计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2017.20170002
54(6):1171-1184,2017

面向标记分布学习的标记增强

耿 新 徐 宁 邵瑞枫
(东南大学计算机科学与工程学院 南京 211189)
(计算机网络和信息集成教育部重点实验室(东南大学) 南京 211189)
(软件新技术与产业化协同创新中心(南京大学) 南京 210093)
(无线通信技术协同创新中心(东南大学) 南京 211189)
(xgeng@seu.edu.cn)

Label Enhancement for Label Distribution Learning

Geng Xin, Xu Ning, and Shao Ruifeng
(*School of Computer Science and Engineering, Southeast University, Nanjing* 211189)
(*Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, Nanjing* 211189)
(*Collaborative Innovation Center of Novel Software Technology and Industrialization (Nanjing University), Nanjing* 210093)
(*Collaborative Innovation Center of Wireless Communications Technology (Southeast University), Nanjing* 211189)

Abstract Multi-label learning (MLL) deals with the case where each instance is associated with multiple labels. Its target is to learn the mapping from instance to relevant label set. Most existing MLL methods adopt the uniform label distribution assumption, i.e., the importance of all relevant (positive) labels is the same for the instance. However, for many real-world learning problems, the importance of different relevant labels is often different. For this issue, label distribution learning (LDL) has achieved good results by modeling the different importance of labels with a label distribution. Unfortunately, many datasets only contain simple logical labels rather than label distributions. To solve the problem, one way is to transform the logical labels into label distributions by mining the hidden label importance from the training examples, and then promote prediction precision via label distribution learning. Such process of transforming logical labels into label distributions is defined as label enhancement for label distribution learning. This paper first proposes the concept of label enhancement with a formal definition. Then, existing algorithms that can be used for label enhancement have been surveyed, and compared in the experiments. Results of the experiments reveal that label enhancement can effectively discover the difference of the label importance hidden in the data, and improve the performance of multi-label learning.

Key words multi-label learning (MLL); label distribution learning (LDL); label enhancement; logical label; label distribution

摘 要 多标记学习(multi-label learning, MLL)任务处理一个示例对应多个标记的情况,其目标是学习一个从示例到相关标记集合的映射.在MLL中,现有方法一般都是采用均匀标记分布假设,也就是

收稿日期:2017-01-03;**修回日期**:2017-03-09
基金项目:国家自然科学基金优秀青年科学基金项目(61622203);江苏省自然科学基金杰出青年基金项目(BK20140022)
This work was supported by the National Natural Science Fundation of China for Excellent Young Scientists (61622203) and Jiangsu Natural Science Funds for Distinguished Young Scholar (BK20140022).

各个相关标记(正标记)对于示例的重要程度都被当作是相等的.然而,对于许多真实世界中的学习问题,不同相关标记的重要程度往往是不同的.为此,标记分布学习将不同标记的重要程度用标记分布来刻画,已经取得很好的效果.但是很多数据中却仅包含简单的逻辑标记而非标记分布.为解决这一问题,可以通过挖掘训练样本中蕴含的标记重要性差异信息,将逻辑标记转化为标记分布,进而通过标记分布学习有效地提升预测精度.上述将原始逻辑标记提升为标记分布的过程,定义为面向标记分布学习的标记增强.首次提出了标记增强这一概念,给出了标记增强的形式化定义,总结了现有的可以用于标记增强的算法,并进行了对比实验.实验结果表明:使用标记增强能够挖掘出数据中隐含的标记重要性差异信息,并有效地提升 MLL 的效果.

关键词 多标记学习;标记分布学习;标记增强;逻辑标记;标记分布

中图法分类号 TP391

多标记学习(multi-label learning, MLL)可以处理一个示例对应多个标记的情况,其目标是学习一个多标记的分类器,将示例映射到与之相关的标记集合上[1-3].在过去的十余年间,MLL 技术已经被广泛地应用于许多领域,例如文本[4]、图像[5]、语音[6]、视频[7]等的分类、识别和检索等,这些领域中的数据往往都含有丰富的语义,适合于用 MLL 来进行建模.

在 MLL 中,现有方法一般都是采用均匀标记分布假设,即与示例相关的标记都被当作是重要程度相等的[3].目前普遍的做法是,对于一个示例 $\boldsymbol{x}$,将 $l_{\boldsymbol{x}}^{y}\in\{0,1\}$ 赋予每个可能的标记 y,用以表示该标记 y 是否描述了示例 $\boldsymbol{x}$.因为 $l_{\boldsymbol{x}}^{y}$ 表达了是与否的逻辑关系,所以本文称这种标记为逻辑标记.在这样的训练样本中,标记的重要程度没有显式的表达.然而对于真实世界中的许多问题,不同相关标记的重要程度常常是不同的.例如:一幅自然场景的图像被标注了“天空”、“水”、“森林”和“云”等多个标记,而这些标记具体描述该图像的程度却是不同的;在人脸情感分析中,人的面部表情常常是多种基础情感(即快乐、悲伤、兴奋、恐惧、愤怒和厌恶)混合的结果,而这些基础情感在一个具体的表情中表达出不同的强度,从而呈现出纷繁复杂的情感.类似的例子还有很多,因为一旦一个示例与多个标记同时相关,那么这些标记一般情况下对它来说不会恰好都一样重要,而更可能会有主次前后之分.为了解决上述问题,对于一个示例 $\boldsymbol{x}$,可以将一个实数 $d_{\boldsymbol{x}}^{y}$ 赋予每个可能的标记 y,表示 y 描述 $\boldsymbol{x}$ 的程度.例如,$\boldsymbol{x}$ 表示一幅人脸表情图像,y 表示一种基础情感,那么 $d_{\boldsymbol{x}}^{y}$ 表示表情 $\boldsymbol{x}$ 中表达出 y 这种情感的强烈程度.不失一般性,假设 $d_{\boldsymbol{x}}^{y}\in[0,1]$,并进一步假设标记集合为完备集,即集合中的所有标记一定可以完整地描述一个示例,所以 $\sum_{y} d_{\boldsymbol{x}}^{y}=1$.满足以上 2 个条件的 $d_{\boldsymbol{x}}^{y}$ 称为 y 对 $\boldsymbol{x}$ 的描述度.对一个示例来讲,所有标记的描述度构成一种类似概率分布的数据结构,所以被称为标记分布,而在以标记分布标注的数据集上学习的过程就称为标记分布学习(label distribution learning, LDL)[8].

标记分布学习的出现使得从数据中学习比多标记更为丰富的语义成为可能,比如可以更精确地刻画与同一示例相关的多个标记的相对重要性差异等.事实上,Geng[8] 曾经指出,单标记学习和 MLL 都可以看作标记分布学习的特例,这也就意味着标记分布学习是一个更为泛化的机器学习框架,在此框架内研究机器学习方法具有重要的理论和应用价值.然而,标记分布学习应用的基础是假设每个示例由一个涵盖所有标记重要程度的标记分布来标注,这一点在很多实际应用中往往无法满足.这些实际应用中的数据多数情况下由单标记或者多标记(均匀标记分布)标注,缺乏完整的标记分布信息.尽管如此,这些数据中的监督信息本质上却是遵循某种标记分布的.这种标记分布虽然没有显式给出,却常常隐式地蕴含于训练样本中.如果能够通过合适的方法将其恢复出来,则可以真正发挥标记分布学习挖掘更多语义信息的优势.

基于上述考虑,本文提出的面向标记分布学习的标记增强是指将训练样本中的原始逻辑标记转化为标记分布的过程,这一过程依赖于对隐藏在训练样本中的标记相关信息的挖掘.假设 $\mathcal{Y}$ 表示样本的原始逻辑标记空间,$\mathcal{D}$ 表示经过标记增强后的标记分布空间,那么,标记增强方法将原始的标记空间 $\mathcal{Y}=\{0,1\}^{c}$ 拓展为 $\mathcal{D}=[0,1]^{c}$,其中 c 表示标记的个数.事实上,$\mathcal{D}$ 构成 c 维欧氏空间中的一个超立方体,而 $\mathcal{Y}$ 仅位于该超立方体的顶点.标记增强利用

隐含于数据中的标记间相关性,可以有效加强示例的监督信息,进而通过标记分布学习获得更好的预测效果.

尽管现有文献中并未明确提出过标记增强的概念,但是许多工作中实际上已经涉及了一些与之相关的方法.例如:在头部姿态估计问题中,文献[9-10]依靠对数据的先验知识,直接假设每个示例的标记分布为高斯分布;文献[11-12]用图模型表示示例间的拓扑结构,通过加入一些模型假设,建立示例间相关性与标记间相关性之间的关系,进而将示例的逻辑标记增强为标记分布;文献[13-16]从训练样本中生成示例对每个标记的模糊隶属度,从而可以将原有的逻辑标记增强为标记分布.上述工作有些是直接为标记分布学习而提出的,有些则是在其他领域提出但可以用来生成标记分布.不管哪种情况,它们都可以统一到同一个概念下,即本文提出的面向标记分布学习的标记增强.

1　符号及形式化定义

首先,本文主要的符号表示如下:示例用 $\boldsymbol{x}$ 表示,第 i 个示例用 $\boldsymbol{x}_i$ 表示;标记用 y 表示,第 j 个标记用 y_j 表示.$\boldsymbol{x}_i$ 的逻辑标记用 $\boldsymbol{L}_i=(l_{\boldsymbol{x}_i}^{y_1},l_{\boldsymbol{x}_i}^{y_2},\cdots,l_{\boldsymbol{x}_i}^{y_c})$ 表示,并且 $\boldsymbol{L}_i\in\{0,1\}^c$,$c$ 是可能的标记数目.y 对于 $\boldsymbol{x}$ 的描述度用 $d_{\boldsymbol{x}}^{y}$ 表示,其满足 $d_{\boldsymbol{x}}^{y}\in[0,1]$ 并且 $\sum\limits_{y}d_{\boldsymbol{x}}^{y}=1$.$\boldsymbol{x}_i$ 的标记分布用 $\boldsymbol{D}_i=(d_{\boldsymbol{x}_i}^{y_1},d_{\boldsymbol{x}_i}^{y_2},\cdots,d_{\boldsymbol{x}_i}^{y_c})$ 表示,并且 $\boldsymbol{D}_i\in[0,1]^c$.假设 $\mathcal{X}=\mathbb{R}^q$ 表示示例的特征空间,$\mathcal{Y}=\{y_1,y_2,\cdots,y_c\}$ 表示标记空间,则面向标记分布学习的标记增强定义如下:

给定训练集 $S=\{(\boldsymbol{x}_i,\boldsymbol{L}_i)\mid 1\leqslant i\leqslant n\}$,标记增强即根据 S 中蕴含的标记间相关性,将每个示例 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 转化为相应的标记分布 $\boldsymbol{D}_i$,从而得到标记分布训练集 $\mathcal{E}=\{(\boldsymbol{x}_i,\boldsymbol{D}_i)\mid 1\leqslant i\leqslant n\}$ 的过程.

