

基于互信息的粒化特征加权多标签学习 k 近邻算法

李 峰 苗 夺 谦 张 志 飞 张 维

(同济大学计算机科学与技术系 上海 201804)

(嵌入式系统与服务计算教育部重点实验室(同济大学) 上海 201804)

(tjleefeng@163.com)

Mutual Information Based Granular Feature Weighted k -Nearest Neighbors Algorithm for Multi-Label Learning

Li Feng, Miao Duoqian, Zhang Zhifei, and Zhang Wei

(Department of Computer Science and Technology, Tongji University, Shanghai 201804)

(Key Laboratory of Embedded Systems and Service Computing (Tongji University), Ministry of Education, Shanghai 201804)

Abstract All features contribute equally to compute the distance between any pair of instances when finding the nearest neighbors in traditional k NN based multi-label learning algorithms. Furthermore, most of these algorithms transform the multi-label problem into a set of single-label binary problems, which ignore the label correlation. The performance of multi-label learning algorithm greatly depends on the input features, and different features contain different knowledge about the label classification, so the features should be given different importance. Mutual information is one of the widely used measures of dependency of variables, and can evaluate the knowledge contained in the feature about the label classification. Therefore, we propose a granular feature weighted k -nearest neighbors algorithm for multi-label learning based on mutual information, which gives the feature weights according to the knowledge contained in the feature. The proposed algorithm firstly granulates the label space into several label information granules to avoid the problem of label combination explosion problem, and then calculates feature weights for each label information granule, which takes label combinations into consideration to merge label correlations into feature weights. The experimental results show that the proposed algorithm can achieve better performance than other common multi-label learning algorithms.

Key words mutual information; feature weight; granulation; multi-label learning; k -nearest neighbors

摘要 传统基于 k 近邻的多标签学习算法,在寻找近邻度量样本间的距离时,对所有特征给予同等的重要性。这些算法大多采用分解策略,对单个标签独立预测,忽略了标签间的相关性。多标签学习算法的分类效果跟输入的特征有很大的关系,不同的特征含有的标签分类信息不同,故不同特征的重要性也不

收稿日期:2016-05-24;修回日期:2016-09-06

基金项目:国家自然科学基金项目(61273304,61573255);高等学校博士学科点专项科研基金项目(20130072130004);上海市自然科学基金项目(14ZR1442600)

This work was supported by the National Natural Science Foundation of China (61273304, 61573255), the Specialized Research Fund for the Doctoral Program of Higher Education of China (20130072130004), and the Natural Science Foundation of Shanghai (14ZR1442600).

通信作者:苗 夺 谦(dqmiao@tongji.edu.cn)

同. 互信息是常用的度量 2 个变量间关联度的重要方法之一, 能够有效度量特征含有标签分类的知识量. 因此, 根据特征含有标签分类知识量的大小, 赋予相应的权重系数, 提出一种基于互信息的粒化特征加权多标签学习 k 近邻算法 (granular feature weighted k -nearest neighbors algorithm for multi-label learning, GFWML- k NN), 该算法将标签空间粒化成多个标签粒, 对每个标签粒计算特征的权重系数, 以解决上述问题和标签组合爆炸问题. 在计算特征权重时, 考虑到了标签间可能的组合, 把标签间的相关性融合进特征的权重系数. 实验表明: 相较于若干经典的多标签学习算法, 所提算法 GFWML- k NN 整体上能取得较好的效果.

关键词 互信息; 特征权重; 粒化; 多标签学习; k -近邻

中图法分类号 TP18

传统机器学习以二类 (binary-class) 分类或者多类 (multi-class) 分类为主, 每个样本只有一个类别标签, 称作单标签学习 (single-label learning). 然而, 现实世界中各领域都存在大量同时拥有多个标签的样本^[1-3]. 比如, 一篇新闻报道可能同时跟多个话题有关^[4], 如体育、娱乐、社会、经济; 一张图片可能同时包含多个语义信息^[5], 如大海、沙滩、城市; 一个基因可能同时具有多个功能^[6], 如翻译、转录; 一个病人的病理图像可能同时展现出了多个病理特性^[7], 如角化过度、颗粒层消失. 这些有多个标签的样本称为多标签数据. 多标签学习 (multi-label learning) 的任务就是通过学习标签已知的多标签数据来预测标签未知的样本相关的多个标签^[1-3]. 相较于单标签数据中样本只有唯一确定的标签, 多标签数据中的样本可能同时拥有多个标签, 从而导致了多标签数据的复杂多变. 多标签数据的复杂性、多变性都增加了多标签学习算法预测的难度.

在面对复杂问题时, 人类往往将复杂问题分解为多个简单问题, 再逐个解决. 机器学习领域沿用了此方法, 传统机器学习中将多类分类问题拆分成多个二类分类问题. 直观上, 多标签学习问题也可以被拆分成多个独立的二类分类问题, 每一个标签对应一个二类分类问题. 但此方法破坏了标签间的相关性, 而多标签数据中的标签间往往存在一定的相关性, 如一部“动作”电影, 同时也是一部“冒险”电影的可能性, 要比它同时是一部“爱情”电影的可能性要大. 因此如何充分利用多标签训练数据中标签间的相关性 (label correlation) 来帮助构建预测模型一直是多标签学习中的研究热点问题之一.

众所周知, k 近邻算法^[8] (k -nearest neighbors, k NN) 是一种无参懒惰学习算法, 无需通过训练来构建预测模型, 待测样本的预测结果直接由其 k 个距离最近的训练样本来确定. k NN 算法是机器学习

领域最简单有效的算法之一, 经常被用于分类或者回归. 因此, 一些基于 k NN 算法的扩展算法被提出用于多标签学习^[9-12]. 像 k NN 算法一样, 这些扩展算法也是懒惰学习算法, 均需先查找样本的 k 个最近邻训练样本, 再采用不同的后续处理方式. 最近邻的选择依据样本间在特征空间上的距离, 所选择的 k 个最近邻训练样本直接决定算法效果. 在处理多标签问题时, 这些算法大多采用了前述的拆分策略, 每个标签单独求解, 忽略了标签间的相关性.

标签组合是挖掘标签相关性的有效方式, 但标签组合数跟标签数呈指数级的关系, 其随着标签数的增长而急剧增加, 加剧了算法的复杂度, 而且多标签数据的标签空间中有的标签相关性很大, 有的则几乎没有相关性. 因此相关性较大的标签应该被放到同一类, 分别考虑各类中的标签相关性.

像传统机器学习算法一样, 多标签学习算法的分类预测效果依赖于输入的特征, 而且相较于传统机器学习算法中只有一个类别, 多标签学习中的有多个标签类别, 特征与标签的相关性更为复杂. 不同的特征对同一标签的分类有不同的重要度, 同一特征对不同的标签也有着不同的重要度. 有的特征对某一类标签分类很重要, 对另一类标签则不是那么重要, 甚至完全不相关. 所以不同的特征应该根据其对标签分类的重要程度被给予相应的权重. 但 k NN 算法在计算样本间距离时, 将所有特征给予同样的重要度. 不重要的特征被同等对待, 会干扰近邻的选择, 影响算法效果.

