计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(8):146−152 [doi: 10.15888/j.cnki.csa.006490]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于标记相关性的多示例多标记算法①

李村合, 田程程, 姜　宇

(中国石油大学 (华东) 计算机与通信工程学院, 青岛 266580)
通讯作者: 李村合, E-mail: licunhe@qq.com

摘　要: 多示例多标记学习 (Multi-Instance Multi-Label, MIML) 是一种新的机器学习框架, 基于该框架上的样本由多个示例组成并且与多个类别相关联, 该框架因其对多义性对象具有出色的表达能力, 已成为机器学习界研究的热点. 解决 MIML 分类问题的最直接的思路是采用退化策略, 通过向多示例学习或多标记学习的退化, 将 MIML 框架下的分类问题简化为一系列的二类分类问题进行求解. 但是在退化过程中会丢失标记之间的关联信息, 降低分类的准确率. 针对此问题, 本文提出了 MIMLSVM-LOC 算法, 该算法将改进的 MIMLSVM 算法与一种局部标记相关性的方法 ML-LOC 相结合, 在训练过程中结合标记之间的关联信息进行分类. 算法首先对 MIMLSVM 算法中的 K-medoids 聚类算法进行改进, 采用的混合 Hausdorff 距离, 将每一个示例包转化为一个示例, 将 MIML 问题进行了退化. 然后采用单示例多标记的算法 ML-LOC 算法继续以后的分类工作. 在实验中, 通过与其他多示例多标记算法对比, 得出本文提出的算法取得了比其他分类算法更优的分类效果.

关键词: 多示例多标记学习; ML-LOC 算法; 标记依赖; 支持向量机

引用格式: 李村合,田程程,姜宇.基于标记相关性的多示例多标记算法.计算机系统应用,2018,27(8):146–152. http://www.c-s-a.org.cn/1003-3254/6490.html

Multi-Instance Multi-Label Algorithm Based on Label Correlation

LI Cun-He, TIAN Cheng-Cheng, JIANG Yu

(College of Computer &Communication Engineering, China University of Petroleum(East China), Qingdao 266580, China)

Abstract: Multi-Instance Multi-Label (MIML) learning is a novel machine learning framework in which an instance is described by multiple instances and associated with multiple labels. This framework has become a hot topic in the field of machine learning because of its excellent expressive ability for polysemous objects. The most direct way to solve the MIML classification problem is the degradation strategy, it takes the multiple instance learning or multiple label learning as a bridge, transforms the MIML problem into a series of binary classification problems. However, the correlation information among labels will be lost in the degradation process, which will affect the classification result. Based on these problems, this study proposes the MIMLSVM-LOC algorithm. The algorithm combines the improved MIMLSVM algorithm with a local label correlation method ML-LOC which considers the correlation information among labels in the training process. The algorithm first improves the K-medoids clustering algorithm in the MIMLSVM algorithm, and then uses the mixed Hausdorff distance to transform each instance packet into an instance, which degradate the MIML problem. Then, the ML-LOC algorithm is used to continue the classification work. In the experiment, the comparison experiment with other MIML algorithms, the result shows that the improved algorithm has better performance than other classification algorithms.

Key words: Multi-Instance Multi-Label (MIML); ML-LOC; label correlations; SVM

① 基金项目: 山东省自然科学基金 (ZR2014FQ018)
Foundation item: Natural Science Foundation of Shandong Province (ZR2014FQ018)
收稿时间: 2017-12-26; 修改时间: 2018-01-16; 采用时间: 2018-01-19; csa 在线出版时间: 2018-07-28

在我们的现实生活中, 对象往往具有丰富的含义. 例如, 一篇报道可由多个段落或章节组成, 在对其按内容进行分类时, 可将其归为“体育”、“科技”等多个类别. 如果把报道的每个段落或章节看成一个示例, 把其所属的类别看成一个标记, 那么, 只用一个示例和一个标记来表示这篇报道则太过笼统, 表示出的对象也具有概念歧义性和语义歧义性[1]. 为了同时考察这两方面的歧义性, 多示例多标记 (Multi-Instance Multi-Label, MIML)[2]学习框架应运而生. 多示例多标记学习框架已经成功应用于图像分类、文本分类、网页分类、基因序列编码[2]等问题. 解决 MIML 问题的基本方法是采用退化策略, 将 MIML 问题转化为单示例多标记或者多示例单标记, 最终将原问题变成经典的单示例单标记问题. 但是, 使用退化策略会造成标记之间联系信息的缺失, 从而影响分类的质量和效率.

