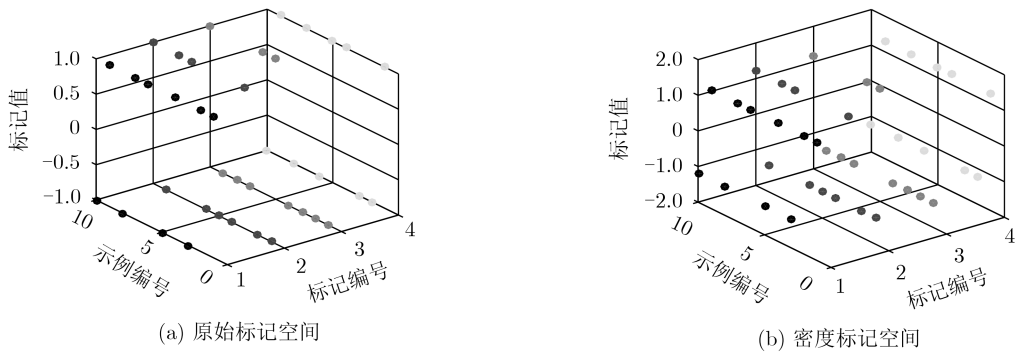
其中，**YD**即为最终的标记密度矩阵。

通过式(8)可以进一步发现，通过标记密度扩大各标记与分割面0之间的距离且以此来区别各标记。为了更加容易理解这个过程，本文给出一个虚拟标记空间数据，如表1所示。

将表1中数据可视化，原始标记空间如图1(a)所示，标记密度空间如图1(b)所示。在添加标记密度信息后的密度标记空间中，正标记与分类阈值面

表 1　标记空间虚拟数据集

标记编号	原标记				密度标记			
	Y_1	Y_2	Y_3	Y_4	Y_1	Y_2	Y_3	Y_4
1	+1	–1	–1	+1	+1.333	–1.273	–1.318	+1.278
2	+1	–1	–1	–1	+1.333	–1.273	–1.318	–1.227
3	–1	+1	–1	–1	–1.182	+1.222	–1.318	–1.227
4	+1	–1	–1	+1	+1.333	–1.273	–1.318	+1.278
5	–1	–1	+1	+1	–1.182	–1.273	+1.167	+1.278
6	+1	–1	+1	–1	+1.333	–1.273	+1.167	–1.227
7	+1	+1	–1	+1	+1.333	+1.222	–1.318	+1.278
8	–1	+1	–1	–1	–1.182	+1.222	–1.318	–1.227
9	+1	–1	–1	+1	+1.333	–1.273	–1.318	+1.278
10	–1	+1	+1	–1	–1.182	+1.222	+1.167	–1.227



(a) 原始标记空间　　(b) 密度标记空间

图 1 标记密度间隔曲面

“0”距离都大于“1”且各不相同，同理，负标记亦是如此。使用标记密度间隔曲面代替原先“+1”或“–1”标记面可以更好地区分各类标记，处理标记分布密度不平衡所导致的分类性能低等问题。

2.3 基于标记密度和组类属属性多标记分类

通过标记密度分类间隔面方法构建的标记密度矩阵为连续数值型标记，无法使用SVM进行分类处理；同时考虑算法稳定性(因随机初始化权值和偏置使得基础ELM算法不稳定)，因而采用核极限学习机[22](Kernel Extreme Learning Machine, KELM)作为最终分类器。但传统KELM适用于单标记和多类标问题，因此，结合ELM学习模型，本文构造了KELM的多标记分类器来解决多标记分类问题。

设多标记数据集$D=\{\boldsymbol{x}_i,\boldsymbol{Y}_i\}_{i=1}^{N}$，其中$\boldsymbol{x}_i\in\boldsymbol{R}^d$是$d$维特征向量，$\boldsymbol{Y}_i\in\boldsymbol{R}^m$为输出标记。随机映射函数$h(\boldsymbol{x}_i)$将$\boldsymbol{x}_i$从输入空间映射到$M$维的特征空间。

对于具有Q个隐藏节点的单隐藏层神经网络形式化定义为

$$\sum_{j=1}^{Q}\boldsymbol{\beta}_j g(\boldsymbol{x}_i)=\sum_{j=1}^{Q}\boldsymbol{\beta}_j g(\boldsymbol{w}_j\cdot\boldsymbol{x}_i+b_j)=o_i \tag{9}$$

在式(9)中，$\boldsymbol{\beta}_j=[\boldsymbol{\beta}_{j1},\boldsymbol{\beta}_{j2},\cdots,\boldsymbol{\beta}_{jm}]^{\mathrm{T}}$表示输出权值，$g(x)$为激活函数，$\boldsymbol{w}_j$=[$\boldsymbol{w}_{j1}$, $\boldsymbol{w}_{j2}$, $\cdots$,$\boldsymbol{w}_{jm}$]$^{\mathrm{T}}$为输入权值，b_j表示为第j个隐藏神经元的偏置，$\cdot$表示为点积。

