

基于标签相关性的 K 近邻多标签分类方法

檀何凤*, 刘政怡

(安徽大学 计算机科学与技术学院, 合肥 230601)

(* 通信作者电子邮箱 1060895242@qq.com)

摘要:针对 K 近邻多标签 (ML-KNN) 分类算法中未考虑标签相关性的问题, 提出了一种基于标签相关性的 K 近邻多标签分类 (CML-KNN) 算法。首先, 计算出标签集合中每对标签间的条件概率; 其次, 对于即将被预测的标签, 将其与已经预测的标签间的条件概率进行排序, 求出最大值; 最后, 将最大值跟对应标签值相乘同时结合最大化后验概率 (MAP) 来构造多标签分类模型, 对新标签进行预测。实验结果表明, 所提算法在 Emotions 数据集上的分类性能均优于 ML-KNN、AdaboostMH、RAkEL、BPMLL 这 4 种算法; 在 Yeast、Enron 数据集上仅在 1~2 个评价指标上低于 ML-KNN 与 RAkEL 算法。由实验分析可知, 该算法取得了较好的分类效果。

关键词: 标签相关性; K 近邻多标签; 条件概率; 多标签分类

中图分类号: TP391.4 **文献标志码:** A

Multi-label K nearest neighbor algorithm by exploiting label correlation

TAN Hefeng*, LIU Zhengyi

(College of Computer Science and Technology, Anhui University, Hefei Anhui 230601, China)

Abstract: Since the Multi-Label K Nearest Neighbor (ML-KNN) classification algorithm ignores the correlation between labels, a multi-label classification algorithm by exploiting label correlation named CML-KNN was proposed. Firstly, the conditional probability between each pair of labels was calculated. Secondly, the conditional probabilities of predicted labels and the conditional probability of the label to be predicted were ranked, then the maximum was got. Finally, a new classification model by combining Maximum A Posteriori (MAP) and the product of the maximum and its corresponding label value was proposed and the new label value was predicted. The experimental results show that the performance of CML-KNN on Emotions dataset outperforms the other four algorithms, namely ML-KNN, AdaboostMH, RAkEL, BPMLL, while only two evaluation metric values are lower than those of ML-KNN and RAkEL on Yeast and Enron datasets. The experimental analyses show that CML-KNN obtains better classification results.

Key words: label correlation; Multi-Label K Nearest Neighbor (ML-KNN); conditional probability; multi-label classification

0 引言

随着信息技术的发展与社会的进步, 多标签分类^[1]已经成为分类问题的重要组成部分。在实际生活中多标签分类的应用也变得越来越广泛。然而多标签分类并不同于传统的单标签分类问题, 多标签数据的多义性, 使得传统的分类算法已不再适应于这些多标签数据集。正因为如此, 研究者们先后提出了大量用于处理多标签数据的分类方法。通过总结这些方法可以分为以下三类: 问题转换法、算法转换法和集成方法。问题转换法是通过某种方法对数据集进行转换, 使其从多标签数据集转化成单标签数据集, 再利用传统的分类方法进行, 如: CO (Copy transformation)^[1] 方法通过直接复制样本将多标签的问题转换成多类问题, 简单易行, 但是完全把多标签问题当成单标签问题进行求解, 忽略了多标签数据集的所有特性; BR (Binary Relevance)^[2] 方法是通过假设标签之间相互独立的前提从而采用一对一的策略, 思路简单且容易理解, 可以利用现有的单标签算法对处理后数据集进行直接

处理, 但其忽略了标签间的相关性^[3], 很大程度上影响了分类的效果; LP (Label Power-set)^[4] 方法考虑到标签间的相关性, 将每个样本所含的标签全部融合成一个新的标签, 但该方法可能导致训练样本不足以及训练复杂度过高等问题。算法转化法是通过传统的分类方法进行改进, 使其能适应于多标签数据的分类, 该方法没有对特定的单标签方法进行限定, 主要有基于 BP 神经网络^[5]、Adaboost^[6]、 K 近邻方法^[7]等的多标签分类方法。集成方法主要是为了实现更好的分类效果, 对常用的问题转换方法和算法转换方法进行结合来处理多标签分类问题。如: RAkEL (Random k-labelsets)^[8] 方法是基于 LP 方法的集成算法, 此方法在考虑标签间相关性的同时, 避免了 LP 方法存在的缺点; ECC (Ensemble Chain Classifiers)^[9] 方法是使用 CC 作为基分类器来进行集成的方法, 降低了 CC 方法在分类准确率方面的负面影响。