针对这一问题, 文章提出了加入了标记相关性影响因子的 MIMLSVM- LOC 算法来对 MIML 问题进行求解. 改进后的算法采用混合 Hausdorff 距离进行聚类并将标记的局部相关性运用到 MIMLSVM 算法中之中, 提高了算法的分类准确率, 对原算法在退化策略实现过程中, 丢失标记之间关联信息的问题实现了优化. 具体来说, ML-LOC 算法[3]是 Huang 和周志华在研究多标记学习问题时首先提出来的, 考察的是标记的局部相关性. 其基本思想是: 若示例集合可以被分成若干组, 则同组中的示例都有一个相同的标记相关性子集. 这种方法, 可以充分的利用标记之间的联系信息来提高分类准确率, 同时减少过大的输出空间, 达到较好的分类效果.

本文的组织结构如下: 第 1 节介绍相关工作, 第 2 节具体说明算法及原理, 第 3 节给出实验数据进行论证, 第 4 节内容是总结全文, 并综合评价提出的算法.

1 相关工作

相较于以前的单示例单标记学习算法、多示例学习[4]算法和多标记学习[5]算法, 多示例多标记学习算法可以说是一种优化与改进.

目前, 在 MIML 框架下解决问题的方式主要有两种: 一种是以退化策略为理论依据, 通过进行多示例学习或多标记学习的转换, 使原 MIML 问题简化为一系列二类分类问题, 这些二类分类问题与标记空间每个类别一一对应. MIMLBOOST、MIMLSVM 和 MIMLSVM$^+$等算法都是基于退化策略解决 MIML 问题的算法. 另一种是直接在 MIML 样本的环境下进行求解, 如 D-MIMLSVM 算法和 M^{3}MIML 算法等.

周志华和 Zhang 提出了 MIMLBOOST 和 MIMLSVM 算法来解决 MIML 问题, 两者都是在退化策略的基础上进行的[6]. MIMLBOOST 算法先将 MIML 问题转化为多示例单标记问题, 然后利用多示例学习算法 MIBOOSTING[7] 进行后面的计算. MIMLSVM 算法则是应用 K-medoids[8]聚类算法将 MIML 问题转化为单示例多标记问题. 由于两者都没有考虑到实际分类中标记之间的联系信息, 使得分类的准确率下降.

Li 在进行对果蝇基因进行标注的实验时, 使 MIMLSVM$^+$ 和 E-MIMLSVM$^+$算法[9]的基本概念出现了雏形. MIMLSVM$^+$算法将 MIML 问题简化成单示例多标记问题进行求解, 并通过 SVM 分类器将标记独立出来. E-MIMLSVM$^+$算法是在 MIMLSVM$^+$算法的基础上优化改进的算法, 应用多任务技术[10−12]使 SVM 分类器中考虑了标记之间的联系信息. E-MIMLSVM$^+$算法的分类质量明显较高, 随之付出的代价是训练时间增多, 并且存储空间加大.

D-MIMLSVM[2]算法和 M^{3}MIML[13]算法是由周志华和 Zhang 提出的基于正则化机制及最大化间隔策略的算法, 皆为直接对 MIML 问题进行求解的典型算法. D-MIMLSVM 算法假设相同样本的标记之间有一定的信息关系, 基于正则化机制对 MIML 问题进行求解. M^{3}MIML 算法是基于最大化间隔策略得到一个二次规划问题, 若分类系统包含T个线性模型, 它们与多个可能的概念类一一对应. 每个示例在其相应的线性模型上的最大输出值, 决定测试示例样本的分类. D-MIMLSVM 算法和 M^{3}MIML 算法都适应于小型数据集[14].

ML-LOC 算法[3]是 Huang 和周志华在研究多标记学习问题时首先提出来的, 考察的是标记的局部相关性. ML-LOC 算法在每一个示例中添加了一个蕴含标记相关性的向量作为 LOC 编码, 拓展了原始的特征向量. 其原理是因为相似的示例具有相似的标记相关性子集, 因此相似示例的 LOC 编码也就越相似. 因此本文首先对 MIMLSVM 算法中的聚类算法进行了改进, 之后结合 ML-LOC 算法对改进聚类方式后的 MIMLSVM 算法进行训练求解.