互信息是常用的度量 2 个变量关联度的方法, 其能表示特征对标签分类的重要度. 粒计算是解决复杂问题的有效方法, 其模拟人类处理复杂问题的方式, 将复杂问题化解成简单问题^[13-14]. 因此, 我们提出了一种基于互信息的粒化特征加权多标签学习 k 近邻算法 (granular feature weighted k -nearest

neighbors algorithm for multi-label learning, GFWML- k NN), 该算法采用粒计算思想将标签空间粒化成多个标签粒, 将相似的标签粒化到同一标签粒中, 使得不同粒中的标签间的相关性最小, 粒内的标签间相关性最大。这样既将复杂问题分解成为了多个简单问题, 大大减少标签组合数, 又最大限度地保留了标签间的相关性。对每个标签粒, 根据特征包含的标签分类信息程度, 对不同的特征给予不同的权重, 将标签间的相关性融入特征的权重系数。对特征进行加权, 寻找与待测样本标签信息更接近的近邻样本, 提高算法的精度。

为了验证所提算法的有效性, 本文将 GFWML- k NN 与标准 ML- k NN 算法, 以及其他经典的多标签算法在多个真实世界多标签数据集上进行了实验对比, 实验结果表明本文所提算法能取得较好的学习效果。

1 研究现状

多标签学习已经受到了越来越多的关注, 成为机器学习领域中的一个研究热点。目前为止, 一系列多标签学习算法已经被提出。这些多标签算法主要可以分为 2 类: 问题转换方法(problem transformation method, PTM)和算法适应方法(algorithm adaption method, AAM)^[1]。

问题转换方法首先将多标签数据集转换成多个单标签数据集, 再采用已有的单标签学习算法来处理每个标签数据。问题转换方法独立于算法, 让数据去适应算法。经典的问题转换方法有 BR(binary relevance)^[15]方法、LP(label powerset)方法等。

BR^[15]方法直接将多标签数据集中的每个标签拆分开来, 形成 $|L|$ 个独立的单标签数据集, $|L|$ 表示标签的数量。BR 方法虽然简单, 但其忽略标签间的相关性。考虑到标签的相关性, Read 等人^[16]提出了一种对 BR 方法的改进算法, 链式的分类(chain classifier, CC)算法用于多标签学习。该算法同样构建 $|L|$ 个二分类标签模型, 但其将先预测出的标签作为后待预测标签的输入特征。CC 虽然考虑到了标签间的相关性, 而且算法较为简单, 但算法结果依赖于标签预测的顺序, 前面预测的标签误差会传递到后面的标签中, 如果前面的标签误差较大, 那么整体算法性能将大打折扣。因此一种集成的链式(ensembles of chain classifiers, ECC)^[17]算法被提出用于解决此缺陷。PW(pairwise binary)^[18]是另一

种对 BR 改进的方法。PW 任意选取 2 个标签进行组合, 这对标签构成一个二类分类问题, 通过标签已知的数据集为每个二类分类问题训练一个分类模型, 这样将有 $|L|(|L|-1)/2$ 个分类模型, 致使 PW 方法计算代价过高。

LP 方法则根据数据集中标签间组合的可能性, 将多标签数据变成 $2^{|L|}$ 单标签数据, 多标签问题变成多类分类问题。LP 方法考虑到了标签间的相关性, 但标签组合的爆炸性问题, 极大增加了算法复杂度。因此, Tsoumakas 等人^[19]提出一种集成的随机标签子集(ensembles of random k -label subsets, RAkEL)方法, 其从标签集中随机选取 k 个标签, 再研究其中可能的标签组合; Read 等人^[20]提出了 EPS(ensembles of pruned sets)算法, 用出现频次较高的标签组合来表示出现频次较低的标签组合, 出现频次较低的被去除。这样既最大限度地保证了标签相关性, 又降低了算法的复杂度。

算法适应方法依赖于某一具体的机器学习算法, 对该具体算法进行改进使其能直接处理多标签数据, 如决策树^[21]、支持向量机^[22]、BP 神经网络^[23]等算法。

Clare 等人^[21]利用多标签熵将传统的 C4.5 决策树算法用于多标签学习, 称作多标签 C4.5, 其允许叶子节点拥有多个标签。Elisseeff 等人^[22]将支持向量机(support vector machine, SVM)算法应用于多标签学习中, 提出了一种 RankSVM 算法。其对每个标签构建一个 SVM 预测模型, 利用排序损失考虑每个样本的相关标签和不相关标签。BPMLL^[23]算法将最流行的神经网络模型之一的 BP 神经网络的扩展成多个输出, 以适应多标签学习, 并提出了一种排序的多标签的误差度量函数, 认为待测样本实际包含的标签应该比不包含的标签排序靠前。Yu 等人^[24]提出了基于粗糙集的多标签学习分类算法和标签局部关系的粗糙集多标签学习分类算法。基于邻域的思想, 文献[5]提出了一种基于邻域粗糙集(neighborhood rough sets)的多标签分类算法, 用于图像语义标注问题。

k NN 算法无需提前训练模型, 优化参数, 相较于其他算法, 具有算法复杂度低且分类效果较好的优势。因此 Spyromitros 等人^[12]对 BR 方法和 k NN 算法结合的 BR k NN 算法进行了扩展, 提出了一种懒惰的多标签分类算法, 用测试样本的 k 个近邻训练样本的标签近似估计它的标签。该方法提升了算法的运行效率, 但其也保留了 BR 方法的不足, 即忽

略了标签间的相关性。Zhang 等人^[9]将 k NN 算法和贝叶斯理论应用于多标签学习中,提出了一种多标签懒惰学习算法(ML- k NN),通过样本的 k 个近邻训练样本的标签信息,用最大后验概率准则预测它的标签。ML- k NN 算法因其算法简单、预测效果好,得到了广泛的关注。但 ML- k NN 算法对每个标签独立预测,未考虑到多个标签间的相关性。因此,文献[25]提出了一种新型多标记懒惰学习算法(IMLLA),该算法同样首先找出测试样本的近邻训练样本,再利用训练数据的分布信息和标签间的相关性来进行预测,考察了样本多个标签之间的相关性。这些算法均没有考虑特征对于标签分类的不同作用。因此本文提出一种基于互信息的粒化特征加权多标签学习 k 近邻算法,以利用标签间的相关性,以及考虑特征对标签分类的重要度,并且不过多增加算法复杂度。

2 基本知识

在介绍所提算法前,先简要介绍互信息和多标签学习的基本概念。

2.1 互信息

定义 1. 熵^[26-27]是度量随机变量不确定性的主要工具,随机变量 X 的熵为

$$H(X) = - \sum_{x \in X} p(x) \text{lb } p(x), \quad (1)$$

其中, $p(x)$ 表示 x 的概率密度。

定义 2. X 和 Y 的联合熵为

$$H(X, Y) = - \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } p(x, y), \quad (2)$$

其中, $p(x, y)$ 是 x 和 y 的联合概率密度。

定义 3. 变量 Y 对于变量 X 条件熵为

$$H(Y | X) = - \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } p(y | x), \quad (3)$$

其中, $p(y | x)$ 是条件概率密度。

性质 1. 条件熵可由熵、联合熵改写为

$$H(Y | X) = H(X, Y) - H(X). \quad (4)$$

证明。

$$\begin{aligned} H(Y | X) &= - \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } p(y | x) = \\ &= - \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } \frac{p(x, y)}{p(x)} = \\ &= \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } p(x) - \\ &\quad \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } p(x, y) = \\ &= H(X, Y) - H(X). \end{aligned}$$

定义 4. 互信息能够度量 2 个变量间的关联程度,指出了一个变量通过另一个变量所获得的知识,对于变量 X 和 Y 的互信息计算如下:

$$I(X; Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } \frac{p(x, y)}{p(x)p(y)}. \quad (5)$$