K 近邻多标签分类方法是使用较多的多标签分类方法, 但是在此方法中没有考虑标签间的相关性, 从而使得其在分类时不能取得很好的性能。本文针对这类问题提出了一种加

收稿日期: 2015-06-01; **修回日期:** 2015-06-24。 **基金项目:** 安徽省科技攻关计划项目 (1301b042020); 高等学校博士学科点专项科研基金资助项目 (20133401110009); 安徽大学研究生学术创新项目 (Ygh100166)。

作者简介: 檀何凤 (1990-), 女, 安徽安庆人, 硕士研究生, 主要研究方向: 机器学习、人工智能; 刘政怡 (1978-), 女, 安徽芜湖人, 副教授, 博士, 主要研究方向: 人工智能。

入标签相关性后的 K 近邻多标签分类算法,有效地解决了标签相关性的问题,并取得了良好的分类效果。

1 相关介绍

1.1 多标签问题的定义

设示例空间为 $X = \{x_1, x_2, \dots, x_q\}^{[10]}$, 标签空间为 $Y = \{y_1, y_2, \dots, y_q\}^{[11]}$, 则多标签学习算法的训练集表示为 $D = \{(x_i, Y_i) \mid 1 \leq i \leq m, x_i \in X, Y_i \subseteq Y\}$ 。多标签学习算法的主要任务是从训练集 D 中学习得到一个多标签分类函数 $f: X \times Y \rightarrow \mathbf{R}$, $f(x, y)$ 可以看作示例 x 具有类别标签 y 的最大可能性,从而根据此分类函数来判断未知示例所含有的标签集合。

1.2 ML-KNN 算法

ML-KNN 算法是采用 K 近邻分类准则,在求出样本的 K 个近邻后统计近邻样本所包含的标签信息,通过最大化后验概率的方式预测未见示例的标签集合。

已知未见示例 x 和其对应的标签集 y 且有 $y \subseteq Y$, 令 y_x 表示样本的标签集向量,对于每一个取自 Y 的标签 l ,在示例 x 含有 l 时, y_x 的分量 $y_x(l) = 1$; 否则取值为 0。设 $N(x)$ 表示未见示例 x 在训练集中的 k 个最近邻集, $C_x(l)$ 表示近邻集合 $N(x)$ 将 l 作为其相关标签的样本个数, H_l^i 为示例 x 含有标签 l 的事件, H_0^i 为示例 x 不含有标签 l 的事件; $E_j^i (0 \leq j \leq |N(x)|)$ 表示在 x 的近邻中有 j 个示例含有标签 l 的事件。

基于贝叶斯概率公式的 ML-KNN 方法的分类函数如式(1)所示:

$$y_x(l) = \arg \max_{b \in \{0,1\}} \frac{P(H_b^i)P(E_j^i | H_b^i)}{P(E_j^i)} = \arg \max_{b \in \{0,1\}} P(H_b^i)P(E_j^i | H_b^i) \quad (1)$$

即通过式(1)值的大小来确定示例 x 是否包含标签 l 。

对于每一个单独的类别 y_j , 其对应的先验概率可以由式(2)得到:

$$P(H_1^i) = \frac{s + \sum_{i=1}^m \tilde{y}_{x_i}}{2s + m}, P(H_0^i) = 1 - P(H_1^i) \quad (2)$$

后验概率 $P(E_j^i | H_b^i)$ 可以由式(3) ~ (4) 得到:

$$P(E_j^i | H_1^i) = \frac{s + c[j]}{(s | N(x) | + 1) + \sum_{p=0}^{|N(x)|} c[p]} \quad (3)$$

$$P(E_j^i | H_0^i) = \frac{s + c'[j]}{(s | N(x) | + 1) + \sum_{p=0}^{|N(x)|} c'[p]} \quad (4)$$