2　标记增强方法

在标记分布学习这一概念提出之后,陆续有文献提出了一些面向标记分布学习的标记增强方法.这些研究有的利用了具体应用中的先验知识,如头部姿态和人脸年龄的先验分布[9-10];有的使用了半监督学习[17]中常用的标记传播方法[11];也有的引入了流形学习[12],均取得了不错的结果.而事实上,在标记分布学习概念提出之前,其他领域也已经出现了一些方法,尽管它们的应用背景和具体目标不尽相同,但是放到标记分布学习的框架之中,却可以用于实现标记增强,经典的如模糊聚类[13]、核隶属度[14]等.

本文将现有的标记增强算法分为 3 种类型,分别是基于先验知识的标记增强、基于模糊方法的标记增强和基于图模型的标记增强.下面分别阐述这 3 种类型中典型的标记增强算法.

2.1　基于先验知识的标记增强

基于先验知识的标记增强算法建立在对数据本身特点有较为深入理解的基础之上,完全依靠先验知识直接将逻辑标记增强为标记分布,或者部分引入先验知识,在此基础上通过挖掘隐含的标记间相关性将逻辑标记增强为标记分布.本节介绍 2 种基于先验知识的标记增强算法,分别是基于先验分布的标记增强算法和基于自适应先验分布的标记增强算法.

2.1.1　基于先验分布的标记增强

在某些特定的应用中,人们根据对数据的了解,可以预先知道每个示例应满足的标记分布的参数模型,这种含参标记分布模型就称为先验分布.一旦利用逻辑标记以及一些启发式方法确定了这种模型中的参数,就可以为每个示例生成相应的标记分布.

例如,在头部姿态估计应用[9]中,示例 $\boldsymbol{x}$ 表示人脸图像,姿态 $\boldsymbol{y}'$ 是一个由俯仰角和偏航角构成的二维向量,训练集是单标记数据,即一个 $\boldsymbol{x}$ 对应一个姿态 $\boldsymbol{y}$,则 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 满足 $\sum\limits_{j=1}^{c}l_{\boldsymbol{x}_i}^{y_j'}=1$ 且 $\boldsymbol{L}_i\in\{0,1\}^c$,训练集 $S=\{(\boldsymbol{x}_i,\boldsymbol{L}_i)\mid 1\leqslant i\leqslant n\}$.文献[9]中提出的标记增强算法假设示例 $\boldsymbol{x}$ 的逻辑标记 $\boldsymbol{L}$ 只是给出了一个“粗糙的”真实姿态 $\hat{\boldsymbol{y}}$,即 $\boldsymbol{L}$ 中 $l_{\boldsymbol{x}_i}^{\hat{y}}=1$ 的那个标记.值得指出的是,尽管在该例子中原始训练集是一个单标记数据集,但是在“粗糙姿态”的假设下,文献[9]实际上将该数据看成一个多标记数据集,即可由给定单标记姿态周边的多个姿态一起共同描述同一个示例.那么,根据头部姿态连续渐变的先验知识,可以假设 $\hat{\boldsymbol{y}}$ 对应的描述度 $d_{\boldsymbol{x}}^{\hat{y}}$ 是 $\boldsymbol{x}$ 的标记分布 $\boldsymbol{D}$ 中最大的元素,而姿态 $\hat{\boldsymbol{y}}$ 的相邻姿态 $\tilde{\boldsymbol{y}}$ 的描述度 $d_{\boldsymbol{x}}^{\tilde{y}}$ 接近 $d_{\boldsymbol{x}}^{\hat{y}}$ 但小于 $d_{\boldsymbol{x}}^{\hat{y}}$,并且随着 $\tilde{\boldsymbol{y}}$ 与 $\hat{\boldsymbol{y}}$ 的距离增大,$d_{\boldsymbol{x}}^{\tilde{y}}$ 呈逐渐减小的趋势.该算法假设标记分布 $\boldsymbol{D}$ 服从一种满足上述条件的典型分布,即离散二维高斯分布:

$$d_{\boldsymbol{x}}^{y}=\frac{1}{2\pi\sqrt{|\boldsymbol{\Sigma}|}Z}\exp\left(-\frac{1}{2}(\boldsymbol{y}-\hat{\boldsymbol{y}})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\hat{\boldsymbol{y}})\right),\quad(1)$$

其中,Z 是规范化因子,保证 $\sum_y d_x^y=1$,$\boldsymbol{\Sigma}$ 是预先给定的协方差矩阵(可通过启发式方法确定).该分布完全是基于关于数据本身的先验知识,由一个表示粗糙真实姿态的单标记生成的离散二维高斯标记分布.

基于先验分布的标记增强算法一般在假设一个先验分布的前提下,利用逻辑标记以及一些启发式方法确定先验分布的参数,直接将示例 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 转化为相应的标记分布 $\boldsymbol{D}_i$,从而得到标记分布训练集 $\mathcal{E}=\{(\boldsymbol{x}_i,\boldsymbol{D}_i)\mid 1\leqslant i\leqslant n\}$.这类方法依赖于算法设计者对数据本身的深入理解.如果这种理解与事实相符则可能获得不错的效果,并且实现起来方便高效.然而,一旦对数据的理解有所偏差,则标记增强后的结果往往并不理想.

2.1.2 基于自适应先验分布的标记增强

如前所述,2.1.1 节中的标记增强算法完全依靠先验知识直接将逻辑标记增强为标记分布.这一做法过于依赖先验知识,在许多对数据的了解不够充分的情况下,其生成的标记分布不一定能够真实反映数据本身的特点.为了解决上述问题,Geng 等人[10]以人脸年龄估计为应用背景,在引入先验分布的前提下,通过自适应方法确定先验分布中的参数,从而将每个示例 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 转化为相应的标记分布 $\boldsymbol{D}_i$.

具体地,示例 $\boldsymbol{x}$ 表示人脸图像,标记 y 表示年龄,训练集是单标记数据,即一张人脸图像对应唯一的年龄,因此示例 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 满足 $\sum_{j=1}^{c} l_{x_i}^{y_j}=1$ 且 $\boldsymbol{L}_i\in\{0,1\}^c$,训练集 $S=\{(\boldsymbol{x}_i,\boldsymbol{L}_i)\mid 1\leqslant i\leqslant n\}$.由于年龄越接近的人脸越相似,可以设想对一幅人脸图像来说,其真实年龄的描述度最高,相邻年龄的描述度向两边逐渐降低.这样,与 2.1.1 节中所举头部姿态估计的例子类似,尽管这里原始训练集是一个单标记数据集,但是根据相邻年龄相似性假设,文献[10]实际上将该数据看成一个多标记数据集,即可由给定单标记年龄相邻的多个年龄一起共同描述同一个示例.据此先验知识,可以假设 $\boldsymbol{x}_i$ 的标记分布 $\boldsymbol{D}_i$ 满足离散高斯分布:

$$d_{x_i}^{y}=\frac{1}{\sigma\sqrt{2\pi}Z}\exp\left(-\frac{(y-\alpha)^2}{2\sigma_\alpha^2}\right),\tag{2}$$

其中,Z 为 $d_{x_i}^y$ 的归一化因子,保证 $\sum_y d_{x_i}^y=1$. α 是 $\boldsymbol{x}_i$ 的真实年龄标记,也就是 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 中满足 $l_{x_i}^{y_j}=1$ 的那个标记,即 $\alpha=y_j$.式(2)中唯一需要进一步确定的参数 σ_α 为高斯分布的标准差.

文献[10]中计算参数 σ_α 的自适应方法为:①设定标准差 σ_α 的初值 σ^0;②通过式(2)计算标记分布;③使用标记分布学习算法[8]预测训练样本的标记分布,计算真实标记和预测标记(即分布中描述度最高的标记)的平均绝对误差,然后筛选误差较小的样本集用以更新标准差 σ_α;④通过求解一个非线性规划问题,计算最优的 σ'_α 使得对步骤③中选出的样本集来说,由 σ'_α 生成的分布和步骤③中预测分布的平均绝对误差最小.算法重复②~④步骤,直到训练样本的真实标记和预测标记的平均绝对误差小于预设阈值,最终确定式(2)中的参数 σ_α.通过上述自适应方法确定了先验分布的参数后,即可直接通过式(2)计算得到示例 $\boldsymbol{x}_i$ 的标记分布 $\boldsymbol{D}_i$,从而得到标记分布训练集 $\mathcal{E}$.

基于自适应先验分布的标记增强算法部分引入先验知识,通过自适应的方法,从训练样本中学习得到先验分布的参数,进而将每个示例 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 转化为相应的标记分布 $\boldsymbol{D}_i$.这类方法部分依赖先验知识,部分依赖从样例中学习,因此相较 2.1.1 节中完全依赖先验知识的方法更能有效利用隐藏在训练数据中的标记间相关性.正如文献[10]中的实验所报告的结果,一般情况下,基于自适应先验分布的标记增强算法效果要优于完全依赖先验分布的标记增强算法.

2.2 基于模糊方法的标记增强

基于模糊方法的标记增强[13-16]利用模糊数学的思想,通过模糊聚类、模糊运算和核隶属度等方法,挖掘出标记间相关信息,将逻辑标记转化为标记分布.值得注意的是,这类方法提出的目的一般是为了将模糊性引入原本刚性的逻辑标记,而并未明确其可以将逻辑标记增强为标记分布.但是,很多模糊标记增强方法实际上可以基于模糊隶属度轻松生成标记分布.

本节介绍 2 种基于模糊方法的标记增强算法,分别是基于模糊聚类的标记增强算法和基于核隶属度的标记增强算法.

2.2.1 基于模糊聚类的标记增强

基于模糊聚类的标记增强[13]通过模糊 C-均值聚类(fuzzy c-means algorithm, FCM)[18]和模糊运算,将训练集 S 中每个示例 $\boldsymbol{x}_i$ 的逻辑标记 $\boldsymbol{L}_i$ 转化为相应的标记分布 $\boldsymbol{D}_i$,从而得到标记分布训练集 $\mathcal{E}=\{(\boldsymbol{x}_i,\boldsymbol{D}_i)\mid 1\leqslant i\leqslant n\}$.