可以看出 $I(X; Y) = I(Y; X)$ 。当 X 与 Y 相互独立时互信息值为 0。

性质 2. 互信息可以用熵和条件熵表示为

$$I(X; Y) = H(Y) - H(Y | X). \quad (6)$$

证明。

$$\begin{aligned} I(X; Y) &= \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } \frac{p(x, y)}{p(x)p(y)} = \\ &= \sum_{x \in X} \sum_{y \in Y} p(x, y) (\text{lb } \frac{p(x, y)}{p(x)} - \text{lb } p(y)) = \\ &= \sum_{x \in X} \sum_{y \in Y} p(x, y) \text{lb } p(y | x) - \sum_{y \in Y} p(y) \text{lb } p(y) = \\ &= H(Y) - H(Y | X). \end{aligned}$$

证毕。

2.2 多标签学习

$F = \{f_1, f_2, \dots, f_b\}$ 表示多标签数据的 b 维的输入特征空间, $L = \{l_1, l_2, \dots, l_q\}$ 表示给定的标签空间。给定一个多标签数据集 $T = \{(\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2), \dots, (\mathbf{X}_n, \mathbf{Y}_n)\}$, 其中样本 $\mathbf{X}_i = (x_i^1, x_i^2, \dots, x_i^b)$ 表示为一个 b 维特征向量, x_i^j 为样本 \mathbf{X}_i 在特征 f_j 上的取值, $\mathbf{Y}_i = (y_i^1, y_i^2, \dots, y_i^q)$ 则表示与 \mathbf{X}_i 相应的标签向量, 如果 \mathbf{X}_i 含有标签 l_j , 则 $y_i^j = 1$, 否则 $y_i^j = 0$ 。多标签学习的任务则是通过已知的数据集 T , 学习出一个分类函数 $h: F \rightarrow L$, 对于一个标签信息未知的样本 \mathbf{X} 能预测出其对应的标签向量 $\mathbf{Y}' = (y^{1'}, y^{2'}, \dots, y^{q'})$ 。

3 GFWML- k NN

针对以往基于 k NN 的多标签学习算法不考虑特征重要性的差异,忽略标签间的相关性以及避免标签组合爆炸问题,本文提出基于互信息的粒化特征加权多标签学习 k 近邻算法。该算法首先基于粒计算的思想,用平衡 k 均值(balanced k -means)聚类算法^[27]将标签空间粒化成多个标签粒,简化标签的组合;然后,对每个标签粒,根据特征与标签粒的互信息的大小,为特征赋予相应的权重系数,权重系数包含了标签相关性的信息,为待测样本找到更贴切的近邻训练样本;最后,根据近邻训练样本的标签信息可以计算待测样本的标签后验概率值,预测出待测样本相应的标签。

3.1 标签粒化及特征加权

互信息能够有效度量 2 个变量间的关联性,用其度量标签对特征的依赖度. 互信息值越大,说明特征含有标签分类的信息越多,该特征越重要,权重系数越高,因此特征的权重系数与其含有的标签分类信息等价.

定义 5. 特征 f_i 对整个标签空间 L 的重要度,即特征 f_i 的权重系数 ω^i 为

$$\omega^i = I(f_i; L), \quad (7)$$

其中, $I(f_i; L)$ 表示特征 $f_i \in F$ 与标签空间 L 的互信息. $I(f_i; L)$ 由定义 4 可得:

$$\begin{aligned} I(f_i; L) &= \sum_{x \in |f_i|} \sum_{y \in \{0,1\}^q} p(x, y) \text{lb} \frac{p(x, y)}{p(x)p(y)} = \\ &= \sum_{x \in |f_i|} \sum_{y \in \{0,1\}^q} \sum_{y_q \in |l_q|} p(x, y, y_q) \text{lb} \frac{p(x, y, y_q)}{p(x)p(y, y_q)} = \\ &\quad \sum_{x \in |f_i|} \sum_{y_1 \in |l_1|} \cdots \sum_{y_q \in |l_q|} p(x_i, y_1, \dots, y_q) \times \\ &\quad \text{lb} \frac{p(x_i, y_1, y_2, \dots, y_q)}{p(x_i)p(y_1, y_2, \dots, y_q)}, \end{aligned} \quad (8)$$

其中, $|f_i|$ 表示特征 f_i 可能的取值; $|l_j|$ 表示标签 l_j 的可能取值,一般为 0 或者 1. 标签空间 L 中标签的个数为 q ,则标签可能的组合为 $\{0,1\}^q$. 由式(8)可以看出 $I(f_i; L)$ 考虑了标签所有可能的组合,将标签间的相关性融入特征的权重系数中. 但每个标签可能的取值为 0 和 1,整个标签空间的标签组合将会有 2^q 种可能,标签组合数随着标签数的增加而呈指数级的增加,导致算法复杂度急剧上升.

为解决上述问题,基于粒计算思想,本文将标签空间 L 粒化成多个子标签空间,称为标签粒 G_e . 由于 $|G_e| < |L|$,因此标签组合 $2^{|G_e|} \leq 2^{|L|}$. 不同粒中的标签相关性最小,同一粒中的标签相关性极大,保证了粒化对标签相关性的损失最小. 本文依据标签空间中标签的数量,对标签粒的个数 r 取折衷值, $r = \lfloor \sqrt{|q|} \rfloor$,防止了标签粒数量过多造成标签间的相关性损失过大,而标签粒数量过少不能起到减少标签组合数的效果.

目前粒计算中主要的粒化方法有粗糙集理论、模糊集理论、聚类分析等^[13-14]. 本文考虑聚类分析中流行的 k 均值(k -means)聚类算法来粒化标签空间,为了防止粒化时出现标签不平衡问题,即有的标签粒中标签数过多,不能有效减少标签组合的数量,降低算法复杂度,因此采用平衡 k 均值(balanced k -means)聚类算法^[27],将标签均匀地分散到各个标签粒中,具体的标签粒化过程如算法 1 所示.

往往同一个特征对不同标签粒的重要度不同,因此需对每一个标签粒 G_e ,计算其对应的特征权重系数.

定义 6. 对标签粒 G_e ,特征 f_i 的权重系数为 $\omega_e^i = I(f_i; G_e) = H(f_i) + H(G_e) - H(f_i, G_e)$. (9)

由式(9)得到关于标签粒 G_e 的一组特征权重系数 $\omega_e = (\omega_e^1, \omega_e^2, \dots, \omega_e^b)$.

算法 1. 标签粒化过程.

输入: 标签空间 L 、训练集 $T = \{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$ 、标签粒个数 r 、迭代次数 $iter$;

输出: r 个标签粒.

步骤 1. 对每个标签粒 G_i 和粒中心 g_i 初始化,

将 G_i 赋值为 ϕ ,从 L 中随机选择标签到 g_i .

步骤 2. 将标签粒化到各个粒中:

WHILE $iter > 0$

FOR $l_j \in L$

步骤 2.1. 计算 l_j 到每个粒中心 g_i 在数据集 T 中的距离 d_{ij} ,令 φ 表示 l_j ,将循环信号 $flag$ 设为真;

步骤 2.2. 对 l_j 粒化:

WHILE $flag$

① 找到离 φ 最近的粒中心 g_k ,将 l_j 插入 G_k 中,根据距离大小对 G_k 中的标签进行排序;

② IF $|G_k| > \lceil |L| / r \rceil$ THEN

③ 将 G_k 中排序最后一个标签赋值给 φ ,并把该标签从 G_k 去除,把 d_{kp} 设为 ∞ ;

④ ELSE 将循环信号 $flag$ 设为假;

⑤ END IF

END WHILE

END FOR

重新计算各粒中心,迭代次数 $iter$ 减 1.