2 本文算法

本文采用二阶处理方法,对每一个标签 $y_i (1 \leq i \leq q)$ 求出它与其他标签的关系。设 F_{ij} 表示在未见示例 x 中 H_i 成立时, H_j 成立的事件; \bar{F}_{ij} 表示在未见示例 x 中 H_i 不成立时, H_j 成立的事件。

$$\begin{cases} P(F_{ij}) = P(H_j | H_i) = P(H_j H_i) / P(H_i) \\ P(\bar{F}_{ij}) = P(H_j | \neg H_i) = P(H_j \neg H_i) / P(\neg H_i) \end{cases} \quad (5)$$

其中: $j > i$, 且

$$P(H_j H_i) = \frac{s + \sum_{i=1}^m (y_i \in Y_i \text{ and } y_j \in Y_i)}{2s + m} \quad (6)$$

$$P(H_j \neg H_i) = \frac{s + \sum_{i=1}^m (y_i \in Y_i \text{ and } y_j \notin Y_i)}{2s + m} \quad (7)$$

$P(H_j H_i)$ 表示未见示例 x 同时具有标签 y_i 和 y_j 的概率, 而 $P(H_j \neg H_i)$ 表示未见示例 x 具有标签 y_j 而不具有 y_i 的概率, 将式(1)、(6)、(7)代入式(5)求得 $P(F_{ij})$ 和 $P(\bar{F}_{ij})$ 。

$$(F_j, m) = \arg \max_{1 \leq i \leq j-1} (P(F_{ij}))$$

$$(\bar{F}_j, n) = \arg \max_{1 \leq i \leq j-1} (P(\bar{F}_{ij}))$$

其中: F_j 表示 $P(F_{ij})$ 中的最大值, 即当 H_m 成立时, H_j 成立的概率的最大值; \bar{F}_j 表示 $P(\bar{F}_{ij})$ 中的最大值, 即当 H_n 不成立时, H_j 成立的概率的最大值; 在求得 F_j 和 \bar{F}_j 的过程中充分考虑了标签之间的相关性, m 和 n 分别表示在对应最大值时的标签。利用求得的最大值和其预测值结合到式(2)中来预测标签 y_i 的值, 即有分类函数:

$$f(x, y_i) = \frac{P(H_j) [a \times P(C_j | H_j) + (1 - a) \times F_j \times \text{prection}(m)]}{P(\neg H_j) [a \times P(C_j | \neg H_j) + (1 - a) \times \bar{F}_j \times \text{prection}(n)]} \quad (8)$$

其中: $\text{prection}(m)$ 和 $\text{prection}(n)$ 分别表示对应最大值的标签的预测值; a 值是为了调节标签相关性的影响度, 从而得到使分类效果最好时的比例关系。

算法 CML-KNN。

输入 训练数据集 X 和标签集 Y , 设置近邻数 K , 平滑参数 s , 参数 a 。

输出 未见示例 x 对应的标签集合。

```

1) for  $i \in \{1, 2, \dots, q\}$  do
2)   for  $j \in \{1, 2, \dots, m\}$  do
3)     calculate label counting  $P(H_j)$  and  $P(\neg H_j)$  according to equation (2);
4)   end
5)   for  $k \in \{1, 2, \dots, m\}$  do
6)     calculate  $F_{ij}$  and  $\bar{F}_{ij}$  according to equation (5);
7)     set  $m$  and  $n$  according to the  $F_{ij}$  and  $\bar{F}_{ij}$ ;
8)   end
9) end
10) for  $i \in \{1, 2, \dots, m\}$  do
11)   identify  $N(x)$  for  $x$ ;
12) end
13) for  $j \in \{1, 2, \dots, q\}$  do
14)   for  $k \in \{1, 2, \dots, |N(x)|\}$ 
15)     calculate  $P(C_j | H_j)$  and  $P(C_j | \neg H_j)$  for  $y_i$  according to equation (3);
16)   end
17) end
18) for  $y_i \in Y_i$  do
19)   set  $f(x, y_i)$  according to equation (8)
20)   if  $f(x, y_i) > 1$ , set  $y_i$  to +1,
21)   else set  $y_i$  to 0;
22) end

```

3 实验结果与分析

为了评测本文算法的性能, 在 MULAN 平台上实现了 CML-KNN 算法。MULAN 是一个基于 WEKA 的开源项目, 包含一些常用的多标签分类和排序算法^[10]。