模糊 C-均值聚类(FCM)是用隶属度确定每个数据点属于某个聚类程度的一种聚类算法. FCM 把 n 个样本分为 p 个模糊聚类,并求每个聚类的中心,使得所有训练样本到聚类中心的加权(权值由样本点对相应聚类的隶属度决定)距离之和最小. 假设 FCM 将训练集 S 分成 p 个聚类,$\boldsymbol{\mu}_k$ 表示第 k 个聚类的中心,则可用:

$$\boldsymbol{m}_{\boldsymbol{x}_i}^k = \frac{1}{\sum_{j=1}^{p}\left(\frac{Dist(\boldsymbol{x}_i,\mu_k)}{Dist(\boldsymbol{x}_i,\mu_j)}\right)^{\frac{1}{\beta-1}}} \tag{3}$$

计算示例 $\boldsymbol{x}_i$ 对于每个聚类的隶属度 $\boldsymbol{m}_{\boldsymbol{x}_i}=(m_{\boldsymbol{x}_i}^1, m_{\boldsymbol{x}_i}^2,\cdots,m_{\boldsymbol{x}_i}^p)$. 其中,$Dist(,)$是任意的距离度量;$\beta$ 是模糊因子,且满足 $\beta>1$. 得到 $\boldsymbol{m}_{\boldsymbol{x}_i}$ 后,进一步构建一个关联矩阵 $\boldsymbol{A}$. 首先初始化一个 $c\times p$ 的零矩阵 $\boldsymbol{A}$,然后更新 $\boldsymbol{A}$ 的第 j 行 $\boldsymbol{A}_j$:

$$\boldsymbol{A}_j=\boldsymbol{A}_j+\boldsymbol{m}_{\boldsymbol{x}_i},\ \text{if } l_{\boldsymbol{x}_i}^{y_j}=1, \tag{4}$$

即 $\boldsymbol{A}_j$ 为所有属于第 j 个类的样本的隶属度向量之和. 经过行归一化后得到的矩阵 $\boldsymbol{A}$ 可以被当作一个"模糊关系"矩阵,$\boldsymbol{A}$ 中的元素 A_{jk} 表示了第 j 个类别(标记)与第 k 个聚类的关联强度.

根据模糊逻辑推理机制[18],将关联矩阵 $\boldsymbol{A}$ 与 $\boldsymbol{x}_i$ 对聚类的隶属度 $\boldsymbol{m}_{\boldsymbol{x}_i}$ 进行模糊合成运算,从而将 $\boldsymbol{x}_i$ 对聚类的隶属度转化为对类别的隶属度:

$$\boldsymbol{V}_i=\boldsymbol{A}\circ\boldsymbol{m}_{\boldsymbol{x}_i}, \tag{5}$$

其中,$\circ$ 表示模糊数学中的合成算子. 最后,对隶属度向量 $\boldsymbol{V}_i$ 进行归一化,使向量中元素的和为 1,即得到标记分布 $\boldsymbol{D}_i$.

基于模糊聚类的标记增强算法利用模糊聚类过程中产生的示例对每个聚类的隶属度,通过类别和聚类的关联矩阵,将示例对聚类的隶属度转化为对类别的隶属度,从而生成标记分布. 在这一过程中,模糊聚类反映了示例空间的拓扑关系,而通过关联矩阵将这种关系转化到标记空间,从而有可能使得简单的逻辑标记产生更丰富的语义,转变为标记分布.

2.2.2　基于核隶属度的标记增强

该标记增强方法源于一种模糊支持向量机中核隶属度的生成过程[14],通过一个非线性映射函数将示例 $\boldsymbol{x}_i$ 映射到高维空间,利用核函数计算该高维空间中正负类的中心、半径和各示例 $\boldsymbol{x}_i$ 到类别中心的距离,进而通过隶属度函数计算示例 $\boldsymbol{x}_i$ 的标记分布.

具体地,对于训练集 S 和某个标记 y_j,根据 $l_{\boldsymbol{x}_i}^{y_j}$ 的逻辑值,将 S 分为正类集合 $C_+^{y_j}$ 和负类集合 $C_-^{y_j}$,用 n_+ 和 n_- 分别表示正类与负类集合中示例的数量. $\Psi_+^{y_j}$ 和 $\Psi_-^{y_j}$ 表示正类和负类的中心:

$$\Psi_+^{y_j}=\frac{1}{n_+}\sum_{\boldsymbol{x}_i\in C_+^{y_j}}\phi(\boldsymbol{x}_i), \tag{6}$$

$$\Psi_-^{y_j}=\frac{1}{n_-}\sum_{\boldsymbol{x}_i\in C_-^{y_j}}\phi(\boldsymbol{x}_i), \tag{7}$$

其中,$\phi(\boldsymbol{x}_i)$是一个非线性映射函数,由核函数 $K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\phi(\boldsymbol{x}_i)\times\phi(\boldsymbol{x}_j)$确定. 正类和负类的半径分别计算为

$$r_+=\max\left\|\Psi_+^{y_j}-\phi(\boldsymbol{x}_i)\right\|, \tag{8}$$

$$r_-=\max\left\|\Psi_-^{y_j}-\phi(\boldsymbol{x}_i)\right\|. \tag{9}$$

样本 $\boldsymbol{x}_i$ 到正类和负类中心的平方距离分别是:

$$d_{i+}^2=\left\|\phi(\boldsymbol{x}_i)-\Psi_+^{y_j}\right\|^2, \tag{10}$$

$$d_{i-}^2=\left\|\phi(\boldsymbol{x}_i)-\Psi_-^{y_j}\right\|^2, \tag{11}$$

则示例 $\boldsymbol{x}_i$ 对于标记 y_j 的隶属度为

$$m_{\boldsymbol{x}_i}^{y_j}=\begin{cases}1-\sqrt{\dfrac{\left\|d_{i+}^2\right\|}{r_+^2+\delta}}, & l_{\boldsymbol{x}_i}^{y_j}=1,\\ 1-\sqrt{\dfrac{\left\|d_{i-}^2\right\|}{r_-^2+\delta}}, & l_{\boldsymbol{x}_i}^{y_j}=0,\end{cases} \tag{12}$$

其中,$\delta>0$,涉及 $\phi(\boldsymbol{x}_i)$的计算均可以由核函数$K(\boldsymbol{x}_i,\boldsymbol{x}_j)$间接计算. 最后,将 $\boldsymbol{m}_{\boldsymbol{x}_i}=(m_{\boldsymbol{x}_i}^{y_1},m_{\boldsymbol{x}_i}^{y_2},\cdots,m_{\boldsymbol{x}_i}^{y_c})$归一化,即可得到 $\boldsymbol{x}_i$ 的标记分布 $\boldsymbol{D}_i$.

基于核隶属度的标记增强算法利用核技巧在高维空间中计算示例对每个类别的隶属度,从而能够挖掘训练数据中类别标记间较为复杂的非线性系.

2.3　基于图模型的标记增强

基于图模型的标记增强算法用图模型表示示例间的拓扑结构,通过引入一些模型假设,建立示例间相关性与标记间相关性之间的关系,进而将示例的逻辑标记增强为标记分布. 本节介绍 2 种基于图模型的标记增强算法,分别是基于标记传播的标记增强算法和基于流形的标记增强算法.

2.3.1　基于标记传播的标记增强

文献[11]将半监督学习[17]中的标记传播技术应用于标记增强中. 该方法首先根据示例间相似度构建一个图,然后根据图中的拓扑关系在示例间传播标记,由于标记的传播会受到路径上权值的影响,会自然形成不同标记的描述度差异,当标记传播收敛时,每个示例的原有逻辑标记即可增强为标记分布.

具体地,假设多标记训练集 $S=\{(\boldsymbol{x}_i,\boldsymbol{L}_i)\mid 1\leqslant i\leqslant n\}$,$G=(V,E,\boldsymbol{W})$表示以 S 中的示例为顶点的全连通图,其中 $V=\{\boldsymbol{x}_i\mid 1\leqslant i\leqslant n\}$表示顶点,$E$ 表示顶点两两之间的边,$\boldsymbol{x}_i$ 与 $\boldsymbol{x}_j$ 之间的边上的权值为它们之间的相似度:

$$\forall_{i,j=1}^{n}: w_{ij}=\begin{cases}\exp\left(-\dfrac{\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2^2}{2}\right), & i\neq j;\\ 0, & i=j.\end{cases} \tag{13}$$

所有边的权值构成相似度矩阵 $\boldsymbol{W}=(w_{ij})_{n\times n}$. 标记传播矩阵 $\boldsymbol{P}$ 由相似度矩阵 $\boldsymbol{W}$ 计算而来: $\boldsymbol{P}=\boldsymbol{Q}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{Q}^{-\frac{1}{2}}$, 这里 $\boldsymbol{Q}=\mathrm{diag}(d_1,d_2,\cdots,d_m)$, 其中 $d_i=\sum_{j=1}^{n} w_{ij}$. 假设所有标记对所有示例的描述度构成一个描述度矩阵 $\boldsymbol{F}$, 该算法使用迭代方法不断更新 $\boldsymbol{F}$. $\boldsymbol{F}$ 的初始值 $\boldsymbol{F}^{(0)}=\boldsymbol{\Phi}=(\phi_{ij})_{n\times c}$ 由示例 $\boldsymbol{x}$ 的逻辑标记构成, 即 $\forall_{i=1}^{n}\ \forall_{j=1}^{c}: \phi_{ij}=l_{x_i}^{y_j}$. 在此基础上, 使用标记传播对描述度矩阵 $\boldsymbol{F}$ 进行更新:

$$\boldsymbol{F}^{(t)}=\alpha\boldsymbol{P}\boldsymbol{F}^{(t-1)}+(1-\alpha)\boldsymbol{\Phi}, \tag{14}$$

其中, α 是平衡参数, 控制了初始的逻辑标记和标记传播对最终描述度的影响程度. 经过迭代, 最终 $\boldsymbol{F}$ 收敛到: $\boldsymbol{F}^{*}=(1-\alpha)(\boldsymbol{I}-\alpha\boldsymbol{P})^{-1}\boldsymbol{\Phi}$, 对 $\boldsymbol{F}^{*}$ 做归一化处理:

$$\forall_{i=1}^{n}\ \forall_{j=1}^{c}: d_{x_i}^{y_j}=f_{ij}^{*}\Big/\sum_{k=1}^{c} f_{ik}^{*}, \tag{15}$$

即得到示例 $\boldsymbol{x}_i$ 的标记分布 $\boldsymbol{D}_i=(d_{x_i}^{y_1},d_{x_i}^{y_2},\cdots,d_{x_i}^{y_c})$.