而对于单隐藏层神经网络的目标是输出的误差最小，这样就可以表示为

$$\sum_{i=1}^{N}\|o_i-y_i\|=0 \tag{10}$$

通过式(9)和式(10)可知在确定β_j, $\boldsymbol{w}_j$和b_j时得到

$$\sum_{j=1}^{Q}\boldsymbol{\beta}_j g(\boldsymbol{w}_j\cdot\boldsymbol{x}_i+b_j)=y_i,i=1,2,\cdots,N \tag{11}$$

将式(11)用矩阵可以表示为

$$\boldsymbol{H}\boldsymbol{\beta}=\boldsymbol{Y} \tag{12}$$

根据式(9)和式(12)，可得多标记ELM的输出函数$\boldsymbol{f}_{ML}(\boldsymbol{x})$为

$$f_{ML}(\boldsymbol{x})=\boldsymbol{H}\boldsymbol{\beta}=\begin{bmatrix}h(\boldsymbol{x}_1)\\h(\boldsymbol{x}_2)\\\vdots\\h(\boldsymbol{x}_N)\end{bmatrix}_{N\times Q}\begin{bmatrix}\boldsymbol{\beta}_1^{\mathrm{T}}\\\boldsymbol{\beta}_2^{\mathrm{T}}\\\vdots\\\boldsymbol{\beta}_Q^{\mathrm{T}}\end{bmatrix} \tag{13}$$

根据岭回归理论[23]，添加L2正则以提高算法稳定性和泛化性能，同时有效避免过拟合。目标函数表示为

$$\left.\begin{aligned}&\min_{\boldsymbol{\beta}}\frac{1}{2}\|\boldsymbol{\beta}\|^2+\frac{C}{2}\sum_{i=1}^{N}\|\boldsymbol{\xi}_i\|^2\\&\text{s.t. }\boldsymbol{\xi}_i=Y_i-f_{ML}(\boldsymbol{x}_i),i=1,2,\cdots,N\end{aligned}\right\} \tag{14}$$

其中，C为正则化系数，可得输出权值β

$$\boldsymbol{\beta}=\boldsymbol{H}^{\mathrm{T}}\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}\right)^{-1}\boldsymbol{Y} \tag{15}$$

其中，$\boldsymbol{I}$为M维单位矩阵，这样多标记学习目标函数表示为

$$f_{ML}(\boldsymbol{x})=\boldsymbol{H}\boldsymbol{\beta}=\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}\right)^{-1}\boldsymbol{Y} \tag{16}$$

当映射函数$h(\boldsymbol{x})$未知时，即引入核矩阵

$$\left.\begin{aligned}&\boldsymbol{\Omega}_{\mathrm{ELM}}=\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}:\boldsymbol{\Omega}_{\mathrm{ELM}(i,j)}=K(\boldsymbol{x}_i,\boldsymbol{x}_j)\\&K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\exp(-\gamma\|x_i-x_j\|^2)\end{aligned}\right\} \tag{17}$$

结合式(8)、式(16)和式(17)，结合标记密度矩阵的多标记核极限学习机分类器目标函数定义为

$$f_{ML}(\boldsymbol{x})=\boldsymbol{H}\boldsymbol{\beta}=\boldsymbol{\Omega}_{\mathrm{ELM}}\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{\Omega}_{\mathrm{ELM}}\right)^{-1}\mathbf{YD} \tag{18}$$

通过将各组类属属性和对应标记密度矩阵输入$f_{ML}(\boldsymbol{x})$中，由式(6)得到类属属性提取矩阵$\boldsymbol{S}$，则在总共K组中第k组标记预测表示为

$$\left.\begin{aligned}&\boldsymbol{\Omega}_{\mathrm{ELM}}^{k}=\boldsymbol{\Omega}_{\mathrm{ELM}}(\boldsymbol{x}(:,\boldsymbol{S}^k\neq 0))\\&\mathbf{YD}^k=\mathbf{YD}(G_k)\\&\boldsymbol{\beta}^k=\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{\Omega}_{\mathrm{ELM}}^{k}\right)^{-1}\mathbf{YD}^k\end{aligned}\right\} \tag{19}$$

最终将K组预测结果合并，得到合并后的最终分类预测模型为

$$\left.\begin{aligned}&\boldsymbol{G}_k^*=\boldsymbol{\Omega}_{\mathrm{ELM}}(\boldsymbol{x}^*(:,\boldsymbol{S}^k\neq 0))\boldsymbol{\beta}^k\\&\boldsymbol{Y}^*=[G_1^*,\boldsymbol{G}_2^*,\cdots,\boldsymbol{G}_K^*]\end{aligned}\right\} \tag{20}$$