3.1 评价指标

多标签分类算法所使用的评价指标跟单标签分类算法不

同,单标签分类算法主要采用准确率、查全率和精度等指标,本文采用如下评价指标度量多标签分类算法的性能。

1) hamming loss^[12]:

$$hloss_s(h) = \frac{1}{p} \sum_{i=1}^p \frac{1}{Q} |h(x_i) \Delta Y_i| \quad (9)$$

其中: Δ 用来度量两个集合之间的对称差, $|*|$ 为返回集合的大小。此指标用于考虑标签之间的对称差,即在预测样本中不属于该样本的标签出现了而属于该样本的标签没有出现。

2) one-error:

$$one-error(f) = \frac{1}{p} \sum_{i=1}^p |[\arg \max_{y \in Y} f(x_i, y)] \notin Y_i| \quad (10)$$

其中:对于任意的谓词 π ,当 π 成立时 $[\pi]$ 取值为1,否则 $[\pi]$ 取值为0。此指标考察在样本的预测标签排序序列中,在序列前面的标签不属于样本标签集合的情况。

3) coverage:

$$coverage(f) = \frac{1}{p} \sum_{i=1}^p \max_{y \in Y_i} rank_f(x_i, y) - 1 \quad (11)$$

此指标考察在样本的预测标签排序序列中,覆盖属于样本的所有标签需要的搜索深度情况。

4) ranking loss:

$$rloss(f) = \frac{1}{p} \sum_{i=1}^p \frac{1}{|Y_i| |Y_i|} |\{(y', y'') \in Y_i \times Y_i \mid f(x_i, y') \leq f(x_i, y''), (y', y'') \in Y_i \times Y_i\}| \quad (12)$$

此指标考察在样本的预测标签排序序列中不相关标签排在相关标签前的情况。

5) average precision:

$$avgprec(f) = \frac{1}{p} \sum_{i=1}^p \frac{1}{|Y_i|} \frac{|\{y' \mid rank_f(x_i, y') \leq rank_f(x_i, y), y' \in Y_i\}|}{rank_f(x_i, y)} \quad (13)$$

此指标考察在样本的预测标签排序序列中,在序列前面的标签属于样本标签集合的情况。

上述五种评价指标中,前四种指标都是在值越小的时候性能越好,average precision 是值越大性能越好。

3.2 数据集

实验分别采用音乐、生物、文本等三个不同领域的数据集来进行结果的比较,数据集的具体信息如表1所示。

表1 数据集

name	instances	labels	cardinality	density
Emotions	593	6	1.869	0.311
Yeast	2417	14	4.237	0.303
Enron	1702	53	3.378	0.064

表1中:cardinality表示标签的基数,为样本的平均标签个数;density表示标签的密度,为标签的基数与标签总数的比值。

3.3 实验结果及分析

实验中各对比算法均采用交叉验证方式,其中:表2~3表示的分别是近邻数 K 为6和11时的实验结果;表4~6为各算法在Emotions、Yeast与Enron数据集上的对比实验结果。符号“↓”表示该指标越小性能越好,“↑”表示该指标越大性能越好,加粗表示在同一行中对应参数下性能最好的结果。

从表2~3可看出:近邻数越多,分类效果越好,这主要是因为 K 值越大,近邻中包含的信息越大,从而能更好地利用标签间的相关性使得预测更加准确,但同时也会使复杂性增加; α 的取值不是越大越好,当 $\alpha=1$ 时对应的是ML-KNN算法,当 $\alpha=0.8$ 时,CML-KNN算法的各评价指标的性能最好。

表4~6用到的对比算法分别为ML-KNN、AdaBoostMH、RAkEL、BPMLL,其中ML-KNN和AdaBoostMH算法属于一阶处理方法,BPMLL是二阶处理方法,而RAkEL是高阶的多标签分类算法。比较实验中CML-KNN和ML-KNN的 K 取值都为11, α 取值为0.8。在表4中的实验结果表明,CML-KNN算法在Emotions数据集上的性能均优于其他算法;在Yeast数据集上CML-KNN算法在评价指标ranking loss上的性能稍低于ML-KNN算法,但是在其他性能指标上的结果均优于其他算法;表6中CML-KNN算法在数据集Enron的hamming loss和one-error评价指标上性能仅差于RAkEL算法,但是在其他