基于标记传播的标记增强算法通过图模型表示示例间的拓扑结构, 构造了基于示例间相关性的标记传播矩阵, 利用传播过程中路径权值的不同使得不同标记的描述度自然产生差异, 从而反映出蕴含在训练数据中的标记间关系.

2.3.2 基于流形的标记增强

基于流形的标记增强算法[12]假设数据在特征空间和标记空间均分布在某种流形上, 并利用平滑假设将 2 个空间的流形联系起来, 从而可以利用特征空间流形的拓扑关系指导标记空间流形的构建, 在此基础上将示例的逻辑标记增强为标记分布.

具体地, 该算法用图 $G=(V,E,\boldsymbol{W})$ 表示多标记训练集 S 的特征空间的拓扑结构, 其中 V 是由示例构成的顶点集合, E 是边的集合, $\boldsymbol{W}=(w_{ij})_{n\times n}$ 是图的边权重矩阵. 首先, 在特征空间中, 假设示例分布的流形满足局部线性, 即任意示例 $\boldsymbol{x}_i$ 可以由它的 K-近邻的线性组合重构, 重构权值矩阵 $\boldsymbol{W}$ 可通过最小化得到:

$$\Omega(\boldsymbol{W})=\sum_{i=1}^{n}\Big\|\boldsymbol{x}_i-\sum_{j\neq i} w_{ij}\boldsymbol{x}_j\Big\|^2,$$

$$\text{s.t. } w_{ij}=0,\ \boldsymbol{x}_j \text{ 不是 } \boldsymbol{x}_i \text{ 的 } K\text{-近邻},$$

$$\boldsymbol{1}^{\mathrm{T}}\boldsymbol{W}_i^{\mathrm{T}}=1, \tag{16}$$

其中, $\boldsymbol{W}_i$ 表示 $\boldsymbol{W}$ 的第 i 行, $\boldsymbol{1}^{\mathrm{T}}$ 表示全部由 1 构成的向量. 通过平滑假设[19], 即特征相似的示例的标记也很可能相似, 可将特征空间的拓扑结构迁移到标记空间中, 即共享同样的局部线性重构权值矩阵 $\boldsymbol{W}$. 这样, 标记空间的标记分布可由最小化得到:

$$\Phi(\hat{\boldsymbol{D}})=\sum_{i=1}^{n}\Big\|\hat{\boldsymbol{D}}_i-\sum_{j\neq i} w_{ij}\hat{\boldsymbol{D}}_j\Big\|^2, \tag{17}$$

其中, $\hat{\boldsymbol{D}}_i=(\hat{d}_{x_i}^{y_1},\hat{d}_{x_i}^{y_2},\cdots,\hat{d}_{x_i}^{y_c})$ 表示 $\boldsymbol{x}_i$ 的标记分布, 为了引入逻辑标记 $\boldsymbol{L}_i=(l_{x_i}^{y_1},l_{x_i}^{y_2},\cdots,l_{x_i}^{y_c})$, 对 $\hat{d}_{x_i}^{y}$ 加入约束:

$$\forall 1\leqslant i\leqslant n, 1\leqslant l\leqslant c\quad l_{x_i}^{y_c}\hat{d}_{x_i}^{y_c}\geqslant\lambda(\lambda>0). \tag{18}$$

值得指出的是, 为了方便构建上述约束条件, 文献[12]中定义的逻辑标记 $\boldsymbol{L}_i\in\{-1,1\}^c$, 而不是其他方法中常用的 $\boldsymbol{L}_i\in\{0,1\}^c$, 但两者本质上并没有区别. 通过求解上述二次规划问题确定 $\hat{\boldsymbol{D}}_i$ 后, 经过归一化即可得到标记分布 $\boldsymbol{D}_i$, 进而得到标记分布训练集 $\mathcal{E}=\{(\boldsymbol{x}_i,\boldsymbol{D}_i)\mid 1\leqslant i\leqslant n\}$.

基于流形的方法通过重构特征空间和标记空间的流形, 利用平滑假设将特征空间的拓扑关系迁移到标记空间中, 建立示例间相关性与标记间相关性之间的关系, 从而将逻辑标记增强为标记分布.

2.4 小结与比较

本节简要介绍了现存的 3 类可用于实现标记增强的方法, 分别是基于先验知识的标记增强、基于模糊方法的标记增强和基于图模型的标记增强, 每一类方法分别介绍了几种典型的实现算法. 这些算法有些是专门为了将逻辑标记增强为标记分布而提出的, 如基于先验知识的标记增强方法和基于图模型的标记增强方法; 有些则是源于其他领域的工作, 但可以借用到本文语境中实现标记增强, 如基于模糊方法的标记增强. 这些不同的增强方法, 或者通过先验知识, 或者通过从样本中学习来获得额外的监督信息, 从而将训练样本原有的简单逻辑标记转化为信息量更为丰富的标记分布. 综合来看, 对于不同类型的标记增强算法, 其优点和缺点总结如下:

1) 基于先验知识的标记增强. 优点: 在对数据有比较深入理解的前提下, 可以充分利用先验知识, 快速高效地实现标记增强; 缺点: 过于依赖先验知识, 在缺乏关于数据的领域知识的情况下无法应用此类方法.

2) 基于模糊方法的标记增强. 优点: 利用模糊隶属度, 将标记间相关信息与逻辑标记融合, 不需要建立精确的数学模型, 有较强的鲁棒性, 数据和参数等对算法影响较小; 缺点: 缺乏深度挖掘标记空间和特征空间信息的明确模型, 往往难以生成符合特定数据特点的标记分布.

3) 基于图模型的标记增强. 优点:充分利用特征空间的拓扑关系来指导标记空间相关性信息挖掘,有良好的数学基础,有利于形成适合数据本身特点的标记增强算法;缺点:模型较为复杂,数据和参数对算法的影响较大.

3 实　　验

本节实验在不同数据集上测试了第 2 节提到的 3 种代表性的标记增强算法. 由于基于先验知识的标记增强方法依赖于对数据本身的深入理解,只有在特定数据集上才能应用,因此这类算法这里不作比较.

实验主要分为 2 个部分:

1) 在一个人造数据集上测试所有对比算法由逻辑标记增强为标记分布的精度;

2) 在 11 个多标记数据集上,将基于标记增强的标记分布学习算法与主流的 MLL 算法进行对比.

3.1 实验设置

3.1.1 数据集

实验中共使用了 12 个数据集,包括 1 个人造数据集和 11 个真实世界的多标记数据集.

人造数据集用来直观展现各个标记增强算法得到的标记分布. 该数据集中,示例 $\boldsymbol{x}$ 是三维向量,标记分布也是三维(即共有 3 个标记). 示例 $\boldsymbol{x}=(x_1', x_2', x_3')$ 的标记分布 $\boldsymbol{D}=(d_{\boldsymbol{x}}^{y_1}, d_{\boldsymbol{x}}^{y_2}, d_{\boldsymbol{x}}^{y_3})$ 由以下方法产生:

$$t_i = a x_i' + b x_i'^2 + c_i'^3 + d,\ i=1,2,3, \tag{19}$$

$$\psi_1 = (\boldsymbol{\omega}_1^{\mathrm{T}} \boldsymbol{t})^2, \tag{20}$$

$$\psi_2 = (\boldsymbol{\omega}_2^{\mathrm{T}} \boldsymbol{t} + \lambda_1 \psi_1)^2, \tag{21}$$

$$\psi_3 = (\boldsymbol{\omega}_3^{\mathrm{T}} \boldsymbol{t} + \lambda_2 \psi_2)^2, \tag{22}$$

$$d_{\boldsymbol{x}}^{y_i} = \frac{\psi_i}{\psi_1+\psi_2+\psi_3}, i=1,2,3, \tag{23}$$

其中 $\boldsymbol{t}=(t_1,t_2,t_3)^{\mathrm{T}}$, $x_i' \in [-1,1]$,参数 $a=1, b=0.5, c=0.2, d=1, \boldsymbol{\omega}_1^{\mathrm{T}}=(4,2,1), \boldsymbol{\omega}_2^{\mathrm{T}}=(1,2,4), \boldsymbol{\omega}_3^{\mathrm{T}}=(1,4,2), \lambda_1=\lambda_2=0.01$.

人造数据集采样方法如下:首先,$\boldsymbol{x}$ 中的 x_1', x_2' 分量在 $[-1,1]^2$ 范围内按照间隔为 0.04 的网格采样,共采样 $51\times51=2\,601$ 次,而 x_3' 为

$$x_3' = \sin((x_1'+x_2')\times\pi). \tag{24}$$

这样共生成 2 601 个分布于三维特征空间流形上的示例,每个示例对应的真实标记分布 $\boldsymbol{D}$ 由式(19)～(23)产生.

为了赋予每个示例逻辑标记,需要将其对应的标记分布二值化. 二值化的过程如下:首先选取标记分布 $\boldsymbol{D}$ 中最大的描述度 $d_{\boldsymbol{x}}^{y_j}$,将对应的标记 y_j 置为相关标记,即令 $l_{\boldsymbol{x}}^{y_j}=1$. 计算相关标记的描述度之和 $\tau=\sum_{y_j\in\mathcal{Y}^+} d_{\boldsymbol{x}}^{y_j}$,其中 $\mathcal{Y}^+$ 表示当前的相关标记集合,如果 τ 小于某个预设的阈值 T,则继续选择尚未选入相关标记集合的其他标记中描述度最大的一个加入 $\mathcal{Y}^+$,该过程持续直至 $\tau\geqslant T$. 所有选入 $\mathcal{Y}^+$ 的标记在 $\boldsymbol{x}$ 的逻辑标记中置为 1,而所有其他标记置为 0. 本实验中设置 $T=0.6$.

得到每个示例的逻辑标记 $\boldsymbol{L}$ 后,使用不同的标记增强算法将 $\boldsymbol{L}$ 增强为标记分布 $\hat{\boldsymbol{D}}$. 随后即可通过比较 $\hat{\boldsymbol{D}}$ 与真实标记分布 $\boldsymbol{D}$ 来评价标记增强算法的表现. 具体的评价指标将在 3.1.3 节中详述.