本文基于标记密度分类间隔面的组类属属性学习(GLSFL- LDCM)算法详细过程如表2所示。

3 实验方案及结果分析

3.1 实验数据描述

为验证本文算法，实验在8个公开基准多标记数据集进行。数据样本数从593～3782不等，特征数从72～1449不等，各数据集详细信息如表3所示。

3.2 实验方案及评价指标

为比较本文算法GLSFL-LDCM算法性能，实验采用ML-kNN[2], LIFT[12], FRS-LIFT[13], FRS-SS-LIFT[13]和LLSF-DL[17]等作为对比算法，使用10折交叉验证进行实验。其中，ML-kNN算法为经典多标记学习算法，其余4个算法都是基于类属属性的多标记学习算法。在对比实验中，各对比算法参数

表 2　GLSFL-LDCM算法步骤

输入：训练数据集$D=\{\boldsymbol{x}_i,\boldsymbol{Y}_i\}_{i=1}^{N}$，测试数据集$D^*=\left\{\boldsymbol{x}_j^*\right\}_{j=1}^{N^*}$，RBF核参数$\gamma$，惩罚因子$C$，类属属性参数：$\alpha$, β, $\boldsymbol{\mu}$，聚类数K；

输出：预测标记Y^*.

Training: training data set D

(1) 用式(1)、式(2)计算余弦相似度，构造标记相关性矩阵LC

(2) 用式(3)谱聚类将标记分组：$\boldsymbol{G}=[G_1,G_2,\cdots,G_K]$

(3) 用式(5)、式(6)构建类属属性提取矩阵$\boldsymbol{S}$

(4) 通过式(7)、式(8)更新标记空间，构造标记密度矩阵：$\boldsymbol{YD}$

(5) For $k=1, 2, \cdots, K$ do

$\boldsymbol{\Omega}_{\mathrm{ELM}}^k=\boldsymbol{\Omega}_{\mathrm{ELM}}(\boldsymbol{x}(:,\boldsymbol{S}^k\neq 0))$

$\mathbf{YD}^k=\mathbf{YD}(\boldsymbol{G}_k)$

$$\boldsymbol{\beta}^k=\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{\Omega}_{\mathrm{ELM}}^k\right)^{-1}\mathbf{YD}^k$$

Prediction: testing data set D^*

(a) For $k=1, 2, \cdots, K$ do

$\boldsymbol{G}_k^*=\boldsymbol{\Omega}_{\mathrm{ELM}}(\boldsymbol{x}^*(:,\boldsymbol{S}^k\neq 0))\boldsymbol{\beta}^k$

(b) $\boldsymbol{Y}^*=[\boldsymbol{G}_1^*,\boldsymbol{G}_2^*,...,\boldsymbol{G}_K^*]$

表 3　多标记数据描述

数据集	样本数	特征数	标记数	标记基数	应用领域
Emotions[1)]	593	72	6	1.869	MUSIC
Genbase[1)]	662	1186	27	1.252	BIOLOGY
Medical[1)]	978	1449	45	1.245	TEXT
Enron[3)]	1702	1001	53	4.275	TEXT
Image[2)]	2000	294	5	1.236	IMAGE
Scene[1)]	2407	294	6	1.074	IMAGE
Yeast[1)]	2417	103	14	4.237	BIOLOGY
Slashdot[3)]	3782	1079	22	0.901	TEXT

均按照原论文参数范围设置，ML-kNN近邻k=10，平滑系数s=1; LIFT, FRS-LIFT和FRS-SS-LIFT聚类比率在[0.1:0.1:1]之间，最终比率r设为0.2；由于LLSF-DL算法各数据集的参数相差很大，对于原论文提及数据集的参数给予设定，未提及的数据集均采用5折交叉验证取最好参数，其中参数α, β和γ的范围{4^{-5}, 4^{-4}, ⋯, 4^4, 4^5}, ρ取{0.1, 1, 10}；本文算法参数聚类数K=[1:1:10]之间，ε=0.01, α=1, $\boldsymbol{\mu}$取{0.1, 1, 10}，惩罚因子C和RBF核参数γ取值范围{1, 10, 100}。

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

[2)]http://cse.seu.edu.cn/PersonalPage/zhangml/files/Image.rar

[3)]http://waikato.github.io/meka/datasets

为有效验证算法性能，本文通过4种评价指标来综合衡量各算法的性能，评价指标包括：Hamming Loss, One-Error, Ranking Loss和Average Precision[2,4,24]，分别对应HL, OE, RL和AP。