END WHILE

步骤 3. 得到标签粒 G_1, G_2, \dots, G_r .

3.2 近邻选择

以往的算法在寻找近邻、度量样本间的相似度和计算样本间距离时,对所有特征给予相同的权重,不考虑特征间重要度的差异,这里以欧氏距离为例,欧氏距离是 k NN 算法中常用的一种样本距离度量,样本 X_i 与 X_j 的距离计算如下:

$$d(X_i, X_j) = \sqrt{\sum_{k=1}^b (x_i^k - x_j^k)^2}. \quad (10)$$

而往往不同特征含有的标签分类信息不同,需将特征重要度的差异体现在距离中,计算出特征加权的距离.

定义 7. 对于标签粒 G_e , 基于特征的权重信息 ω_e 得到样本间特征加权的欧氏距离:

$$D_e(\mathbf{X}_i, \mathbf{X}_j) = \sqrt{\sum_{k=1}^b (\omega_e^k)^2 (x_i^k - x_j^k)^2}. \quad (11)$$

从式(11)可以看到当所有特征的权重系数 $\omega_e^k = 1$ 时, $D_e(\mathbf{X}_i, \mathbf{X}_j) = d(\mathbf{X}_i, \mathbf{X}_j)$.

性质 3. 特征加权的欧氏距离与一般的欧氏距离有关系:

$$D_e(\mathbf{X}_i, \mathbf{X}_j) = d(\omega_e \mathbf{X}_i, \omega_e \mathbf{X}_j). \quad (12)$$

证明.

$$\begin{aligned} D_e(\mathbf{X}_i, \mathbf{X}_j) &= \\ &\sqrt{\sum_{k=1}^b (\omega_e^k)^2 (x_i^k - x_j^k)^2} = \\ &\sqrt{\sum_{k=1}^b (\omega_e^k x_i^k - \omega_e^k x_j^k)^2} = \\ &d(\omega_e \mathbf{X}_i, \omega_e \mathbf{X}_j). \end{aligned}$$

证毕.

其中, $\omega_e \mathbf{X}_i = (\omega_e^1 x_i^1, \omega_e^2 x_i^2, \dots, \omega_e^b x_i^b)$. 因此可以将特征的权重系数带入特征的差异度量中, 那么加权的欧氏距离可以通过对样本的特征赋予相应的权重, 再利用一般的欧氏距离进行运算.

在标签粒 G_e 上, 样本 \mathbf{X} 得到一个与标签已知的训练样本的加权欧氏距离集 $D_e = \{D_e(\mathbf{X}, \mathbf{X}_1), D_e(\mathbf{X}, \mathbf{X}_2), \dots, D_e(\mathbf{X}, \mathbf{X}_n)\}$. 对距离集 D_e 中的距离值升序排列后, 第 k 个距离值设为阈值 t , 获得样本 \mathbf{X} 在标签粒 G_e 上训练集中的 k 个最近邻训练样本 $N_e(\mathbf{X}) = \{\mathbf{X}_i | D_e(\mathbf{X}, \mathbf{X}_i) \leq t\}$.

3.3 标签预测

根据 k 个近邻 $N_e(\mathbf{X})$, 基于经典的 ML- k NN 算法^[9], 预测标签未知的测试样本 \mathbf{X} 含有标签粒 G_e 中标签的概率值, 将得到的各标签粒的结果最后组合得到测试样本 \mathbf{X} 的标签概率向量 $\mathbf{P} = (p_1, p_2, \dots, p_q)$, 由标签概率向量可以得到预测的标签向量 $\mathbf{Y}' = (y^{1'}, y^{2'}, \dots, y^{q'})$.

算法 2. 粒化特征加权的多标签 k 近邻算法.

输入: 测试样本 \mathbf{X} 、近邻数 k 、训练集 $T = \{(\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2), \dots, (\mathbf{X}_n, \mathbf{Y}_n)\}$ 、标签粒个数 r 、迭代次数 $iter$ 、标签空间 L ;

输出: \mathbf{X} 的标签概率向量 \mathbf{P} 、标签向量 \mathbf{Y}' .

步骤 1. 根据算法 1 对标签空间 L 进行粒化, 得到标签粒 $G_e (1 \leq e \leq r)$.

步骤 2. 对每个标签粒进行如下处理:

FOR $e=1$ to r

步骤 2.1. 根据式(9)得到关于标签粒 G_e 的特征系数 ω_e ;

步骤 2.2. 对训练样本和测试样本的特征进行加权得到 $\omega_e \mathbf{X}_i (1 \leq i \leq n)$ 和 $\omega_e \mathbf{X}$;

步骤 2.3. 根据式(10)计算测试样本 \mathbf{X} 与所有训练样本 \mathbf{X}_i 的加权距离, 找到 \mathbf{X} 的 k 个近邻训练样本 $N_e(\mathbf{X})$;

FOR $i \in G_e$

步骤 2.4. 根据经典 ML- k NN 算法^[9], 统计出 $N_e(\mathbf{X})$ 中拥有标签 l_j 的样本个数 $C_X(l_j)$, 得到 \mathbf{X} 含有标签 l_j 的概率预测值:

$$p_j = \frac{p(H_1^j) p(E_{C_X(l_j)}^j | H_1^j)}{\sum_{a=0}^1 p(H_a^j) p(E_{C_X(l_j)}^j | H_a^j)}.$$

END FOR

END FOR

步骤 3. 将所有标签的概率预测值组合后得到 \mathbf{X} 所有的标签概率:

$$\mathbf{P} = (p_1, p_2, \dots, p_q).$$

步骤 4. 由标签概率得到 \mathbf{X} 的预测标签向量

$\mathbf{Y}' = (y^{1'}, y^{2'}, \dots, y^{q'})$, 如果 $p_j \geq 0.5$, 则 $y^{j'} = 1$, 否则 $y^{j'} = 0$.

上述算法步骤 2.4 中的 $p(H_a^j)$ 表示训练集中含有标签 l_j 的 ($a=1$) 或者不含有的 ($a=0$) 样本的比率, 即先验概率 $p(E_{C_X(l_j)}^j | H_a^j)$ 表示训练集中样本含有 l_j ($a=1$) 或不含有 l_j ($a=0$) 时, 其近邻中有 $C_X(l_j)$ 个样本含有 l_j 的条件概率, 通过该两者可以算出测试样本含有 l_j 的后验概率, 具体计算方式请见文献[9].