Table 1　Attributes of the Benchmark Multi-label Data Sets

表 1　基准多标记数据集属性

No.	Dataset	$\lvert S\rvert$	$dim(S)$	$L(S)$	$F(S)$	$LCard(S)$	$LDen(S)$	$DL(S)$	$PDL(S)$	Domain
1	cal500	502	68	174	numeric	26.044	0.150	502	1.000	audio
2	emotion	593	72	6	numeric	1868	0.311	27	0.046	audio
3	medical	978	1449	45	nominal	1245	0.028	94	0.096	text
4	llog	1460	1004	75	nominal	1.180	0.016	304	0.208	text
5	enron	1702	1001	53	nominal	3.378	0.064	753	0.442	text
6	msrc	1868	898	19	numeric	6315	0.332	947	0.507	image
7	image	2000	294	5	numeric	1.236	0.247	20	0.010	image
8	scene	2.407	294	5	numeric	1.074	0.179	15	0.006	image
9	yeast	2.417	103	14	numeric	4.237	0.303	198	0.082	biology
10	slashdot	3782	1079	22	nominal	1.181	0.054	156	0.041	text
11	corel5k	5000	499	374	nominal	3.522	0.009	3175	0.635	image

实验第 2 部分使用 11 个基准多标记数据集①，这些数据集均来自真实应用场景，在本实验中用于比较基于标记增强的 MLL 与传统 MLL 算法. 表 1 总结了这些数据集的各种属性. 其中，$|S|$，$dim(S)$，$L(S)$和 $F(S)$分别表示数据集的样本数目、特征维度、类别标记数目和特征类型. 另外还有一些关于多标记数据的统计指标[20]，包括标记基数 $LCard(S)$、标记密度 $LDen(S)$、独特标记集合 $DL(S)$和独特标记集合占比 $PDL(S)$.

3.1.2 对比算法

如 2.1.1 节所述，由于基于先验知识的标记增强方法依赖于对数据本身的深入理解，只有在特定数据集上才能应用，因此这类算法这里不作比较. 本实验中主要考查 4 种标记增强算法，分别是基于模糊聚类的标记增强算法(FCM)、基于核隶属度的标记增强算法(KM)、基于标记传播的标记增强算法(LP)和基于流形的标记增强算法(ML). 其中，FCM 中 $\beta=2$，KM 中的核函数为线性核函数($K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j$)，LP 算法中 $\alpha=0.5$，ML 算法中 K 近邻设置为 $c+1$.

实验的第 1 部分在人造数据集上直接对比 4 种标记增强算法生成的标记分布与真实标记分布相比的相似程度. 第 2 部分在 11 个多标记数据集上对比基于标记增强的 MLL 和传统的 MLL 算法. 所谓基于标记增强的 MLL 是指先分别用 4 种标记增强算法将数据集中原有的逻辑标记增强为标记分布；然后用标记分布学习算法 SA-BFGS[8] 为从示例到标记分布的映射建模；最后在测试时，用 3.1.1 节介绍的二值化方法将预测出的标记分布转化为逻辑标记.

用于对比的传统 MLL 算法包括 4 种 MLL 领域的主流算法，分别为 binary relevance(BR)[21]，calibrated label ranking (CLR)[22]，ensemble of classifier chains (ECC)[20]和 RAndom K-labELsets (RAKEL)[23]. 这 4 种算法都在 MULAN MLL 包[24]上运行，基分类器为 logistic regression 模型，ECC 全体尺度设置为 30，RAKEL 的全体尺度设置为 $2c$ 且 $K=3$.

3.1.3 评价指标

标记增强算法的输出是标记分布，评价算法表现的一个自然指标就是增强后的标记分布与真实分布之间的平均距离或相似度. 根据文献[8]中的建议，本实验中采用 6 种代表性的标记分布评价指标，分别为 Chebyshev 距离(Cheb)、Clark 距离(Clark)、Canberra 距离(Canber)、Kullback-Leibler 散度(KL-div)、余弦相关系数(Cosine)和交叉相似度(Intersec). 其中，前 4 个指标是距离指标，后 2 个是相似度指标. 假设真实标记分布和增强后的标记分布分别为 $\boldsymbol{D}=(d_1,d_2,\cdots,d_c)$ 和 $\hat{\boldsymbol{D}}=(\hat{d}_1,\hat{d}_2,\cdots,\hat{d}_c)$，则各指标的计算公式如表 2 所示. 其中，指标名称后的“↓”表示“越低越好”，“↑”表示“越高越好”.

Table 2 Evaluation Measures for Label Enhancement Algorithms

表 2 标记增强算法的评价指标

Measure	Formula
Cheb ↓	$Dist_1(D,\hat{D})=\max_j \lvert d_j-\hat{d}_j\rvert$
Clack ↓	$Dist_2(\boldsymbol{D},\hat{\boldsymbol{D}})=\sqrt{\sum_{j=1}^{c}\frac{(d_j-\hat{d}_j)^2}{(d_j+\hat{d}_j)^2}}$
Canber ↓	$Dist_3(\boldsymbol{D},\hat{\boldsymbol{D}})=\sum_{j=1}^{c}\frac{\lvert d_j-\hat{d}_j\rvert}{d_j+\hat{d}_j}$
KL-div ↓	$Dist_4(\boldsymbol{D},\hat{\boldsymbol{D}})=\sum_{j=1}^{c}d_j\ln\frac{d_j}{\hat{d}_j}$
Cosine ↑	$Sim_1(\boldsymbol{D},\hat{\boldsymbol{D}})=\frac{\sum_{j=1}^{c}d_j\hat{d}_j}{\sqrt{\sum_{j=1}^{c}d_j^2}\sqrt{\sum_{j=1}^{c}\hat{d}_j^2}}$
Intersec ↑	$Sim_2(\boldsymbol{D},\hat{\boldsymbol{D}})=\sum_{j=1}^{c}\min(d_j,\hat{d}_j)$

Notes: “↓” indicates the smaller the better, and “↑” indicates the larger the better.

将标记增强算法与传统的 MLL 方法比较时，我们使用了在 MLL 中常用的 5 种评价指标，分别是 Hamming-loss, One-error, Coverage, Ranking-loss 和 Average-precision[25].

3.2 实验结果

3.2.1 标记分布比较实验

为了直观可视化标记增强的效果，假设人造数据集中标记分布的 3 个分量分别对应 RGB 颜色空间的 3 个颜色通道. 这样，每个示例的标记分布就可以用示例空间中点的颜色来直观表示. 根据 3.1.1 节描述的人造数据采样方法可知，2 601 个样本点分布在三维示例空间的一个流形上. 因此，通过比较这个流形上的颜色模式即可直观判断不同算法标记增强的效果. 图 1 显示了人造数据集上的真实标记分布(图 1(a))以及 4 种标记增强算法得到的标记分布

① http://meka.sourceforge.net/#datasets 和 http://mulan.sourceforge.net/datasets.html

(图 1(b)～1(e)).为了方便观察,图 1 中对流形上的颜色应用了去相关拉伸(decorrelation stretch)技术来增强颜色对比度.由图 1 可以看出,LP 算法标记增强后的颜色模式非常接近真实标记分布,KM 算法和 ML 算法记增强后的颜色模式也与真实标记分布相似,FCM 算法的颜色模式和真实值差距较大.

进一步,我们对 4 种标记增强算法在人造数据集上的表现进行了定量分析.表 3 给出了 4 种标记增强算法在表 2 所示的 6 种评价指标上的比较结果,每个结果后的括号中给出了相应算法的排序,并且统计了每种算法在所有指标上的平均排序.表 3 中的结果与图 1 的可视化比较结果一致,即根据平均排序:LP＞KM＞ML＞FCM. LP 算法的表现最好,因为该算法在保留了原始的逻辑标记的前提下,使用示例间相关信息对逻辑标记进行了增强;KM 算法使用了模糊方法,不会对增强后的标记分布引入过多的误差,因此该算法也能够得到较为不错的结果;ML 算法使用了流形模型,将示例间的相关信息与标记相关信息建立了联系,但使用的约束项不能够保留较多的原始逻辑信息,因此标记增强效果

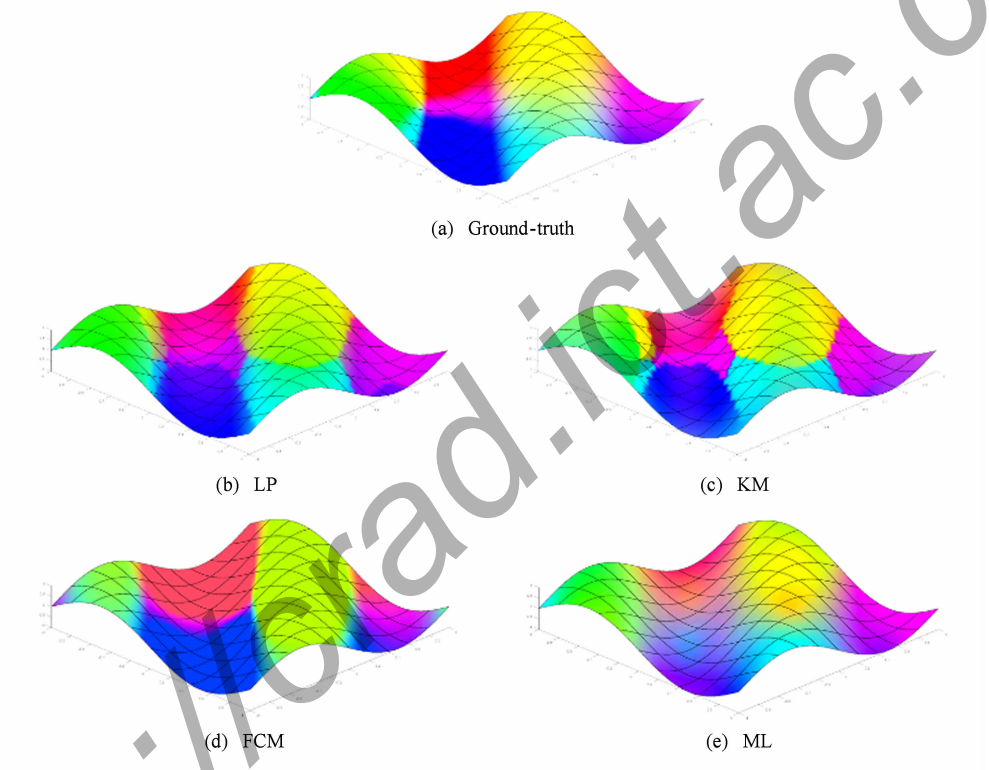

Fig. 1　Visual comparison between the label distributions generated by the label enhancement algorithms and the ground-truth label distributions

图 1　标记增强算法生成的标记分布与真实标记分布的可视化对比

Table 3　Quantitative Comparison Between the Label Distributions Generated by the Label Enhancement Algorithms and the Ground-truth Label Distributions

表 3　标记增强算法生成的标记分布与真实标记分布的定量对比

Criterion	LP(Rank)	KM(Rank)	FCM(Rank)	ML(Rank)
Cheb ↓	**0.125 663(1)**	0.162 92(2)	0.200 96(4)	0.177 705(3)
Clack ↓	**0.443 606(1)**	0.474 112(2)	0.529 347(4)	0.484 739(3)
Canber ↓	**0.651 589(1)**	0.707 676(2)	0.785 12(4)	0.729 24(3)
KL-div ↓	**0.076 333(1)**	0.126 129(2)	0.151 067(4)	0.134 479(3)
Cosine ↑	**0.975 115(1)**	0.951 5(2)	0.929 369(4)	0.941 934(3)
Intersec ↑	**0.874 337(1)**	0.837 08(2)	0.799 04(4)	0.822 295(3)
Average Rank	1	2	4	3

Notes:"↓" indicates the smaller the better, and "↑" indicates the larger the better.