2 改进的算法

2.1 改进的 MIMLSVM 算法

MIMLSVM 算法[1]是一种退化算法, 该算法将 MIML 问题退化为单示例多标记问题进行求解. 给定 MIML 训练样本集$D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)\}$, 其中, n为训练样本集样本的个数, x_i是训练集中的第i个示例包, $Y_i\in Y$是与X_i相关的类别标记的集合$\left\{y_1^{(i)},y_2^{(i)},\cdots,y_{l_i}^{(i)}\right\}$. 示例包的集合可以表示为$\Gamma=\{X_i|i=1,2,\cdots,m\}$. MIMLSVM 算法使用构造性聚类的方法将多示例多标记样本转化为单示例多标记样本. 在原始 MIMLSVM 算法中, 每个示例包看作是原子对象, 使用 k-medoids 聚类将示例空间Γ划分为k份, 聚类个数k是事先确定好的[10]. k-medoids 聚类中使用了 Hausdorff 距离来度量包之间的距离, Hausdorff 距离的定义如下:

$$d_h(A,B)=\max\left\{\max_{a\in A}\min_{b\in B}\|a-b\|,\max_{b\in B}\min_{a\in A}\|a-b\|\right\} \quad (1)$$

其中,a和b分别是包$A=\{a_1,a_2,\cdots,a_{n_a}\}$、$B=\left\{b_1,b_2,\cdots,b_{n_b}\right\}$中的示例, $|a-b|$是这两个示例之间的欧式距离. 考虑到在基于包间距离测量方法中, 由于最大 Hausdorff 距离对孤立点敏感, 而最小 Hausdorff 距离只考虑到包间距离最近的 2 个示例. 因此, 从均衡性角度考虑, 本文采用文献[6]中提出的混合 Hausdorff 距离$mixH(A,B)$对 MIMLSVM 算法进行改进. 公式如下:

$$d_h(A,B)=mixH(A,B)=\frac{\sum_{a\in A}\min_{b\in B}\|a-b\|+\sum_{b\in B}\min_{a\in A}\|b-a\|}{|A|+|B|} \quad (2)$$

其中, $|\cdot|$为测量集合的势. $mixH(A,B)$考虑到了包间示例的几何关系.

上述聚类过程完成后, 示例空间Γ已经被划分为k份, 每一份的聚类中心可以用$M_t(t=1,2,\cdots,k)$来表示. 接下来计算每个包X_i到聚类中心M_t的 Hausdorff 距离$d_h(X_i,M_t)(t=1,2,\cdots,k)$, 使用这$k$个距离可以构成一个向量$z_i$, 即$z_i$的第$t$维是包$X_i$和第$t(t=1,2,\cdots,k)$个聚类中心的距离. 可以看出来, 向量$z_i$编码了示例包$X_i$的一些空间结构分布信息, 因此包$X_i$可以用$z_i$来表示, 即包$X_i$转化为了示例$z_i$, 这个转化过程称之为构造性聚类转化. 这样多示例多标记样本集$(X_i,Y_i)(i=1,2,\cdots,m)$就变成了单示例多标记样本集$(z_i,Y_i)(i=1,2,\cdots,m)$, 其中$\Phi(X_i)=z_i$, $\Phi(x)$表示的是使用构造性聚类将包X_i变为示例z_i. 注意到在转换时, 会丢失示例和标记之间的联系信息.

在多示例多标记样本集向单示例多标记样本集转化完成后, MIMLSVM 算法使用多标记学习中的 MLSVM 算法[13]对样本集进行训练. MLSVM 算法将多标记问题分解为$|y|$个二类分类问题, 从而为每一个类别标记建立一个 SVM 分类器. 事实上, 示例z_i参与了每一个分类器的构建, $\varphi(z_i,y)=+1$仅当$y\in Y_i$即示例z_i对应的类别标记集合Y_i包含y时, 否则$\varphi(z_i,y)=-1$. 至此, 多标记学习转化为传统的监督学习问题, 可以使用标准的 SVM 来进行训练分类了[14].

在为每个标记建立完分类器后, 对于一个未知标记的样本, 根据 T-criterion[15]方法, 由该样本在标记的分类器上的所得值决定该样本的标记所属类别.

2.2 ML-LOC 算法

ML-LOC 算法是由 Huang S.J.和 Zhou Z.H.提出并命名的, 是对多标记学习算法的改进, 旨在对标记局部相关性进行考察. 其主要思想是将示例样本集划分成若干组, 并且同一组内的示例具有一致的标记相关性子集. 同时减少过大的输出空间, 将 ML-LOC 算法与改进的 MIMLSVM 算法结合后, 不仅可以使标记的作用信息更加完全的发挥作用, 使分类更加精准高效, 同时能够精简输出量, 优化空间存储[6].