4 数据实验

4.1 数据集及实验设置

为了验证算法的有效性, 本文选取了来自 Mulan Library^[29] 的涵盖多个领域的 5 个真实世界的多标签数据集进行实验, 多标签数据集对应的名称、领域、样本数量、特征维度、标签空间中标签数量、标签基数等详细信息如表 1 所示:

Table 1 Multi-Label Datasets Used in the Experiments

表 1 多标签数据集

Name	Domain	# Instance	Dimension of Feature	# Label	Label Cardinality
Emotions	Music	593	72	6	1.869
Medical	Text	978	1449	45	1.245
Yeast	Biology	2417	103	14	4.239
CAL500	Music	502	68	174	26.044
Genbase	Biology	662	1185	27	1.252

1) Emotions 数据集^[30]包含了 593 个标注了情感的歌曲样本, 每个样本由 72 个特征来描述, 即 8 韵律特征和 64 音色特征和 6 个可能的情感标签表示, 每个标签代表了一个基于 Tellegen-Watson-Clark 模型的歌曲情感聚类。

2) Medical 数据集^[31]包含了 978 个病历样本, 其含有 1449 个特征, 每个样本的特征由诊断历史记录和观察到的症状组成, 标签则是 45 种 ICD-9-CM 疾病编码。

3) Yeast 数据集^[22]用于描述酵母菌的基因功能分类, 其包含了 2417 个样本, 每个样本表示一个 yeast 基因, 每个基因对应于一个 103 维的特征向量, 标签空间是 14 种可能的基因功能。

4) CAL500 数据集^[32]含有 502 首流行乐曲, 以及 174 个风格、情绪、乐器等语义关键词, 每个样本由 68 个特征表示。

5) Genbase 数据集^[33]是一个关于蛋白质功能分类的多标签数据集, 由 662 个蛋白质样本组成, 每个蛋白质由 1185 个蛋白基序表示。标签为 27 个蛋白家族功能类别, 如抗氧化酶、结构蛋白、受体等。

Medical 和 Genbase 数据集的特征值为离散型, 而其他数据集的特征值为连续型。对于离散型的特征, 可以很容易计算其与标签的互信息, 而连续型的特征则比较困难。因此, 我们对连续型特征采用二值等距区间离散化方法。

除了经典的多标签懒惰学习算法 ML-*k*NN^[9], 还将本文所提算法与其他常见的多标签学习算法 RankSVM^[20], BPMML^[21], BR*k*NN^[12]进行了对比。所有的实验在 Matlab2012b 上完成。为了取得最佳效果, 根据相应文献的建议选取最优的参数配置。ML-*k*NN 算法中, 近邻数为 10、平滑因子为 1; RankSVM 选用了度为 8 的多项式核函数; BPMML 的隐藏层节点数为特征数的 20%; BR*k*NN 的近邻数为 10。根据表 1 统计的多标签数据集的标签数, 实验中本文算法 GFWML-*k*NN 在数据集 Emotions, Medical, Yeast, CAL500, Genbase 的标签粒数分别为 2, 6, 3, 15, 5。

4.2 评价指标

传统机器学习中有许多评价指标, 但这些指标不能直接用于多标签学习。这里选用了 5 个常用的多标签学习算法评价指标, 分别为汉明损失、1-错误率、覆盖率、排序损失和平均精度^[4]。给定一个多标签测试集 $D = \{(\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2), \dots, (\mathbf{X}_m, \mathbf{Y}_m)\}$, 用 $Y_i = \{l_j \mid y'_j = 1\}$ 表示与 \mathbf{X}_i 相关的标签集, \bar{Y}_i 表

示 Y_i 的补集, \mathbf{Y}'_i 表示测试样本 \mathbf{X}_i 的标签预测信息, 这 5 个评价指标定义如下:

1) 汉明损失(Hamming loss, Hamloss). 度量算法预测出的标签信息与实际的标签信息的平均差异值:

$$\text{Hamloss} = \frac{1}{mq} \sum_{i=1}^m \sum_{j=1}^q [\![y'_i \neq y_i]\!].$$

当 $y'_i \neq y_i$ 时, $[\![y'_i \neq y_i]\!]$ 值为 1, 否则为 0。

2) 1-错误率(one error, Onerror). 计算算法预测的排序最靠前的标签实际不是测试样本的标签的比率:

$$\text{Onerror} = \frac{1}{m} \sum_{i=1}^m [\!(\arg \min_{l \in L} r_i(l)) \notin Y_i]\!],$$

$(\arg \min_{l \in L} r_i(l)) \notin Y_i$ 条件满足时, $[\!(\arg \min_{l \in L} r_i(l)) \notin Y_i]\!$ 值为 1, 否则为 0。

3) 覆盖率(Coverage). 计算要囊括测试样本实际包含的所有标签所需最大排序距离:

$$\text{Coverage} = \frac{1}{m} \sum_{i=1}^m \max_{l \in Y_i} r_i(l) - 1.$$

4) 排序损失(ranking loss, Rankloss). 评价有多少测试样本实际不包含的标签比实际包含的标签排序高:

$$\text{Rankloss} = \frac{1}{m} \sum_{i=1}^m \frac{1}{|\bar{Y}_i|} \times$$

$$\left| \{(l_a, l_b); r_i(l_a) > r_i(l_b), (l_a, l_b) \in Y_i \times \bar{Y}_i\} \right|.$$

5) 平均准确率(average precision, Avgprec). 用于评价给定一个测试样本实际包含的标签, 平均有多少实际包含的标签排序比其高:

$$\text{Avgprec} = \frac{1}{m} \sum_{i=1}^m \frac{1}{|\bar{Y}_i|} \sum_{l \in Y_i} \frac{|\{l' \in \bar{Y}_i : r_i(l') \leqslant r_i(l)\}|}{r_i(l)}.$$

前面 4 项评价指标的值越低说明算法效果越好, 而平均准确率值越高则说明算法效果越好。

4.3 实验结果

4.3.1 近邻参数选择

首先讨论了近邻数 k 的选择以及验证标签空间的粒化没有对标签相关性造成过大的损失。以 Emotions 数据集为例, 图 1~5 给出了 Emotions 数据集的 5 项评价指标随着近邻数 k 增加的变化曲线。其中, k 以步长 2 从 2 增加到 20。图 1~5 中, ML-*k*NN 曲线表示经典的多标签懒惰学习算法; FWML-*k*NN 曲线表示未粒化的特征加权 ML-*k*NN 算法; GFWML-*k*NN 曲线表示粒化的特征加权 ML-*k*NN 算法。

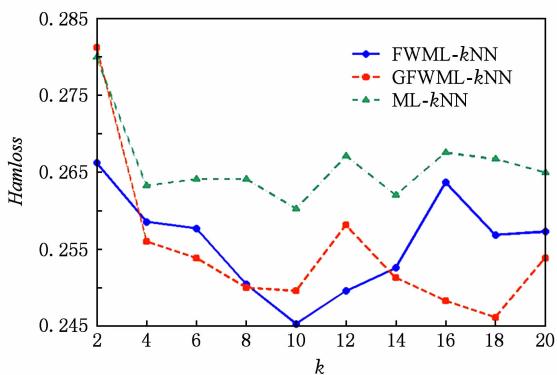


Fig. 1 Hamming loss of varying the number of nearest neighbors

图1 汉明损失随着近邻数增加的变化曲线

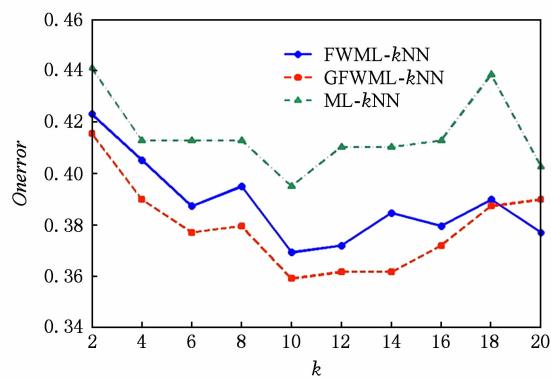


Fig. 2 One error of varying the number of nearest neighbors

图2 1-错误率随着近邻数增加的变化曲线

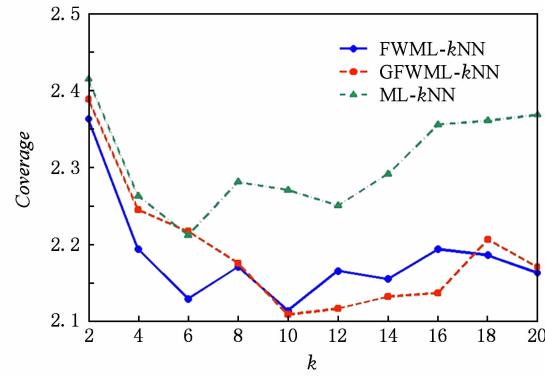


Fig. 3 Coverage of varying the number of nearest neighbors

图3 覆盖率随着近邻数增加的变化曲线

在各项评价指标上,粒化和未粒化的特征加权 $ML-kNN$ 以及经典的 $ML-kNN$ 的性能均随着近邻数 k 的增加而快速提升,而后达到最优值后逐渐略微下降。其中,粒化和未粒化的特征加权 $ML-kNN$ 的性能变化趋势十分接近,且在各个近邻数上基本