与 KM 算法接近;FCM 算法的标记分布生成中使用了简单的模糊合成运算,不能有效地挖掘标记相关信息,因此标记增强的效果较差.

3.2.2 MLL 比较实验

实验的第 2 部分在 11 个真实世界的多标记数据集上比较基于标记增强的 MLL 与传统 MLL 算法.在每个数据集上采用 10 倍交叉验证,记录每个算法分别在 5 个 MLL 评价指标上的平均值,结果如表 4~8 所示.其中,评价指标后的"↓"表示"越低越好","↑"表示"越高越好".在每种指标和每个

Table 4 Comparison of Multi-label Learning Algorithms on Ranking-loss↓ (mean±std)

表 4 MLL 算法在 Ranking-loss↓ 指标(均值±标准差)上的比较

Dataset	LP(Rank)	ML(Rank)	KM(Rank)	FCM(Rank)	BR(Rank)	CLR(Rank)	ECC(Rank)	RAKEL(Rank)
cal500	**0.181±0.003(1)**	0.188±0.002(2)	0.197±0.004(3)	0.485±0.010(8)	0.258±0.003(6)	0.239±0.026(5)	0.205±0.004(4)	0.444±0.005(7)
emotions	**0.182±0.012(1)**	0.231±0.012(5)	0.202±0.010(2)	0.435±0.020(8)	0.233±0.016(6)	0.222±0.014(3)	0.227±0.017(4)	0.254±0.020(7)
medical	0.034±0.006(3)	**0.032±0.005(1)**	0.100±0.013(6)	0.409±0.026(8)	0.091±0.005(4)	0.123±0.026(7)	0.032±0.007(2)	0.095±0.033(5)
llog	**0.125±0.005(1)**	0.158±0.005(3)	0.358±0.012(6)	0.453±0.011(8)	0.328±0.007(5)	0.190±0.015(4)	0.154±0.009(2)	0.412±0.010(7)
enron	0.091±0.003(3)	0.090±0.012(2)	0.287±0.010(6)	0.497±0.013(8)	0.312±0.009(7)	**0.089±0.002(1)**	0.120±0.004(4)	0.241±0.005(5)
msra	**0.141±0.014(1)**	0.157±0.010(2)	0.167±0.008(3)	0.505±0.015(8)	0.368±0.021(7)	0.288±0.018(5)	0.332±0.047(6)	0.223±0.075(4)
image	0.181±0.008(2)	**0.143±0.007(1)**	0.325±0.101(7)	0.493±0.014(8)	0.314±0.014(6)	0.294±0.009(4)	0.276±0.005(3)	0.311±0.010(5)
scene	0.087±0.006(2)	**0.064±0.003(1)**	0.490±0.038(7)	0.494±0.017(8)	0.229±0.010(6)	0.127±0.003(3)	0.151±0.005(4)	0.205±0.008(5)
yeast	0.174±0.004(2)	**0.168±0.003(1)**	0.351±0.005(7)	0.503±0.004(8)	0.190±0.004(4)	0.198±0.003(5)	0.190±0.003(3)	0.245±0.004(6)
slashdot	0.132±0.005(3)	**0.095±0.003(1)**	0.251±0.009(6)	0.434±0.005(8)	0.240±0.008(5)	0.260±0.007(7)	0.123±0.004(2)	0.190±0.005(4)
corel5k	0.145±0.002(2)	0.163±0.003(3)	0.300±0.004(5)	0.472±0.004(7)	0.416±0.003(6)	**0.114±0.002(1)**	0.292±0.003(4)	0.627±0.004(8)
Average Rank	1.909	2	5.273	7.909	5.636	4.091	3.455	5.727

Table 5 Comparison of Multi-label Learning Algorithms on One-error↓ (mean±std)

表 5 MLL 算法在 One-error↓ 指标(均值±标准差)上的比较

Dataset	LP(Rank)	ML(Rank)	KM(Rank)	FCM(Rank)	BR(Rank)	CLR(Rank)	ECC(Rank)	RAKEL(Rank)
cal500	**0.120±0.015(1)**	0.141±0.016(2)	0.230±0.025(4)	0.957±0.024(8)	0.921±0.025(7)	0.331±0.111(6)	0.191±0.021(3)	0.286±0.039(5)
emotions	**0.303±0.027(1)**	0.352±0.021(3)	0.326±0.010(2)	0.635±0.020(8)	0.375±0.027(6)	0.356±0.030(5)	0.353±0.040(4)	0.392±0.035(7)
medical	0.213±0.021(4)	**0.179±0.019(1)**	0.380±0.049(6)	0.978±0.013(8)	0.297±0.036(5)	0.688±0.143(7)	0.182±0.019(2)	0.208±0.071(3)
llog	0.748±0.011(2)	**0.683±0.018(1)**	0.857±0.013(5)	0.979±0.004(8)	0.884±0.011(6)	0.900±0.019(7)	0.785±0.009(3)	0.838±0.014(4)
enron	0.311±0.013(2)	**0.258±0.090(1)**	0.490±0.015(6)	0.917±0.013(8)	0.648±0.019(7)	0.376±0.017(3)	0.424±0.013(5)	0.412±0.016(4)
msra	0.097±0.028(2)	**0.085±0.011(1)**	0.128±0.019(3)	0.593±0.045(8)	0.464±0.032(7)	0.312±0.085(5)	0.420±0.105(6)	0.302±0.103(4)
image	0.353±0.017(2)	**0.272±0.009(1)**	0.537±0.110(6)	0.763±0.020(8)	0.538±0.019(7)	0.514±0.014(4)	0.486±0.018(3)	0.515±0.017(5)
scene	0.270±0.016(2)	**0.194±0.008(1)**	0.802±0.027(7)	0.819±0.014(8)	0.475±0.014(6)	0.371±0.008(3)	0.373±0.008(4)	0.444±0.012(5)
yeast	0.241±0.011(2)	**0.228±0.009(1)**	0.382±0.012(7)	0.682±0.056(8)	0.285±0.008(6)	0.270±0.007(5)	0.256±0.007(4)	0.251±0.008(3)
slashdot	0.558±0.009(4)	**0.382±0.009(1)**	0.638±0.010(5)	0.931±0.009(7)	0.734±0.017(6)	0.979±0.003(8)	0.481±0.014(3)	0.453±0.005(2)
corel5k	0.755±0.005(5)	**0.647±0.007(1)**	0.736±0.004(4)	0.989±0.004(8)	0.919±0.006(7)	0.721±0.007(3)	0.699±0.006(2)	0.819±0.010(6)
Average Rank	2.455	1.273	5	7.909	6.364	5.091	3.545	4.364

Table 6 Comparison of Multi-label Learning Algorithms on Hamming-loss↓ (mean±std)

表 6 MLL 算法在 Hamming-loss↓ 指标(均值±标准差)上的比较

Dataset	LP(Rank)	ML(Rank)	KM(Rank)	FCM(Rank)	BR(Rank)	CLR(Rank)	ECC(Rank)	RAKEL(Rank)
cal500	0.167±0.004(6)	**0.138±0.002(1)**	0.141±0.002(3)	0.216±0.002(8)	0.214±0.004(7)	0.165±0.005(5)	0.146±0.002(4)	0.138±0.002(1)
emotions	**0.223±0.007(1)**	0.243±0.010(2)	0.253±0.008(3)	0.356±0.010(8)	0.265±0.013(5)	0.270±0.011(7)	0.254±0.013(4)	0.269±0.011(6)
medical	0.017±0.001(3)	0.283±0.027(8)	0.032±0.000(6)	0.111±0.001(7)	0.022±0.003(4)	0.024±0.002(5)	0.013±0.001(2)	**0.010±0.003(1)**
llog	**0.016±0.000(1)**	0.021±0.001(5)	0.070±0.004(7)	0.093±0.001(8)	0.052±0.003(6)	0.019±0.002(4)	0.016±0.000(1)	0.017±0.001(3)
enron	0.063±0.003(3)	**0.051±0.001(1)**	0.068±0.001(5)	0.154±0.003(8)	0.105±0.003(7)	0.072±0.002(6)	0.064±0.001(4)	0.058±0.001(2)
msra	0.279±0.017(4)	**0.209±0.009(1)**	0.219±0.005(2)	0.355±0.009(7)	0.404±0.037(8)	0.342±0.033(5)	0.353±0.037(6)	0.237±0.079(3)
image	0.190±0.005(2)	**0.156±0.004(1)**	0.265±0.051(4)	0.352±0.009(8)	0.287±0.008(6)	0.305±0.005(7)	0.244±0.005(3)	0.286±0.007(5)
scene	0.127±0.005(2)	**0.076±0.003(1)**	0.281±0.010(7)	0.285±0.005(8)	0.184±0.005(6)	0.181±0.004(5)	0.133±0.002(3)	0.171±0.005(4)
yeast	0.214±0.004(3)	**0.196±0.003(1)**	0.286±0.004(7)	0.355±0.010(8)	0.219±0.003(5)	0.222±0.002(6)	0.216±0.002(4)	0.202±0.003(2)
slashdot	0.060±0.002(5)	**0.043±0.001(1)**	0.090±0.002(6)	0.134±0.000(8)	0.130±0.003(7)	0.058±0.001(4)	0.049±0.001(3)	0.048±0.001(2)
corel5k	0.024±0.000(5)	**0.010±0.001(1)**	0.097±0.000(7)	0.103±0.001(8)	0.027±0.000(6)	0.011±0.001(2)	0.015±0.001(4)	0.012±0.001(3)
Average Rank	3.227	2.136	5.182	7.818	6.091	5.091	3.5	2.955

Table 7　Comparison of Multi-label Learning Algorithms on Coverage↓ (mean±std)