首先给定一个训练样本集 D, 令 D 表示为$D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)\}$, 且$x_i\in X=R^d$是特征向量, d 是特征空间维度数. y_i是x_i对应的标记向量且$y_i=[y_{il},y_{i2},\cdots,y_{in}]$, 若示例$x_i$具有第$l$个标记那么$y_{il}=1$, 否则$y_{il}=-1$. ML-LOC 算法使示例多了一个 LOC 编码信息c_i, 其在每个示例x_i各不相同, 其内容是标记的相关性, 且作为一个向量存在, 使初始的特征向量得到了拓展. 由于同组的示例具有一致的标记相关性, 故其 LOC 编码也基本一致. 我们假设, 分类函数$f=[f_1,f_2,\cdots f_L]$内有 L 个函数, 意味着每个类别标记对应一个分类函数, 并且其中每一个函数f_i都是一个线性模型:

$$f_l(x,c)=\langle w_l,[\Phi(x),c]\rangle=\left\langle w_l^x,\Phi(x)\right\rangle+\left\langle w_l^c,c\right\rangle \quad (3)$$

在这里, $\Phi(x)$是核函数, 示例x被从低维特征空间映射到高维特征空间, $w_l=[w_l,w_l]$是权重向量, w_l和w_l与特征空间中的示例x和向量c一一对应. ML-LOC 算法的优化目标是:

$$\min_{f,c}\sum_{i=1}^{n}V(x_i,c_i,y_i,f)+\lambda_1\Omega(f)+\lambda_2Z(C) \quad (4)$$

其中, V是损失函数, 是基于训练集的数据, 具体使用的为 hamming loss, 其研究的是基于单个标记上的误分次数, 其定义如式 (5):

$$V(x_i,c_i,y_i,f)=\sum_{i=1}^{L} loss(x_i,c_i,y_{il},f_l) \tag{5}$$

其中, $loss(x,c,y,f)=\max\{0,1-yf([\Phi(x),c])\}$就是 SVM 中的 hamming loss. 式 (4) 中的第二项Ω函数是一个与模型复杂度相关的正则化因子, 具体表示如式 (6):

$$\Omega(f)=\sum_{l=1}^{L}\|w_l\|^2 \tag{6}$$

式 (4) 中的Z部分也是一个正则化因子, 可以实现对局部标记性编码C的加强, 即认为相似的示例具有相似的编码. 而λ_1、λ_2作为平衡因子, 作用是使函数的三部分趋于平衡.

使用 k-means 聚类函数把训练样本集划分成m组$\{G_1,G_2,\cdots,G_m\}$, 同组的示例拥有同样的标记相关性子集, 即G_i中的示例具有相同的标记相关性子集S_i. 要得到标记相关性过程十分繁琐, S_i要通过求G_i中全部示例的标记平均值p_j来求得, p_j可表示为:

$$p_j=\frac{1}{|C_i|}\sum\nolimits_{x_k\in G_i} y_k \tag{7}$$

任意一个示例x_i, c_i指的是相关性子集S_i对示例x_i的影响系数, 则每一个示例x_i的 LOC 编码为$c_i=\{c_{i1},c_{i2},\cdots,c_{im}\}$. 那么显然, y_i与p_i的相似度, 与S_i对示例x_i的影响程度成正比, 即与c_{ij}的值成正比. 基于对标记局部相关性研究, 我们定义编码上的正则化因子为:

$$Z(c)=\sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}\|y_i-p_i\|^2 \tag{8}$$

那么最终可以得出 ML-LOC 算法模型为:

$$\begin{aligned} &Z(c)=\sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}\|y_i-p_i\|^2\\ &\text{s.t. } y_{il}\langle w_i,[\varphi(x),c_i]\rangle\geqslant 1-\xi_{il}\\ &\xi_{il}\geqslant 0,\ \forall i\in\{1,2,\cdots,n\}\\ &\sum_{j=1}^{m} c_{ij}=1,\ \forall i\in\{1,2,\cdots,n\}\\ &0\leqslant c_{ij}\leqslant 1,\ \forall i\in\{1,2,\cdots,n\},\ j\in\{1,2,\cdots,m\} \end{aligned} \tag{9}$$

可以看到, c_{ij}的取值范围是在$[0,1]$之间, 且 LOC 编码$c_i=\{c_{i1},c_{i2},\cdots,c_{im}\}$中所有值加和值为 1. 改进的 MIMLSVM 算法是基于多示例多标记框架提出的模型, 通过聚类将每一个包转化为一个示例, 将 MIML 问题简化为了单示例多标记问题. 而由于 ML-LOC 算法就是基于单示例多标记的算法, 因此在使用改进的 MIMLSVM 将问题转化之后, 再用 ML-LOC 算法继续以后的工作. MIMLSVM-LOC 算法既是对 MIMLSVM 算法的改进优化, 又考虑到标记局部相关性的特性, 使得分类结果更加完善精准[16–18].