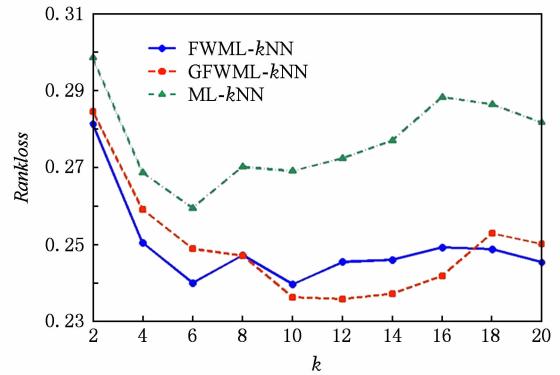


Fig. 4 Ranking loss of varying the number of nearest neighbors

图4 排序损失随着近邻数增加的变化曲线

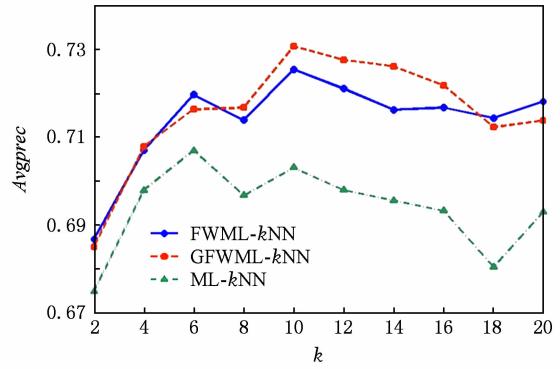


Fig. 5 Average precision of varying the number of nearest neighbors

图5 平均准确率随着近邻数增加的变化曲线

都优于经典 $ML-kNN$ 。当近邻数 $k=10$, 性能最优, 因此, 本文近邻数设为 10。

$GFWML-kNN$ 算法取得的最优值除了在汉明损失 $Hamloss$ 上比未粒化的特征加权多标签学习懒惰算法略大一点, 在 1-错误率 $Onerror$ 、排序损失 $Rankloss$ 和平均精度 $Avgprec$ 上均要优于未粒化的算法, 两者取得相同的覆盖率 $Coverage$ 最优值。综上, $GFWML-kNN$ 算法的性能不但不差于未粒化的特征加权 $ML-kNN$ 算法, 反而略优, 说明 $GFWML-kNN$ 算法的粒化几乎保留了标签间的相关性, 找到更合适的近邻, 提高了算法的性能。

4.3.2 实验对比

实验采用了十折交叉验证 (ten-fold cross-validation) 方法, 实验结果用均值士标准差表示。表 2~5 表示了各个多标签学习算法在多标签数据集 Emotions, Medical, Yeast, CAL500, Genbase 上取得的实验结果, 其中各项评价指标的最优值用粗体标注, ↓(↑) 表示该项评价指标值越小(越大) 算法效果越好。

Table 2 Experimental Results Obtained by Multi-label Algorithms (Mean±Std. deviation) on the Emotions Dataset

表 2 Emotions 数据集的实验结果(均值士标准差)

Evaluation Criteria	GFWML-kNN	ML-kNN	RankSVM	BPMML	BRkNN
Hamming Loss ↓	0.250±0.075	0.260±0.103	0.378±0.113	0.318±0.114	0.268±0.099
One Error ↓	0.359±0.251	0.395±0.218	0.623±0.342	0.572±0.235	0.408±0.176
Coverage ↓	2.110±0.933	2.272±1.120	2.982±1.047	3.067±1.133	1.892±0.996
Ranking Loss ↓	0.236±0.123	0.269±0.129	0.429±0.234	0.429±0.180	0.314±0.110
Average Precision ↑	0.731±0.112	0.703±0.115	0.568±0.195	0.560±0.084	0.677±0.097

↓ : The smaller the value is, the better the performance is. ↑ : The larger the value is, the better the performance is.

从表 2 可以看出, GFWML-kNN 在绝大多数评价指标上取得了较好的性能. 除了在覆盖率 Coverage 上仅次于 BRkNN, 但 BRkNN 在其他 4 项评价指标上算法性能明显劣于 GFWML-kNN, 因此总体来说 GFWML-kNN 优于 BRkNN. 而对于其他 3 种对比算法 ML-kNN, RankSVM, BPMML, GFWML-kNN 算法在所有评价指标上均取得了更好的性能, 尤其相比于经典的多标签懒惰学习算法 ML-kNN. 所以在 Emotions 数据集上, GFWML-kNN 算法的性能明显优于其他对比算法的性能.

对于 Medical 数据集, 由表 3 可知, 在评价指标

汉明损失 Hamloss、1-错误率 Onerror 和平均精度 Avgprec 上, GFWML-kNN 取得最优值, 排序损失 Rankloss 的最优值和覆盖率 Coverage 的最优值则分别由 BPMML 和 BRkNN 获得, RankSVM 在所有评价指标上取得最差值. GFWML-kNN 算法在评价指标排序损失 Rankloss 上, 性能略微差于 BPMML, 而明显优于剩下的 3 种对比算法. 虽然 BRkNN 在覆盖率 Coverage 表现最优, 但其在其他 4 种评价指标上较差, 特别是明显差于 GFWML-kNN. 此外, 相较于 ML-kNN 算法, 本文所提算法 GFWML-kNN 在各项评价指标上的性能均显著优于该算法.

Table 3 Experimental Results Obtained by Multi-label Algorithms (Mean±Std. deviation) on the Medical Dataset

表 3 Medical 数据集的实验结果(均值士标准差)

Evaluation Criteria	GFWML-kNN	ML-kNN	RankSVM	BPMML	BRkNN
Hamming Loss ↓	0.011±0.004	0.015±0.007	0.079±0.152	0.018±0.009	0.017±0.008
One Error ↓	0.166±0.113	0.242±0.180	0.952±0.403	0.324±0.188	0.295±0.181
Coverage ↓	2.265±1.892	2.685±2.279	22.638±17.129	1.700±1.962	1.032±0.469
Ranking Loss ↓	0.035±0.031	0.042±0.036	0.555±0.447	0.030±0.030	0.104±0.081
Average Precision ↑	0.867±0.071	0.811±0.112	0.1002±0.412	0.786±0.124	0.745±0.126

↓ : The smaller the value is, the better the performance is. ↑ : The larger the value is, the better the performance is.