3.3　实验结果及分析

在8个数据集中对比实验结果如表4所示，分别对应HL↓, OE↓, RL↓和AP↑指标的对比实验结果，包括各指标的均值和标准差。其中，“↑”表示指标数值越高越好，“↓”表示指标数值越低越好，各数据集中指标表现最优的用加粗字体表示。

同时，为了更加明确各算法性能差异，本文使用显著性水平5%的成对t检验[25]进行算法对比，并在表格中使用●/○表示本文算法GLSFL-LDCM优于/差于对比算法，未标识的表示与本文算法无显著性差异。最后在各评价中给出win/tie/loss，“win”表示GLSFL-LDCM优于该对比算法，“tie”表示无显著性差异，“loss”表示GLSFL-LDCM差于该对比算法。

表4实验结果表明：对于HL和OE指标，GLSFL-LDCM算法实验结果总体占优，但对于离散特征数据分类效果则并不理想。原因在于本文算法使用RBF作为分类算法核函数，离散特征导致映射后特征区分度降低。但对于连续特征数据，GLSFL-LDCM较其他5种对比算法均表现优秀。而对于RL和AP指标，GLSFL-LDCM在全部结果中均占优。

各算法在8个数据集的时间消耗如表5所示，其中，算法编号1～6分别表示：ML-kNN, LIFT, FRS-LIFT, FRS-SS-LIFT, LLSF-DL和GLSFL-LDCM。本文算法时间消耗均小于1 s。

为清晰展示各算法的性能表现，使用箱线图展示4个评价指标实验结果，如图2所示。其中，“+”为异常数据，表明算法并不稳定。通过图2可知，本文算法在HL, OE和RL评价指标中位数均小于其他对比算法，且上边缘和下边缘都较为理想；对于AP，本文算法中位数高于其他对比算法。

基于以上的实验结果和统计分析表明本文提出的GLSFL-LDCM算法在综合性能方面有较好的表现，提供了一种多标记组类属属性学习方法。

3.4　模型分解性实验及分析

为验证本文算法两个重要部分(组类属属性和标记密度分类间隔面)对实验结果的影响，同时鉴于本文算法采用KELM作为分类器，因此为了对比的公平性，将KELM与传统类属属性学习结合(LSFL-KELM)，KELM与组类属属性结合(GLSFL-KELM)以及KELM与标记密度分类间隔面结合(LDCM-KELM)。