2.3 改进算法的流程

MIMLSVM-LOC 算法是基于退化策略的多示例多标记 SVM 算法. 该算法首先使用改进的 k-medoids 聚类算法得到聚类中心$M_t(t=1,2,\cdots,k)$, 使用M_t将多示例多标记样本集(X_u,Y_u)转化为单示例多标记样本集(Z_u,Z_u), 然后使用 ML-LOC 算法分类模型继续求解. MIMLSVM-LOC 算法的流程如下:

将每一个包看作是原子对象, 则示例空间可划分为$\Gamma=\{X_i|i=1,2,\cdots,m\}$.

(1) 在Γ上使用 k-medoids 聚类将Γ分为k份, 每一份的聚类中心用$M_t(t=1,2,\cdots,k)$来表示.

1) 随机选取k个点$M_t(t=1,2,\cdots,k)$, 给聚类中心赋初值.

2) 采用公式 (2) 中的混合 Hausdorff 距离$mixH(A,B)$, 将剩余示例包$X_u\in(\Gamma-M_t(t=1,2,\cdots,k))$分配到各个聚类中去: $index=\arg\min\limits_{t\in\{1,\cdots,k\}} d_H(X_u,M_t),\Gamma_{index}\cup\{X_u\}$.

3) 重新计算聚类中心.

$$M_t=\arg\min_{A\in\Gamma}\sum_{B\in\Gamma} d_M(A,B)(t=1,2,\cdots,K)$$

4) 重复 2) 和 3) 的过程,直至M_t不变.

(2) 得到每个包X_i到聚类中心M_t的 Hausdorff 值, 使用这k个距离作为一个示例, 从而将多示例多标记样本集(X_u,Y_u)转化为单示例多标记样本集(Z_u,Z_u), 其中,

$$\begin{aligned} z_u&=(z_{u1},z_{u2},\cdots,z_{uk})\\ &=(d_h(X_u,M_1),d_h(X_u,M_2),\cdots,d_h(X_u,M_k))^{\cdot} \end{aligned}$$

(3) 对于标记$y\in Y$, 得到训练集$D_y=\{(z_u,\varphi(z_u,y))\}$, 使用 ML-LOC 算法对该多标记问题进行求解得到分类函数$f_y=ML-LOCTrain\left(D_y,\lambda_1,\lambda_2,m\right)$.

1) 通过 k-means 聚类[8]将D_y分成m组并为求解式 (8) 完成初始化.

2) 使用交替优化方法求解式 (9), 即固定其中的两个变量不变, 优化剩余的一个变量. 由于式 (9) 是下界有界的, 它将收敛达到最低限度.

3) 建立m个回归模型, 用于对未知标记示例求解 LOC 编码.

(4) 对于未知标记的样本集X, 使用构造性聚类将其转化为单示例样本集$Z_t=(d_h(X^*,M_1),d_h(X^*,M_2),\cdots,(d_h(X^*,M_k)))$, 然后使用训练出来的回归模型计算 LOC 编码$\left(c_1^*,c_2^*,\cdots,c_m^*\right)$, 得到单示例测试样本集.

(5) 使用训练出来的分类函数进行预测:

$$Y^*=\left\{\begin{matrix}\arg\max\\ y\in L\end{matrix} f_y(Z^*)|f_y(z^*)<0,\forall y\in L\right\}\cup\left\{y|f_y(z^*)\geqslant 0,y\in L\right\}$$

3 实验

本文实验所用电脑配置 Windows 7 操作系统, Intel 酷睿 i7 处理器, 4 G 运行内存和 128 G 固态硬盘, 运行环境为 Matlab R2014b.

3.1 数据集

本文实验数据来自南京大学周志华等人提供的图像数据集与文本数据集[19]. 这两个数据集中的样本都是 MIML 样本, 即数据集中的每个样本由多个示例表示其概念, 并且由多个标记来标识其所属语义范畴.