如表 4 所示, 本文所提算法 GFWML-kNN 在汉明损失 Hamloss、排序损失 Rankloss 和平均精度 Avgprec 上表现最优, 而在 1-错误率 Onerror 上略

次于 RankSVM 和 BPMML, 但在覆盖率 Coverage 上优于这 2 种算法. 相较于 GFWML-kNN, BRkNN 算法依旧在覆盖率 Coverage 上表现较好, 在剩下的评

Table 4 Experimental Results Obtained by Multi-label Algorithms (Mean±Std. deviation) on the Yeast Dataset

表 4 Yeast 数据集的实验结果(均值士标准差)

Evaluation Criteria	GFWML-kNN	ML-kNN	RankSVM	BPMML	BRkNN
Hamming Loss ↓	0.196±0.027	0.197±0.031	0.200±0.028	0.210±0.034	0.205±0.032
One Error ↓	0.238±0.085	0.240±0.104	0.233±0.130	0.234±0.118	0.244±0.130
Coverage ↓	6.387±0.728	6.384±0.677	6.95±0.998	6.417±0.774	5.281±0.608
Ranking Loss ↓	0.172±0.043	0.173±0.036	0.187±0.054	0.174±0.042	0.215±0.049
Average Precision ↑	0.758±0.063	0.756±0.059	0.756±0.071	0.750±0.068	0.739±0.058

↓ : The smaller the value is, the better the performance is. ↑ : The larger the value is, the better the performance is.

价指标表现较差. 所以在 Yeast 数据集, GFWML- k NN 算法整体上略好于其他对比算法.

在 CAL500 数据集上, 虽然 GFWML- k NN 的算法效果没有在对比的算法中取得最优值, 但从表 5 可见, GFWML- k NN, ML- k NN, RankSVM, BPMILL

在所有指标上几乎没有差异, 而 BR k NN 虽在覆盖率 Coverage 上表现最优, 但其也在剩下 4 项评价指标上表现最差, 所以总体而言在 CAL500 数据集中前 4 种算法表现无异, BR k NN 表现较差. 特别地本文所提算法 GFWML- k NN 略优于 ML- k NN.

Table 5 Experimental Results Obtained by Multi-label Algorithms (Mean±Std. deviation) on the CAL500 Dataset

表 5 CAL500 数据集的实验结果(均值士标准差)

Evaluation Criteria	GFWML- k NN	ML- k NN	RankSVM	BPMILL	BR k NN
Hamming Loss ↓	0.139±0.019	0.140±0.020	0.138±0.021	0.143±0.048	0.149±0.020
One Error ↓	0.118±0.087	0.120±0.069	0.116±0.079	0.116±0.079	0.206±0.186
Coverage ↓	130.320±9.576	130.092±9.449	129.700±9.031	129.604±8.639	95.388±3.758
Ranking Loss ↓	0.183±0.018	0.183±0.018	0.183±0.018	0.181±0.017	0.287±0.057
Average Precision ↑	0.493±0.037	0.492±0.039	0.498±0.039	0.497±0.037	0.433±0.064

↓ : The smaller the value is, the better the performance is. ↑ : The larger the value is, the better the performance is.

Genbase 数据集中平均每个样本含有的标签数极少, 所以除了 RankSVM 算法外, 其他算法都在 Genbase 数据集上取得了很好的结果. 从表 6 可知, BPMILL 在 1-错误率 Onerror 上取得最佳值, 其预测的每个相关性最大的标签均为测试样本实际含有的标签. GFWML- k NN 取得的排序损失 Rankloss 和平均精度 Avgprec 优于其它算法, 1-错误率

Onerror 上和覆盖率 Coverage 上分别仅次于 BPMILL 和 BR k NN. 各个算法取得的汉明损失 Hamloss 差异性较小. 虽然在该数据集上 ML- k NN 表现已经很优异, 但 GFWML- k NN 除了汉明损失 Hamloss 外在各项指标上仍显著优于 ML- k NN. 因此, 在 Genbase 数据集, GFWML- k NN 整体上好于其他算法.

Table 6 Experimental Results Obtained by Multi-label Algorithms (Mean±Std. deviation) on the Genbase Dataset

表 6 Genbase 数据集的实验结果(均值士标准差)

Evaluation Criteria	GFWML- k NN	ML- k NN	RankSVM	BPMILL	BR k NN
Hamming Loss ↓	0.0048±0.0100	0.0045±0.0086	0.0483±0.0612	0.0044±0.083	0.0043±0.0093
One Error ↓	0.003±0.019	0.015±0.048	0.559±0.526	0.000±0.000	0.018±0.056
Coverage ↓	0.614±1.540	0.703±1.705	6.977±9.789	0.749±1.714	0.211±0.373
Ranking Loss ↓	0.008±0.027	0.010±0.032	0.255±0.378	0.010±0.030	0.012±0.042
Average Precision ↑	0.991±0.026	0.983±0.039	0.542±0.489	0.986±0.039	0.980±0.049

↓ : The smaller the value is, the better the performance is. ↑ : The larger the value is, the better the performance is.

从整个数据集来看, 所有算法中 GFWML- k NN 算法的整体算法效果表现突出. 对比于经典的多标签懒惰学习算法 ML- k NN, GFWML- k NN 在所有评价指标上都好于该算法, 尤其是在特征数远多于标签数的 Emotions, Medical, Genbase 数据集上显著优于该算法, 因为特征数越多, 越不相关的特征越多, 特征间重要度的差异越大. 对于另一种基于 k NN 的多标签学习算法 BR k NN, 本文所提算法除了评价指标覆盖率外, 均明显优于该算法. 这些结果与分析验证了在寻找 k 近邻时特征加权的有效性.

5 结 论

以往基于 k NN 的多标签学习算法, 在寻找近邻时, 往往把所有特征同等对待, 而且大多将多标签学习问题分解为多个独立的二类分类问题, 对每个标签单独求解, 忽略了标签间的相关性. 因此本文提出了一种基于互信息的粒化特征加权多标签学习 k 近邻算法. 该算法考虑了特征对标签分类的重要性, 为特征赋予相应的权重, 并将标签间的相关性融入到

特征的权重系数中,这样更利于寻找更相关的近邻样本,提升标签预测的准确性。而基于粒计算解决复杂问题的思想,粒化标签空间以解决标签组合爆炸性问题,而极少损失标签间的相关性。实验结果表明本文所提算法 GFWML-kNN 算法整体上能取得较好的预测效果,而且优于以往将所有特征同等对待的算法,验证了考虑特征重要度和标签间的相关性是必要的。但该算法对于含有大量标签的数据集处理速度较慢,寻找一个更合理的粒化方式是未来的研究工作。