表 4 对比算法实验结果

数据集	ML-kNN	LIFT	FRS-LIFT	FRS-SS-LIFT	LLSF-DL	GLSFL-LDCM
	HL ↓					
Emotions	0.1998±0.0167●	0.1854±0.0260●	0.1798±0.0290●	0.1809±0.0310●	0.2035±0.0082●	**0.1782±0.0154**
Genbase	0.0043±0.0017●	0.0011±0.0016●	0.0015±0.0009●	0.0017±0.0011●	0.0008±0.0014●	**0.0006±0.0005**
Medical	0.0158±0.0015●	0.0115±0.0013●	**0.0087±0.0014**○	0.0089±0.0013	0.0092±0.0004	0.0089±0.0021
Enron	0.0482±0.0043●	0.0365±0.0034○	**0.0341±0.0032**○	0.0372±0.0034○	0.0369±0.0034○	0.0468±0.0021
Image	0.1701±0.0141●	0.1567±0.0136●	0.1479±0.0103●	0.1468±0.0097●	0.1828±0.0152●	**0.1397±0.0133**
Scene	0.0852±0.0060●	0.0772±0.0047●	0.0740±0.0052●	0.0751±0.0057●	0.1008±0.0059●	**0.0682±0.0084**
Yeast	0.1934±0.0116●	0.1919±0.0083●	0.1875±0.0114●	0.1869±0.0111●	0.2019±0.0060●	**0.1855±0.0079**
Slashdot	0.0221±0.0010●	0.0159±0.0009○	0.0159±0.0011○	0.0160±0.0011○	**0.0158±0.0012**○	0.0196±0.0010
win/tie/loss	8/0/0	6/0/2	5/0/3	5/1/2	5/1/2	–
数据集	ML-kNN	LIFT	FRS-LIFT	FRS-SS-LIFT	LLSF-DL	GLSFL-LDCM
	OE ↓					
Emotions	0.2798±0.0441●	0.2291±0.0645●	**0.2155±0.0608**	0.2223±0.0651●	0.2583±0.0201●	0.2157±0.0507
Genbase	0.0121±0.0139●	0.0015±0.0047	0.0015±0.0047	0.0030±0.0094●	**0.0000±0.0000**○	0.0015±0.0048
Medical	0.2546±0.0262●	0.1535±0.0258●	**0.1124±0.0279**○	0.1186±0.0231○	0.1285±0.0271●	0.1226±0.0383
Enron	0.5158±0.0417●	0.4279±0.0456●	0.3084±0.0444●	0.3256±0.0437●	0.2704±0.0321●	**0.2221±0.0227**
Image	0.3195±0.0332●	0.2680±0.0256●	0.2555±0.0334●	0.2490±0.0226●	0.3180±0.0326●	**0.2365±0.0224**
Scene	0.2185±0.0313●	0.1924±0.0136●	0.1841±0.0156●	0.1836±0.0195●	0.2323±0.0267●	**0.1562±0.0316**
Yeast	0.2251±0.0284●	0.2177±0.0255●	0.2147±0.0171●	0.2085±0.0156●	0.2267±0.0239●	**0.2072±0.0250**
Slashdot	0.0946±0.0143●	0.0898±0.0134●	**0.0858±0.0162**○	0.0864±0.0138○	0.0887±0.0123●	0.0874±0.0107
win/tie/loss	8/0/0	7/1/0	4/2/2	6/0/2	7/0/1	–
数据集	ML-kNN	LIFT	FRS-LIFT	FRS-SS-LIFT	LLSF-DL	GLSFL-LDCM
	RL ↓					
Emotions	0.1629±0.0177●	0.1421±0.0244●	0.1401±0.0299●	0.1406±0.0280●	0.1819±0.0166●	**0.1375±0.0226**
Genbase	0.0062±0.0082●	0.0034±0.0065●	0.0043±0.0071●	0.0051±0.0077●	0.0071±0.0031●	**0.0017±0.0025**
Medical	0.0397±0.0093●	0.0262±0.0072●	0.0248±0.0108●	0.0236±0.0074●	0.0218±0.0080●	**0.0148±0.0096**
Enron	0.1638±0.0222●	0.1352±0.0190●	0.0953±0.0107●	0.1046±0.0099●	0.0927±0.0069●	**0.0735±0.0084**
Image	0.1765±0.0202●	0.1425±0.0169●	0.1378±0.0149●	0.1323±0.0171●	0.1695±0.0162●	**0.1294±0.0127**
Scene	0.0760±0.0100●	0.0604±0.0047●	0.0601±0.0061●	0.0592±0.0072●	0.0803±0.0133●	**0.0515±0.0093**
Yeast	0.1666±0.0149●	0.1648±0.0121●	0.1588±0.0150●	0.1560±0.0138●	0.1716±0.0145●	**0.1551±0.0100**
Slashdot	0.0497±0.0072●	0.0418±0.0062●	0.0289±0.0038●	0.0311±0.0038●	0.0307±0.0058●	**0.0126±0.0018**
win/tie/loss	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	–
数据集	ML-kNN	LIFT	FRS-LIFT	FRS-SS-LIFT	LLSF-DL	GLSFL-LDCM
	AP ↑					
Emotions	0.7980±0.0254●	0.8236±0.0334●	0.8280±0.0411●	0.8268±0.0400●	0.7504±0.0120●	**0.8316±0.0265**
Genbase	0.9873±0.0121●	0.9958±0.0078●	0.9944±0.0078●	0.9935±0.0085●	0.9928±0.0024●	**0.9962±0.0057**
Medical	0.8068±0.0248●	0.8784±0.0145●	0.9096±0.0176●	0.9087±0.0155●	0.9028±0.0172●	**0.9122±0.0281**
Enron	0.5134±0.0327●	0.5620±0.0321●	0.6611±0.0408●	0.6481±0.0287●	0.6632±0.0182●	**0.6923±0.0159**
Image	0.7900±0.0203●	0.8240±0.0169●	0.8314±0.0177●	0.8364±0.0162●	0.7943±0.0177●	**0.8444±0.0118**
Scene	0.8687±0.0164●	0.8884±0.0081●	0.8913±0.0084●	0.8921±0.0101●	0.8609±0.0182●	**0.9082±0.0173**
Yeast	0.7659±0.0194●	0.7685±0.0148●	0.7762±0.0172●	0.7790±0.0167●	0.7633±0.0160●	**0.7798±0.0140**
Slashdot	0.8835±0.0116●	0.8927±0.0091●	0.9045±0.0098●	0.9038±0.0074●	0.9017±0.0095●	**0.9247±0.0059**
win/tie/loss	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	–

表 5　各算法的时耗对比(s)