图像数据集包含 2000 个场景图像, 每个图像都被分配一组标记. 共预定义 5 个类别, 分别是日落、沙漠、树、山以及海洋, 其中, 单标记样本数目约占 77% 左右, 双标记样本约占 22% 左右, 三标记样本数约占 0.75% 左右. 在该数据集中, 每幅图片所对应的 MIML 样本的示例数为 9, 每一个示例用一个 15 维的特征向量表示.

文本数据集主要包括常见的七大类别. 该数据集从 Reuters 中移除无类别或无正文的文本和单类别文本, 随机选取部分样本, 获得 2000 个文档. 利用滑动窗口技术[14]将每篇文档表示为一个示例包, 包中的每个示例对应文档中的大小为 50 的滑动窗口所包围的一段文本片段. 包中的示例采取基于词频的词袋模型进行表示, 利用降维的方式, 将词频为前 2% 的词汇予以保留, 最终包中的所有示例都可以用 243 维的特征向量进行表示. 表 1 总结了图像样本集 (Scene) 和文本样本集 (Reuters) 的特征.

表 1 样本集特征

样本集	样本数	类别数目	实例数		样本所属标记数 (k)			训练集样本数	测试集样本数
			Min	Max	k=1	k=2	k≥3		
Scene	2000	5	9	9	1543	442	15	1000	1000
Reuters	2000	7	2	26	1701	290	9	1000	1000

3.2 评价指标

本文采用多示例多标记领域的五个评价指标 *hamming loss*、*one-error*、*coverage*、*ranking loss* 和 *average precision* 进行评价. 设数据集$S=\{(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)\}$.

(1) *hamming loss*, 定义如下:

$$hloss_s(h)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|y|}|h(X_i)\Delta Y_i| \tag{10}$$

其中, Δ表示两个集合之间的对称差, *hamming loss* 评价的是样本在标记相同上的分类误差值, 结果值小, 则说明算法表现更好.

(2) *one-error*, 定义如下:

$$one-error_s(h)=\frac{1}{n}\sum_{i=1}^{n}\left\{\left[\arg\max_{y\in Y}h(X_i,y)\notin Y_i\right]\right\} \tag{11}$$

其评价的主要内容是, 统计所考察样本的标记序列中, 最前端的标记不属于样本集合的情况, 同样的, 结果值小说明算法表现更优.

(3) *coverage*, 定义如下:

$$\mathrm{cov}erage_s(h)=\frac{1}{n}\sum_{i=1}^{n}\max_{y\in Y_i}rank^h(X_i,y)-1 \tag{12}$$

coverage 用于评价样本的标记集中, 覆盖某样本所有标记所用的搜索深度, 结果值小, 对应的算法效果更优.

(4) *ranking loss*, 定义如下:

$$rloss_s(h)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|Y_i|}|R_i| \tag{13}$$

其评价的内容是标记序列发生误排序的情况, 同以上四种标准, 结果值小则算法效果越好.

(5) *average precision*, 定义如下:

$$avgpre_s(h)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|Y_i|}\frac{|P_i|}{rank^h(X_i,l)} \quad (14)$$

其中, $p_i=\{l'|rank^h(X_i,l')\leqslant rank^h(X_i,l),l'\in Y_i\}$其主要用来评价样本的标记序列中的排序性能, 结果值大, 表示算法效果更优. 总的来说, 对于 *hamming loss*、*one-error*、*coverage*、*ranking loss* 而言, 取值小说明算法效果更好, 而对 *average precision* 取值大表示算法效果更好.

3.3 实验算法

本文将提出的 MIMLSVM-LOC 算法与 MIMLSVM$^+$ 算法、MIMLBOOST 算法和 MIMLSVM 算法进行对比. 这三个算法都是基于退化策略的算法, 其中, MIMLSVM 算法和 MIMLSVM$^+$算法将 MIML 问题转化为单示例多标记问题, 即为每一个标记建立一个 SVM 分类器进行求解, MIMLBOOST 算法将 MIML 问题多示例简化成单标记问题, 最终利用多示例学习算法 MIBOOSTING[7]解决问题.