表 7　MLL 算法在 Coverage↓ 指标(均值±标准差)上的比较

Dataset	LP(Rank)	ML(Rank)	KM(Rank)	FCM(Rank)	BR(Rank)	CLR(Rank)	ECC(Rank)	RAKEL(Rank)
cal500	**0.747±0.007(1)**	0.780±0.008(3)	0.777±0.010(2)	0.888±0.006(7)	0.852±0.014(6)	0.794±0.010(5)	0.788±0.008(4)	0.971±0.001(8)
emotions	**0.318±0.031(1)**	0.357±0.009(5)	0.334±0.002(2)	0.482±0.003(8)	0.363±0.015(6)	0.351±0.016(3)	0.356±0.013(4)	0.381±0.019(7)
medical	0.052±0.001(3)	**0.048±0.008(1)**	0.125±0.015(6)	0.436±0.024(8)	0.118±0.007(5)	0.143±0.030(7)	0.048±0.009(2)	0.117±0.040(4)
llog	**0.159±0.006(1)**	0.162±0.008(2)	0.340±0.012(5)	0.409±0.011(7)	0.377±0.008(6)	0.225±0.016(4)	0.192±0.010(3)	0.459±0.011(8)
enron	**0.242±0.005(1)**	0.256±0.017(3)	0.577±0.015(6)	0.736±0.013(8)	0.601±0.014(7)	0.243±0.006(2)	0.300±0.009(4)	0.523±0.008(5)
msra	**0.543±0.020(1)**	0.574±0.016(2)	0.586±0.017(3)	0.868±0.007(8)	0.759±0.018(7)	0.720±0.023(5)	0.743±0.033(6)	0.628±0.210(4)
image	0.198±0.007(2)	**0.168±0.007(1)**	0.297±0.089(5)	0.445±0.012(8)	0.301±0.012(7)	0.286±0.008(4)	0.272±0.005(3)	0.298±0.010(6)
scene	0.171±0.009(4)	**0.067±0.003(1)**	0.419±0.029(7)	0.424±0.014(8)	0.207±0.009(6)	0.120±0.007(2)	0.141±0.004(3)	0.186±0.006(5)
yeast	**0.451±0.005(1)**	0.454±0.004(2)	0.633±0.004(7)	0.783±0.012(8)	0.474±0.005(3)	0.492±0.006(5)	0.476±0.004(4)	0.558±0.006(6)
slashdot	0.148±0.005(3)	**0.112±0.003(1)**	0.267±0.009(6)	0.437±0.005(8)	0.259±0.009(5)	0.272±0.007(7)	0.139±0.004(2)	0.212±0.005(4)
corel5k	0.328±0.005(2)	0.372±0.006(3)	0.608±0.004(5)	0.735±0.004(6)	0.758±0.003(7)	**0.267±0.004(1)**	0.562±0.007(4)	0.886±0.004(8)
Average Rank	1.818	2.182	4.909	7.636	5.909	4.091	3.545	5.909

Table 8　Comparison of Multi-label Learning Algorithms on Average-precision↑ (mean±std)

表 8　MLL 算法在 Average-precision↑ 指标(均值±标准差)上的比较

Dataset	LP(Rank)	ML(Rank)	KM(Rank)	FCM(Rank)	BR(Rank)	CLR(Rank)	ECC(Rank)	RAKEL(Rank)
cal500	0.496±0.005(2)	**0.501±0.003(1)**	0.476±0.005(3)	0.159±0.004(8)	0.300±0.005(7)	0.395±0.042(5)	0.463±0.006(4)	0.353±0.006(6)
emotions	**0.779±0.012(1)**	0.737±0.013(5)	0.760±0.010(2)	0.564±0.020(8)	0.730±0.015(6)	0.742±0.016(3)	0.740±0.021(4)	0.717±0.023(7)
medical	0.837±0.018(3)	**0.865±0.014(1)**	0.710±0.031(5)	0.112±0.013(8)	0.762±0.022(4)	0.400±0.062(7)	0.860±0.015(2)	0.700±0.234(6)
llog	0.390±0.009(2)	**0.405±0.013(1)**	0.220±0.007(4)	0.080±0.003(8)	0.215±0.009(5)	0.194±0.018(7)	0.342±0.009(3)	0.197±0.013(6)
enron	0.661±0.007(2)	**0.681±0.053(1)**	0.446±0.010(6)	0.131±0.005(8)	0.381±0.009(7)	0.610±0.008(3)	0.559±0.008(4)	0.539±0.006(5)
msra	**0.800±0.020(1)**	0.792±0.012(2)	0.771±0.011(3)	0.440±0.013(8)	0.540±0.015(7)	0.624±0.022(4)	0.567±0.048(6)	0.601±0.200(5)
image	0.775±0.009(2)	**0.824±0.006(1)**	0.650±0.086(6)	0.486±0.011(8)	0.649±0.012(7)	0.666±0.008(4)	0.685±0.008(3)	0.661±0.010(5)
scene	0.842±0.009(2)	**0.885±0.004(1)**	0.428±0.027(7)	0.422±0.013(8)	0.692±0.010(6)	0.778±0.004(3)	0.766±0.005(4)	0.713±0.008(5)
yeast	0.753±0.006(2)	**0.765±0.005(1)**	0.603±0.006(7)	0.424±0.012(8)	0.734±0.004(4)	0.730±0.003(5)	0.741±0.004(3)	0.720±0.005(6)
slashdot	0.579±0.009(4)	**0.711±0.005(1)**	0.489±0.009(5)	0.193±0.007(8)	0.427±0.014(6)	0.250±0.007(7)	0.628±0.009(2)	0.617±0.004(3)
corel5k	0.241±0.002(4)	**0.297±0.002(1)**	0.218±0.004(5)	0.027±0.004(8)	0.123±0.003(6)	0.274±0.002(2)	0.264±0.003(3)	0.122±0.004(7)
Average Rank	2.273	1.455	4.818	8	5.909	4.545	3.454	5.545

数据集上表现最好的算法的结果用黑体表示.算法在每种指标和每个数据集上的排序显示在其结果后的括号中,并统计了每种算法在所有数据集上的平均排序.

为了从统计意义上比较 8 种算法在 11 个数据集上的表现,这里采用一种由 Demšar[26] 提出的两步统计假设检验法.该方法首先对原假设"所有算法的平均排序是一样的"应用 Friedman 检验.表 9 给出了在显著水平为 0.05、对比算法数为 8、数据集数为 11 时,各个评价指标上的 Friedman 统计值 F_F 及关键值(critical value).可以看出,所有指标上的 F_F 值均大于关键值,因此在所有指标上都可以拒绝原假设,即在每个指标上所有算法的平均排序不是都一样的.在此基础上,该方法第 2 步用 Nemenyi 检验来检验算法两两之间的平均排序比较,结果可用如图 2 所示的 CD(critical difference)图来表示.Nemenyi 检验在显著水平为 0.05、对比算法数为 8、数据集数为 11 时,$CD=2.8096$.在 CD 图中,每个算法的平均排序被标注在同一条坐标轴上.如果 2 个算法的平均排序之差小于 CD 值,则说明两者没有显著差异,在 CD 图中即将这 2 个算法用一条线段连起来.

Table 9　Friedman Statistics on Each Evaluation Measure

表 9　各种评价指标上的 Friedman 统计值

Measure	F_F	Critical Value
Ranking-loss	22.914 507 77	3.135 5
One-error	28.972 392 64	3.135 5
Hamming-loss	14.634 028 11	3.135 5
Coverage	19.206 896 55	3.135 5
Average-precision	27.367 647 06	3.135 5

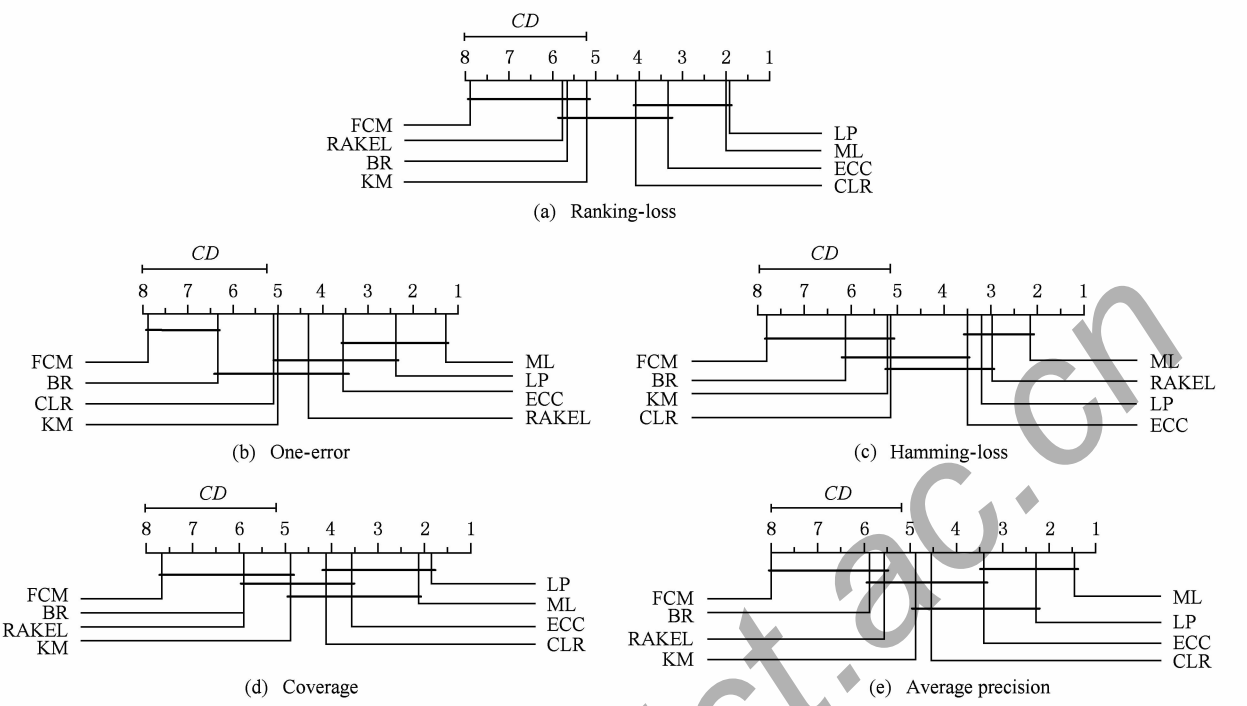

Fig. 2 *CD* diagrams (*CD*=2.809 6) of the Nemenyi tests on the eight algorithms for the five evaluation measures

图 2 在 5 个不同评价指标上对 8 种算法对比应用 Nemenyi 检验的 *CD* 图

基于表 4～8 和图 2 中的实验结果,我们可以得出以下 8 条结论:

1) 总体上,4 种标记增强算法排名大致为 LP≅ML≻KM≻FCM.