数据集	1	2	3	4	5	6
Emotions	0.2	0.4	54.0	8.7	0.1	0.1
Genbase	1.0	2.9	15.0	1.7	0.9	0.2
Medical	4.3	12.5	66.3	14.8	2.3	0.4
Enron	6.5	48.1	1292.7	182.7	0.6	0.6
Image	3.4	8.1	1805.2	320.5	0.1	0.2
Scene	5.4	7.9	2174.1	404.2	0.1	0.2
Yeast	3.5	44.3	13113.4	3297.7	0.2	0.3
Slashdot	34.1	84.5	11895.5	2650.0	1.1	0.8
平均	7.3	26.1	3802.0	860.0	0.7	0.4

通过在Emotions, Genbase, Medical和Scene等4个不同领域数据上进行对比实验，选用HL和AP作为评价指标，结果如表6所示。可以发现，LSFL-KELM较KELM分类精度有一定提高，表明类属属性学习可以提高多标记分类的性能；同时可以发现本文所提的基于标记密度分类间隔面的组类属属性学习方法(GLSFL-LDCM)在使用相同分类器KELM情况下，对比KELM和LSFL-KELM的分类精度有较为明显的提高，证明了本文算法的有效性。

为了进一步对比所提组类属属性与传统类属属

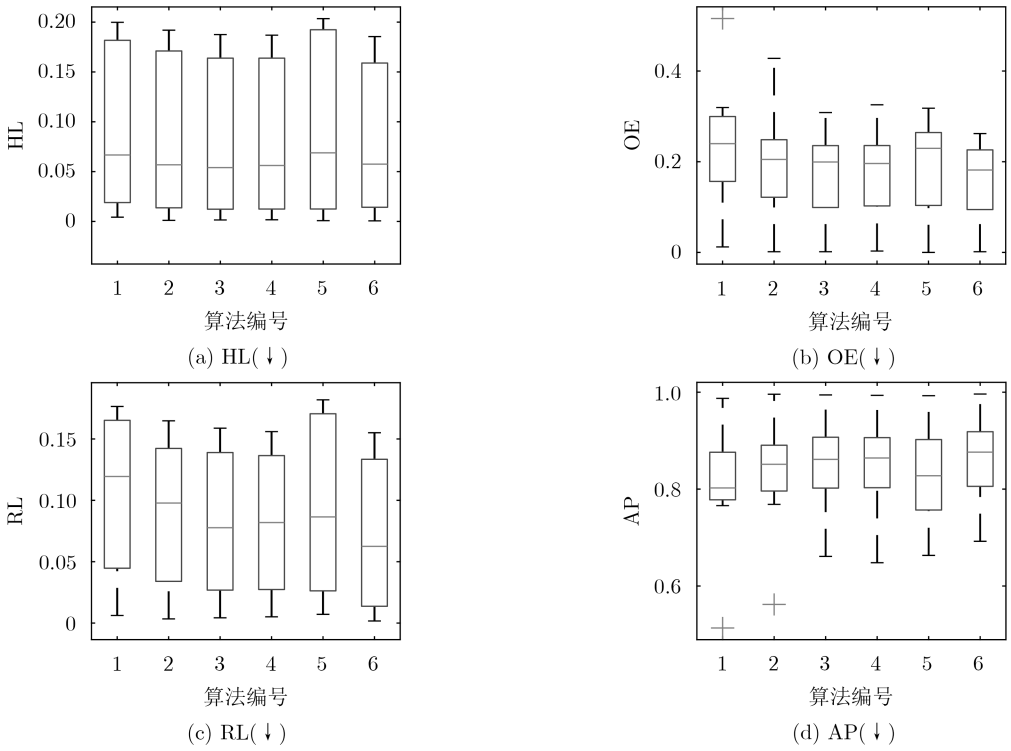


图 2 算法性能比较

表 6　模型分解对比实验

数据集	KELM	LSFL-KELM	GLSFL-KELM	LDCM-KELM
		HL↓		
Emotions	0.1840±0.0275	0.1837±0.0253	0.1824±0.0196	**0.1802±0.0295**
Genbase	0.0010±0.0008	0.0008±0.0005	**0.0006±0.0006**	0.0007±0.0006
Medical	0.0094±0.0030	0.0093±0.0017	**0.0091±0.0016**	0.0092±0.0019
Scene	0.0706±0.0051	0.0693±0.0079	0.0683±0.0059	**0.0682±0.0062**
数据集	KELM	LSFL-KELM	GLSFL-KELM	LDCM-KELM
		AP↑		
Emotions	0.8144±0.0369	0.8223±0.0252	0.8296±0.0278	**0.8306±0.0429**
Genbase	0.9926±0.0046	0.9928±0.0048	**0.9961±0.0046**	0.9956±0.0038
Medical	0.9077±0.0262	0.9092±0.0229	0.9124±0.0205	**0.9126±0.0306**
Scene	0.9010±0.0127	0.9024±0.0186	**0.9059±0.0132**	0.9033±0.0152