MIMLBOOST 和 MIMLSVM 算法的参数根据文献[10]的实验设置为它们的最佳值, 即将 MIMLBOOST 的 boosting rounds 的值设为 25, MIMLSVM 的高斯核为$\gamma=0.2^2$, 聚类个数k为训练个数的 20%. MIMLSVM$^+$ 的核函数的两个参数分别为$\gamma_1=10^{-5}$和$\gamma_2=10^{-2}$. 根据文献[3]中对 ML-LOC 算法的实验, 设置$\lambda_1=1$、$\lambda_2=2$、$m=5$. 实验采用 3.1 小节所介绍的数据集, 采用 10 折交叉验证的方式, 重复 10 次实验之后求得实验的平均值及方差.

3.4 实验结果

表 2 和表 3 分别显示了四种不同的 MIML 算法在图像数据集、文本数据集上的实验数据值. 在表 3 和表 4 中, 黑体部分是 MIMLSVM-LOC 算法的实验结果. 从表中数据可以分析, MIMLSVM 算法的性能总体上优于 MIMLBOOST 算法, 这是由于训练集中具有两个以上类别标记的样本较少, 使得 MIMLBOOST 在从 MIML 样本简化为多示例单标记样本的过程中出现类别不平衡的问题, 影响了分类效果. MIMLSVM$^+$算法的性能总体上优于 MIMLSVM 算法, MIMLSVM-LOC 算法的性能总体上优于 MIMLSVM$^+$算法, 可见, 增加了标记之间的关联关系可以提高分类的效果. 因此, MIMLSVM-LOC 算法在两种多示例多标记学习任务中的总体表现要优于其它 MIML 分类算法.

表 2 场景样本集上实验结果

Metric	MIMLSVM-LOC	MIMLSVM$^+$	MIML-BOOST	MIMLSVM
hamming-loss	0.190±0.011	0.198±0.004	0.229±0.022	0.194±0.005
one-error	0.330±0.011	0.354±0.011	0.417±0.009	0.386±0.016
coverage	0.958±0.048	1.091±0.045	0.960±0.014	1.034±0.003
ranking-loss	0.180±0.010	0.200±0.010	0.203±0.005	0.217±0.012
average-precision	0.793±0.009	0.769±0.008	0.771±0.012	0.750±0.017

表 3 文本样本集上实验结果

Metric	MIMLSVM-LOC	MIMLSVM$^+$	MIML-BOOST	MIMLSVM
hamming-loss	0.027±0.001	0.033±0.002	0.176±0.002	0.170±0.007
one-error	0.050±0.002	0.060±0.002	0.540±0.008	0.525±0.022
coverage	0.265±0.005	0.27±0.010	1.553±0.054	1.521±0.071
ranking-loss	0.017±0.001	0.02±0.003	0.130±0.005	0.135±0.024
average-precision	0.968±0.004	0.96±0.003	0.658±0.006	0.667±0.013

表 4 各算法在两个数据集上的训练时间

		MIMLSVM-LOC	MIML-SVM$^+$	MIML-BOOST	MIMLSVM
Training (minutes)	Scene	9.27±0.07	6.64±0.32	3009.85±59.17	9.18±0.38
	Reuters	5.46±0.04	0.93±0.02	2994.18±42.26	5.32±0.06

表 4 列举了四种算法在两个数据集上所用时间值观察表 4 可知, 黑体部分 MIMLSVM-LOC 算法耗时比 MIMLSVM$^+$算法略多, 与 MIMLSVM 算法基本持平, 远远小于 MIMLBOOST 算法, 因此 MIMLSVM$^+$算法耗时最短, MIMLSVM-LOC 和 MIMLSVM 算法总体耗时相差不大, 但是分类效果却优于 MIMLSVM 算法. 而 MIMLBOOST 在文本样本集耗时虽较图像样本集有所减少, 但仍远远高于其他算法. 综上所述, MIMLSVM-LOC 算法在图像和文本数据集上, 整体表现优异, 比较适用于解决多示例多标记问题.

4 总结

针对退化策略中具有的标记关联信息丢失的问题, 本文提出了改进的 MIMLSVM 算法并将改进的算法与 ML-LOC 算法相结合, 提出 MIMLSVM-LOC. 该算法采用混合 Hausdorff 距离进行聚类并借鉴 ML-LOC 算法的思想, 将示例样本集划分成若干组, 并且同一组内的示例具有一致的标记相关性子集, 减少过大的输出空间. 改进后的 MIMLSVM-LOC 算法, 不仅考虑到示例的几何关系同时也考虑到了标记的作用信息, 使分类更加精准高效, 同时能够精简输出量, 优化空间存储.