参 考 文 献

- [1] Tsoumakas G, Katakis I, Vlahavas I. Mining multi-label data [G] //Data Mining and Knowledge Discovery Handbook. Berlin: Springer, 2010: 667–685
- [2] Tsoumakas G, Katakis I. Multi label classification: An overview [J]. International Journal of Data Warehousing and Mining, 2007, 3(3): 1–13
- [3] Zhang Minling, Zhou Zhihua. A review on multi-label learning algorithms [J]. IEEE Trans on Knowledge and Data Engineering, 2014, 26(8): 1819–1837
- [4] Schapire R, Singer Y. BoosTexter: A boosting-based system for text categorization [J]. Machine Learning, 2000, 39(2/3): 135–168
- [5] Yu Ying, Pedrycz W, Miao Duoqian. Neighborhood rough sets based multi-label classification for automatic image annotation [J]. International Journal of Approximate Reasoning, 2013, 54(9): 1373–1387
- [6] Pavlidis P, Weston J, Cai J, et al. Combining microarray expression data and phylogenetic profiles to learn functional categories using support vector machines [C] //Proc of the 5th Annual Int Conf on Computational Biology. New York: ACM, 2001: 242–248
- [7] Zhang Gang, Zhong Ling, Huang Yonghui. A machine learning method for histopathological image automatic annotation [J]. Journal of Computer Research and Development, 2015, 52(9): 2135–2144 (in Chinese)
(张钢, 钟灵, 黄永慧. 一种病理图像自动标注的机器学习方法 [J]. 计算机研究与发展, 2015, 52(9): 2135–2144)
- [8] Altman N. An introduction to kernel and nearest-neighbor nonparametric regression [J]. The American Statistician, 1992, 46 (3): 175–185
- [9] Zhang Minling, Zhou Zhihua. ML-KNN: A lazy learning approach to multi-label learning [J]. Pattern Recognition, 2007, 40(7): 2038–2048
- [10] Wieczorkowska A, Synak P, Raś Z. Multi-label classification of emotions in music [C] //Proc of the 2006 Int Conf on Intelligent Information Processing and Web Mining. Berlin: Springer, 2006: 307–315
- [11] Brinker K, Hüllermeier E. Case-based multilabel ranking [C] //Proc of the 20th Int Joint Conf on Artificial Intelligence. San Francisco: Morgan Kaufmann, 2007: 702–707
- [12] Spyromitros E, Tsoumakas G, Vlahavas I. An empirical study of lazy multilabel classification algorithms [G] //LNCS 5138: Proc of the 5th Hellenic Conf on Artificial Intelligence. Berlin: Springer, 2008: 401–406
- [13] Yao Yiyu. Interpreting concept learning in cognitive informatics and granular computing [J]. IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics, 2009, 39(4): 855–866
- [14] Yao Yiyu, Zhao Liqian. A measurement theory view on the granularity of partitions [J]. Information Sciences, 2012, 213(23): 1–13
- [15] Boutell M, Luo J, Shen X, et al. Learning multi-label scene classification [J]. Pattern Recognition, 2004, 37(9): 1757–1771
- [16] Read J, Pfahringer B, Holmes G, et al. Classifier chains for multi-label classification [G] //LNAI 5782: Proc of the 20th European Conf on Machine Learning. Berlin: Springer, 2009: 254–269
- [17] Read J, Pfahringer B, Holmes G, et al. Classifier chains for multi-label classification [J]. Machine Learning, 2011, 85 (3): 335–359
- [18] Hüllermeier E, Fürnkranz J, Cheng W, et al. Label ranking by learning pairwise preferences [J]. Artificial Intelligence, 2008, 172(16): 1897–1916
- [19] 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
- [20] Read J, Pfahringer B, Holmes G. Multi-label classification using ensembles of pruned sets [C] //Proc of the 8th IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2008: 995–1000
- [21] Clare A, King R. Knowledge discovery in multi-label phenotype data [G] //LNCS 2168: Proc of the 5th European Conf on Principles of Data Mining and Knowledge Discovery. Berlin: Springer, 2001: 42–53
- [22] Elisseeff A, Weston J. A kernel method for multi-labelled classification [C] //Proc of Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press, 2001: 681–687
- [23] Zhang Minling, Zhou Zhihua. Multilabel neural networks with applications to functional genomics and text categorization [J]. IEEE Trans on Knowledge and Data Engineering, 2006, 18(10): 1338–1351
- [24] Yu Ying, Pedrycz W, Miao Duoqian. Multi-label classification by exploiting label correlations [J]. Expert Systems with Applications, 2014, 41(6): 2989–3004
- [25] Zhang Minling. An improved multi-label lazy learning approach [J]. Journal of Computer Research and Development, 2012, 49(11): 2271–2282 (in Chinese)
(张敏灵. 一种新型多标记懒惰学习算法 [J]. 计算机研究与发展, 2012, 49(11): 2271–2282)

- [26] Shannon C. A mathematical theory of communication [J]. Bell System Technical Journal, 1948, 27(3): 379–423
- [27] Shannon C. A mathematical theory of communication [J]. Bell System Technical Journal, 1948, 27(4): 623–666
- [28] Tsoumakas G, Katakis I, Vlahavas I. Effective and efficient multi-label classification in domains with large number of labels [C] //Proc of ECML/PKDD 2008 Workshop on Mining Multidimensional Data. Berlin: Springer, 2008: 287–313
- [29] Tsoumakas G, Spyromitros-Xioufis E, Vilcek I. Mulan: a java library for multi-label learning [J]. Journal of Machine Learning Research, 2011, 12(7): 2411–2414
- [30] Trohidis K, Tsoumakas G, Kalliris G, et al. Multi-label classification of music into emotions [J]. Eurasip Journal on Audio Speech & Music Processing, 2008, 2011(1): 325–330
- [31] Pestian J, Brew C, Matykiewicz P, et al. A shared task involving multi-label classification of clinical free text [C] // Proc of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing. Stroudsburg, PA: Association for Computational Linguistics, 2007: 97–104
- [32] Turnbull D, Barrington L, Torres D, et al. Semantic annotation and retrieval of music and sound effects [J]. IEEE Trans on Audio, Speech, and Language Processing, 2008, 16(2): 467–476
- [33] Diplaris S, Tsoumakas G, Mitkas P, et al. Protein classification with multiple algorithms [G] //LNCS 3746: Proc of the 10th Panhellenic Conf Informatics (PCI 2005). Berlin: Springer, 2005: 448–456



Li Feng, born in 1989, PhD candidate in Tongji University. Student member of CCF. His main research interests include multi-label learning and granular computing.



Miao Duoqian, born in 1964. Professor and PhD supervisor in Tongji University. Distinguished member of CCF. His main research interests include rough sets, granular computing, and machine learning.



Zhang Zhifei, born in 1986. PhD and lecturer in Tongji University. Member of CCF. His main research interests include natural language processing and machine learning (zhifeizhang@tongji.edu.cn).



Zhang Wei, born in 1978. PhD candidate in Tongji University. Student member of CCF. Her main research interests include rough sets and machine learning (zhangweismile@163.com).

《信息安全研究》期刊简介

习近平总书记指出“没有网络安全就没有国家安全,没有信息化就没有现代化”. 数字时代信息安全工具的大众化是不可阻挡的历史潮流. 大众化的信息安全已经直接影响到我们每个人的利益,信息安全已成为国家、地方区域经济结构优化提升和转型发展的新机遇. 在信息安全上升为国家战略、行业迎来崭新发展机遇形势下,《信息安全研究》期刊应时代而生.

《信息安全研究》是由国家发改委主管、国家信息中心主办的中文学术期刊,其宗旨是集中展示和报道国际、国内网络和信息安全研究领域研究成果及最新应用,传播信息安全基础理论和技术策略,服务国家信息安全形势发展需要. 所刊登的论文均经过专家严格评审.

《信息安全研究》于2015年10月创刊发行. 刊期为月刊,每期96页,由《信息安全研究》杂志社出版,国内外公开发行.

《信息安全研究》将以研究致以应用,搭建信息安全领域的学术交流平台,愿意和同行业及社会各界建立联系,友好合作,共赢美好未来. 欢迎大家积极投稿、赐稿,洽谈合作.

投稿信箱:ris@cei.gov.cn

编辑部联系人:崔先生(185 0008 6481)

马先生(158 1058 2450)