性的区别，图3给出了在Emotions和Scene数据集中类属属性提取系数矩阵。

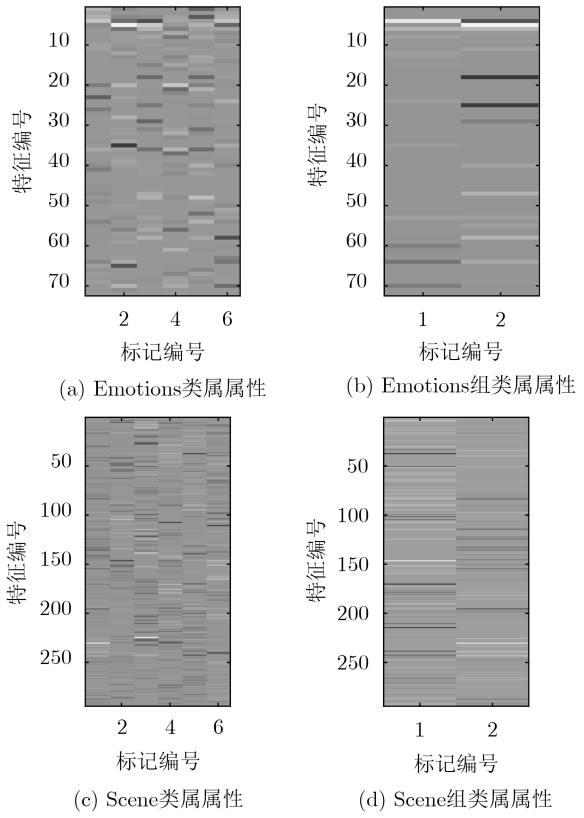


图 3 类属属性提取系数矩阵对比

通过图3可以发现，在Emotions和Scene数据集中，采用传统类属属性提取方法需将全部标记进行提取，而采用组类属属性仅提取组内的类属属性，因此可以有效降低算法的时间消耗。同时在分组过程中考虑标记之间的相关性，提高了多标记分类的精度，上述实验也证明了这一点。

4 结论

本文提出基于标记密度分类间隔面的组类属属性学习方法，通过标记密度先验知识，控制各标记与分割面的距离，提高了分类的精度；同时，使用谱聚类将标记分组，采用组类属属性进行分类，在考虑了标记相关性的同时进一步减少了时间消耗。对比目前性能先进的类属属性多标记分类算法，本文提出的算法在分类精度上和时间消耗上都有不错的表现，多组实验结果也证明该算法的稳定性和可靠性。

参 考 文 献

[1] ZHANG Minling and ZHOU Zhihua. ML-KNN: A lazy learning approach to multi-label learning[J]. *Pattern Recognition*, 2007, 40(7): 2038–2048. doi: 10.1016/j.patcog.2006.12.019.

[2] LIU Yang, WEN Kaiwen, GAO Quanxue, *et al*. SVM based multi-label learning with missing labels for image annotation[J]. *Pattern Recognition*, 2018, 78: 307–317. doi: 10.1016/j.patcog.2018.01.022.

[3] ZHANG Junjie, WU Qi, SHEN Chunhua, *et al*. Multilabel image classification with regional latent semantic dependencies[J]. *IEEE Transactions on Multimedia*, 2018, 20(10): 2801–2813. doi: 10.1109/TMM.2018.2812605.

[4] AL-SALEMI B, AYOB M, and NOAH S A M. Feature ranking for enhancing boosting-based multi-label text categorization[J]. *Expert Systems with Applications*, 2018, 113: 531–543. doi: 10.1016/j.eswa.2018.07.024.

[5] ZHANG Minling and ZHOU Zhihua. Multilabel neural networks with applications to functional genomics and text categorization[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2006, 18(10): 1338–1351. doi: 10.1109/TKDE.2006.162.

[6] GUAN Renchu, WANG Xu, YANG M Q, *et al*. Multi-label deep learning for gene function annotation in cancer pathways[J]. *Scientific Reports*, 2018, 8: No. 267. doi: 10.1038/s41598-017-17842-9.

[7] SAMY A E, EL-BELTAGY S R, and HASSANIEN E. A context integrated model for multi-label emotion detection[J]. *Procedia Computer Science*, 2018, 142: 61–71. doi: 10.1016/j.procs.2018.10.461.

[8] ALMEIDA A M G, CERRI R, PARAISO E C, *et al*. Applying multi-label techniques in emotion identification of short texts[J]. *Neurocomputing*, 2018, 320: 35–46. doi: 10.1016/j.neucom.2018.08.053.

[9] TSOUMAKAS G and KATAKIS I. Multi-label classification: An overview[J]. *International Journal of Data Warehousing and Mining*, 2007, 3(3): No. 1. doi: 10.4018/jdwm.2007070101.