2) 在 Ranking-loss(图 2(a))和 Coverage(图 2(d))指标上,LP 的平均排序是最小的(最优的).

3) 在 One-error(图 2(b))、Hamming-loss(图 2(c))和 Average precision(图 2(e))指标上,ML 的平均排序是最小的(最优的).

4) 在 5 种评价指标中,ML 和 LP 并没有显著差异,而且这 2 种标记增强算法都显著地优于 MLL 算法 BR.

5) 在 One-error 和 Average precision 指标上,ML 显著地优于 MLL 算法 BR,CLR 和 RAKEL.

6) 在 Ranking-loss, Coverage 和 Average precision 指标上,LP 显著优于 MLL 算法 BR 和 RAKEL.

7) 在 5 种指标上,KM 优于 BR,且 KM 与 ECC,CLR,RAKEL 没有显著的差异.

8) 4 种标记增强方法中,FCM 平均排序是最差的,但与 BR 没有显著差异.

对比 3.2.1 节中的实验结果,ML 在第 2 部分实验的表现好于 KM 主要是因为这里采用的数据集的标记空间维度(标记个数)明显高于 3.2.1 节中所采用的人造数据集.这时,KM 中对标记逐个处理的手法自然使得其在标记个数较多时表现不佳.另外,总体来看,基于图模型的标记增强(LP 和 ML)平均性能要优于基于模糊方法的标记增强(KM 和 FCM),这主要是因为 KM 和 FCM 都并非为了面向标记分布的标记增强而专门设计,因此一些非针对性的做法,如模糊操作的简单运用或者对不同标记的逐个处理使得这类方法直接用于标记增强效果并不理想.当然,在一些特殊情况下,如标记个数比较少时,模糊方法也可能取得不错的表现.例如,在人造数据集上 KM 的表现就要优于 ML.

综上,可以认为,一方面,有效的标记增强(如 LP 和 ML 算法)能够显著提升传统 MLL 的效果.这说明标记增强确实有助于挖掘蕴含在训练样本中的标记间相关信息;另一方面,由于 KM 和 FCM 算法并非为标记分布专门设计,所以标记增强的效果并不理想,这也说明了针对面向标记分布的标记增强进行专门研究的必要性.

4 结 论

本文提出了面向标记分布的标记增强这一新概

念.这类方法可以从训练样本中挖掘隐藏的标记间相关性信息,将样本中原有的简单逻辑标记增强为包含更多监督信息的标记分布,从而为后续的机器学习过程提供了更多可用信息.

本文综述了可用于面向标记分布的标记增强方法.它们有些是专门为标记分布学习提出的增强方法,有些则是在其他领域(如模糊分类)提出,但可以用于实现标记增强.本文进一步通过在1个人造数据集和11个真实世界的多标记数据集上的实验,比较了已有标记增强算法的表现,显示了标记增强方法的优势.实验结果表明,建立在良好标记增强基础上的MLL算法能够取得比建立在逻辑标记基础上的传统MLL算法更好的表现,这也说明了对标记增强方法进一步深入研究的必要性.未来在标记增强方面的研究至少包括3点重要内容:

1) 建立可用于标记增强实验的标准数据集.目前适合于标记增强算法实验的数据集还非常少,本文不得不借助人造数据或者多标记数据集来进行初步实验.而适合于标记增强的专门数据集中应包括真实的标记分布以及对应的逻辑标记.

2) 建立标记增强算法的性能评价体系.该评价体系应能全面反映算法从简单逻辑标记恢复真实标记分布的能力.

3) 提出能够充分利用标记间相关性的标记增强算法.标记间相关性既可以体现在标记空间,也可能从示例空间迁移而来,这方面不论是理论层面还是应用层面都还有很大的研究空间.

参 考 文 献

[1] Gibaja E, Ventura S. A tutorial on multilabel larning [J]. ACM Computing Surveys, 2015, 47(3): 1-38

[2] Tsoumakas G, Katakis I, Vlahavas I. Mining multi-label data[G]//Data Mining and Knowledge Discovery Handbook. Berlin: Springer, 2009: 667-685

[3] Zhang Minling, Zhou Zhihua. A review on multi-Label learning algorithms [J]. IEEE Trans on Knowledge & Data Engineering, 2014, 26(8): 1819-1837

[4] Rubin N T, Chambers A, Smyth P, et al. Statistical topic models for multi-label document classification [J]. Machine Learning, 2012, 88(1): 157-208

[5] Cabral S R, Torre F D, Costeira P J, et al. Matrix completion for multi-label image classification [C] //Proc of NIPS 2011. Cambridge, MA: MIT Press, 2011: 190-198

[6] Lo H Y, Wang J C, Wang H M, et al. Cost-sensitive multi-Label learning for audio tag annotation and retrieval [J]. IEEE Trans on Multimedia, 2011, 13(3): 518-529

[7] Wang Jingdong, Zhao Yinghai, Wu Xiuqing, et al. A transductive multi-label learning approach for video concept detection [J]. Pattern Recognition, 2011, 44(10/11): 2274-2286

[8] Geng Xin. Label distribution learning [J]. IEEE Trans on Knowledge and Data Engineering, 2016, 28(7): 1734-1748

[9] Geng Xin, Xia Yu. Head pose estimation based on multivariate label distribution [C] //Proc of IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2014: 3742-3747

[10] Geng Xin, Wang Qin, Xia Yu. Facial age estimation by adaptive label distribution learning [C] //Proc of the 22nd Int Conf on Pattern Recognition. Piscataway, NJ: IEEE, 2014: 4465-4470

[11] Li Yukun, Zhang Minling, Geng Xin. Leveraging implicit relative labeling-importance information for effective multi-label learning [C] //Proc of IEEE Int Conf on Data Mining. Piscataway, NJ, IEEE, 2015: 251-260

[12] Hou Peng, Geng Xin, Zhang Minling. Multi-label manifold learning [C] //Proc of the 30th AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2016: 1680-1686

[13] Gayar N E, Schwenker F, Palm G. A study of the robustness of KNN classifiers trained using soft labels [C] //Proc of the 2nd Conf Artificial Neural Networks in Pattern Recognition. Berlin: Springer, 2006: 67-80

[14] Jiang Xiufeng, Yi Zhang, Lv Jiancheng. Fuzzy SVM with a new fuzzy membership function [J]. Neural Computing & Applications, 2006, 15(3/4): 268-276

[15] Lin Xiaotong, Chen Xuewen. Mr. KNN: Soft relevance for multi-label classification [C] //Proc of the 19th ACM Int Conf on Information and Knowledge Management. New York: ACM, 2010: 349-358

[16] Jiang J Y, Tsai C C, Lee S J. FSKNN: Multi-label text categorization based on fuzzy similarity and k nearest neighbors [J]. Expert Systems with Applications, 2012, 39(3): 2813-2821

[17] Zhu Xiaojin, Goldberg A B. Introduction to Semi-Supervised Learning [M]. Williston, VT: Morgan & Claypool, 2009

[18] Klir J G, Yuan B. Fuzzy sets and fuzzy logic [G] //Theory and Applications. Upper Saddle River, NJ: Prentice Hall, 1995

[19] Zhu Xiaojin, Lafferty J, Rosenfeld R. Semi-supervised learning with graphs [D]. Pittsburgh, PA: Language Technologies Institute, School of Computer Science, Carnegie Mellon University, 2005

[20] Read J, Pfahringer B, Holmes G, et al. Classifier chains for multi-label classification [J]. Machine Learning, 2011, 85(3): 333-359

[21] Boutell R M, Luo Jiebo, Shen Xipeng, et al. Learning multi-label scene classification [J]. Pattern Recognition, 2004, 37(9): 1757-1771

[22] Fürnkranz J, Hüllermeier E, Mencía E L, et al. Multi-label classification via calibrated label ranking [J]. Machine Learning, 2008, 73(2): 133-153

[23] Tsoumakas G, Katakis I, Vlahavas I. Random *k*-labelsets for multilabel classification [J]. IEEE Trans on Knowledge and Data Engineering, 2011, 23(7): 1079-1089

[24] Tsoumakas G, Eleftherios S, Vilcek J, et al. MULAN: A Java library for multi-label learning [J]. Journal of Machine Learning Research, 2011, 12(7): 2411-2414

[25] Zhang Minling, Zhou Zhihua. A review on multi-label learning algorithms [J]. IEEE Trans on Knowledge and Data Engineering, 2014, 26(8): 1819-1837

[26] Demšar J. Statistical comparisons of classifiers over multiple data sets [J]. Journal of Machine Learning Research, 2006, 7(1): 1-30

Geng Xin, born in 1978. PhD, professor. His main research interests include machine learning, pattern recognition, and computer vision.

Xu Ning, born in 1988. PhD candidate. His main research interests include pattern recognition and machine learning.

Shao Ruifeng, born in 1994. Master candidate. His main research interests include pattern recognition and machine learning.

《计算机研究与发展》征订启事

《计算机研究与发展》(Journal of Computer Research and Development)是中国科学院计算技术研究所和中国计算机学会联合主办、科学出版社出版的学术性刊物,中国计算机学会会刊. 主要刊登计算机科学技术领域高水平的学术论文、最新科研成果和重大应用成果. 读者对象为从事计算机研究与开发的研究人员、工程技术人员、各大专院校计算机相关专业的师生以及高新企业研发人员等.

《计算机研究与发展》于1958年创刊,是我国第一个计算机刊物,现已成为我国计算机领域权威性的学术期刊之一. 并历次被评为我国计算机类核心期刊,多次被评为"中国百种杰出学术期刊". 此外,还被《中国学术期刊文摘》、《中国科学引文索引》、"中国科学引文数据库"、"中国科技论文统计源数据库"、美国工程索引(Ei)检索系统、日本《科学技术文献速报》、俄罗斯《文摘杂志》、英国《科学文摘》(SA)等国内外重要检索机构收录.

国内邮发代号:2-654;国外发行代号:M603

国内统一连续出版物号:CN11-1777/TP

国际标准连续出版物号:ISSN1000-1239

联系方式:

100190 北京中关村科学院南路6号《计算机研究与发展》编辑部

电话: +86(10)62620696(兼传真);+86(10)62600350

Email:crad@ict.ac.cn

http://crad.ict.ac.cn