实验数据说明 MIMLSVM-LOC 算法的表现优于其它 MIML 分类算法, 比较适合应用于多示例多标记问题. 但是, 提出的方法仍存在待改进的地方:

(1) 参数取值问题. 本文中, 参数选取的值多为经验参数, 缺乏理论的指导. 参数的取值对于实验结果具有一定的影响, 因此, 如何使参数取值更加恰当是一个亟待解决的问题.

(2) 核函数选取问题. 本文实验中选用多示例核函数, 在实际应用中, 由于样本的不同, 核函数的选取也会影响实验的结果. 因此, 如何动态选取核函数, 仍是值得研究的方向.

参考文献

1 Zhou ZH, Zhang ML, Huang SJ, *et al*. MIML: A framework for learning with ambiguous objects. arXiv: 0808.3231, 2012.

2 Zhou ZH, Zhang ML, Huang SJ, *et al*. Multi-instance multi-label learning. Artificial Intelligence, 2012, 176(1): 2291–2320. [doi: 10.1016/j.artint.2011.10.002]

3 Huang SJ, Zhou ZH. Multi-label learning by exploiting label correlations locally. Proceedings of the 26th AAAI Conference on Artificial Intelligence. Toronto, ON, Canada. 2012. 949–955.

4 Zhou ZH, Zhang ML. Solving multi-instance problems with classifier ensemble based on constructive clustering. Knowledge and Information Systems, 2007, 11(2): 155–170. [doi: 10.1007/s10115-006-0029-3]

5 Osojnik A, Panov P, Džeroski S. Multi-label classification via multi-target regression on data streams. Machine Learning, 2017, 106(6): 745–770. [doi: 10.1007/s10994-016-5613-5]

6 谢红薇, 李晓亮. 基于多示例的 K-means 聚类学习算法. 计算机工程, 2009, 35(22): 179–181. [doi: 10.3969/j.issn.1000-3428.2009.22.061]

7 Wang JZ. Semi-supervised learning using ensembles of multiple 1D-embedding-based label boosting. International Journal of Wavelets, Multiresolution and Information Processing, 2016, 14(2): 1640001. [doi: 10.1142/S0219691316400014]

8 Arora P, Dr D, Varshney S. Analysis of K-means and K-medoids algorithm for big data. Procedia Computer Science, 2016, 78: 507–512. [doi: 10.1016/j.procs.2016.02.095]

9 Li YX, Ji SW, Kumar S, *et al*. Drosophila gene expression pattern annotation through multi-instance multi-label learning. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2012, 9(1): 98–112. [doi: 10.1109/TCBB.2011.73]

10 Zhou ZH, Zhang ML. Multi-instance multi-label learning with application to scene classification. Proceedings of the 19th International Conference on Neural Information Processing Systems. Canada. 2007. 1609–1616.

11 Su C, Yang F, Zhang SL, *et al*. Multi-task learning with low rank attribute embedding for person Re-identification. Proceedings of 2015 IEEE International Conference on Computer Vision. Santiago, Chile. 2015. 3739–3747.

12 Evgeniou T, Micchelli C A, Pontil M. Learning multiple tasks with kernel methods. The Journal of Machine Learning Research, 2005, 6: 615–637.

13 Boutell MR, Luo JB, Shen XP, *et al*. Learning multi-label scene classification. Pattern Recognition, 2004, 37(9): 1757–1771. [doi: 10.1016/j.patcog.2004.03.009]

14 Andrews S, Tsochantaridis I, Hofmann T. Support vector machines for multiple-instance learning. Advances in Neural Information Processing Systems, 2003, 15(2): 561–568.

15 Tong S, Koller D. Support vector machine active learning with applications to text classification. The Journal of Machine Learning Research, 2002, 2: 45–66.

16 Teisseyre P. CCnet: Joint multi-label classification and feature selection using classifier chains and elastic net regularization. Neurocomputing, 2017, 235: 98–111. [doi: 10.1016/j.neucom.2017.01.004]

17 Li CH, Zhang YL, Lu L. An MIMLSVM algorithm based on ECC. Applied Intelligence, 2015, 42(3): 537–543. [doi: 10.1007/s10489-014-0608-z]

18 Briggs F, Fern XZ, Raich R. Context-aware MIML instance annotation: Exploiting label correlations with classifier chains. Knowledge and Information Systems, 2015, 43(1): 53–79. [doi: 10.1007/s10115-014-0781-8]

19 Sebastiani F. Machine learning in automated text categorization. ACM Computing Surveys, 2002, 34(1): 1–47. [doi: 10.1145/505282.505283]