[10] ZHANG Minling and ZHOU Zhihua. A review on multi-label learning algorithms[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2014, 26(8): 1819–1837. doi: 10.1109/TKDE.2013.39.

[11] CRAMMER K, DREDZE M, GANCHEV K, *et al*. Automatic code assignment to medical text[C]. Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing, Stroudsburg, USA, 2007: 129–136.

[12] ZHANG Minling and WU Lei. Lift: Multi-label learning with label-specific features[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(1): 107–120. doi: 10.1109/TPAMI.2014.2339815.

[13] XU Suping, YANG Xibei, YU Hualong, *et al*. Multi-label learning with label-specific feature reduction[J]. *Knowledge-Based Systems*, 2016, 104: 52–61. doi: 10.1016/j.knosys.2016.04.012.

[14] SUN Lu, KUDO M, and KIMURA K. Multi-label

classification with meta-label-specific features[C]. 2016 IEEE International Conference on Pattern Recognition, Cancun, Mexico, 2016: 1612–1617. doi: 10.1109/ICPR.2016.7899867.

[15] HUANG Jun, LI Guorong, HUANG Qingming, *et al*. Joint feature selection and classification for multilabel learning[J]. *IEEE Transactions on Cybernetics*, 2018, 48(3): 876–889. doi: 10.1109/TCYB.2017.2663838.

[16] WENG Wei, LIN Yaojin, WU Shunxiang, *et al*. Multi-label learning based on label-specific features and local pairwise label correlation[J]. *Neurocomputing*, 2018, 273: 385–394. doi: 10.1016/j.neucom.2017.07.044.

[17] HUANG Jun, LI Guorong, HUANG Qingming, *et al*. Learning label-specific features and class-dependent labels for multi-label classification[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2016, 28(12): 3309–3323. doi: 10.1109/TKDE.2016.2608339.

[18] HUANG Guangbin, ZHU Qinyu, and SIEW C K. Extreme learning machine: Theory and applications[J]. *Neurocomputing*, 2006, 70(1/3): 489–501. doi: 10.1016/j.neucom.2005.12.126.

[19] HUANG Guangbin, ZHOU Hongming, DING Xiaojian, *et al*. Extreme learning machine for regression and multiclass classification[J]. *IEEE Transactions on Systems*, *Man*, *and Cybernetics*, *Part B* (*Cybernetics*), 2012, 42(2): 513–529. doi: 10.1109/TSMCB.2011.2168604.

[20] 赵小强, 刘晓丽. 基于公理化模糊子集的改进谱聚类算法[J]. 电子与信息学报, 2018, 40(8): 1904–1910. doi: 10.11999/JEIT170904.

ZHAO Xiaoqiang and LIU Xiaoli. An improved spectral clustering algorithm based on axiomatic fuzzy set[J]. *Journal of Electronics & Information Technology*, 2018, 40(8): 1904–1910. doi: 10.11999/JEIT170904.

[21] BOYD S, PARIKH N, CHU E, *et al*. Distributed optimization and statistical learning via the alternating direction method of multipliers[J]. *Foundations and Trends® in Machine learning*, 2010, 3(1): 1–122. doi: 10.1561/2200000016.

[22] LIU Xinwang, WANG Lei, HUANG Guangbin, *et al*. Multiple kernel extreme learning machine[J]. *Neurocomputing*, 2015, 149: 253–264. doi: 10.1016/j.neucom.2013.09.072.

[23] 邓万宇, 郑庆华, 陈琳, 等. 神经网络极速学习方法研究[J]. 计算机学报, 2010, 33(2): 279–287. doi: 10.3724/SP.J.1016.2010.00279.

DENG Wanyu, ZHENG Qinghua, CHEN Lin, *et al*. Research on extreme learning of neural networks[J]. *Chinese Journal of Computers*, 2010, 33(2): 279–287. doi: 10.3724/SP.J.1016.2010.00279.

[24] ZHOU Zhihua, ZHANG Minling, HUANG Shengjun, *et al*. Multi-instance multi-label learning[J]. *Artificial Intelligence*, 2012, 176(1): 2291–2320. doi: 10.1016/j.artint.2011.10.002.

[25] PAPINENI K, ROUKOS S, WARD T, *et al*. BLEU: A method for automatic evaluation of machine translation[C]. The 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, USA, 2002: 311–318. doi: 10.3115/1073083.1073135.

王一宾：男，1970年生，教授，研究方向为多标记学习，机器学习，软件安全等.

裴根生：男，1992年生，硕士，研究方向为机器学习，数据挖掘，统计等.

程玉胜：男，1969年生，教授，研究方向为数据挖掘，机器学习等.