表2 $K=6$ 时原始算法与不同 α 值的本文算法的实验结果比较

数据集	评价指标	ML-KNN	CML-KNN		
			$\alpha=0.75$	$\alpha=0.8$	$\alpha=0.9$
Emotions	hamming loss ↓	0.1965 ± 0.0269	0.1971 ± 0.0260	0.1928 ± 0.0273	0.1945 ± 0.0263
	one-error ↓	0.2688 ± 0.0807	0.2768 ± 0.0733	0.2685 ± 0.0806	0.2718 ± 0.0745
	coverage ↓	1.7800 ± 0.1615	1.7597 ± 0.1522	1.7631 ± 0.1626	1.7580 ± 0.1543
	ranking loss ↓	0.1633 ± 0.0294	0.1625 ± 0.0291	0.1622 ± 0.0306	0.1614 ± 0.0296
	average precision ↑	0.8018 ± 0.0398	0.8025 ± 0.0378	0.8039 ± 0.0404	0.8039 ± 0.0386
Yeast	hamming loss ↓	0.1920 ± 0.0100	0.1932 ± 0.0112	0.1905 ± 0.0108	0.1918 ± 0.0113
	one-error ↓	0.2400 ± 0.0267	0.2387 ± 0.0196	0.2329 ± 0.0262	0.2363 ± 0.0239
	coverage ↓	6.2287 ± 0.1890	6.2071 ± 0.2346	6.1939 ± 0.2207	6.2034 ± 0.2358
	ranking loss ↓	0.1679 ± 0.0128	0.1674 ± 0.0123	0.1657 ± 0.0123	0.1667 ± 0.0123
	average precision ↑	0.7627 ± 0.0200	0.7646 ± 0.0166	0.7666 ± 0.0177	0.7656 ± 0.0177
Enron	hamming loss ↓	0.0523 ± 0.0019	0.0523 ± 0.0018	0.0521 ± 0.0019	0.0522 ± 0.0018
	one-error ↓	0.3096 ± 0.0242	0.3278 ± 0.0329	0.3078 ± 0.0264	0.3237 ± 0.0288
	coverage ↓	13.2002 ± 1.059	13.1726 ± 1.0440	13.1574 ± 1.0455	13.1632 ± 1.0494
	ranking loss ↓	0.0938 ± 0.0095	0.0929 ± 0.0095	0.0928 ± 0.0095	0.0929 ± 0.0095
	average precision ↑	0.6290 ± 0.0148	0.6295 ± 0.0155	0.6354 ± 0.0153	0.6294 ± 0.0154

评价指标上,CML-KNN 算法的性能均显著优于其他所有比较算法。从实验分析可知,CML-KNN 算法在真实数据集上的分类效果明显优于其他对比分类算法。

表 7 中表示的是本文算法同针对 ML-KNN 改进后的这一类算法在数据集 Yeast 上的实验结果比较,其中:GML-

KNN^[13]是基于粒计算的 ML-KNN 算法,IMLLA 是将近邻间的相关性加入 ML-KNN 算法后改进得到的算法。从表 7 可看出,CML-KNN 算法的性能明显优于 ML-KNN 算法与 GML-KNN 算法,且分类性能与 IMLLA 算法接近,已达到 IMLLA 算法的分类效果。

表 3 K=11 时原始算法与不同 a 值得本文算法的实验结果比较

数据集	评价指标	ML-KNN	CML-KNN		
			a=0.75	a=0.8	a=0.9
Emotions	hamming loss ↓	0.191 8 ±0.028 0	0.194 3 ±0.019 7	0.188 4 ±0.019 8	0.192 1 ±0.022 9
	one-error ↓	0.265 2 ±0.079 6	0.265 0 ±0.083 3	0.258 3 ±0.097 6	0.264 7 ±0.089 1
	coverage ↓	1.743 0 ±0.157 7	1.715 7 ±0.131 4	1.705 8 ±0.189 2	1.714 3 ±0.123 8
	ranking loss ↓	0.153 4 ±0.023 3	0.151 8 ±0.031 8	0.148 2 ±0.033 4	0.150 9 ±0.031 5
	average precision ↑	0.809 3 ±0.041 6	0.806 6 ±0.041 3	0.816 1 ±0.043 4	0.810 3 ±0.041 8
Yeast	hamming loss ↓	0.192 1 ±0.012 1	0.193 7 ±0.011 5	0.190 8 ±0.010 4	0.191 7 ±0.011 2
	one-error ↓	0.229 6 ±0.028 5	0.231 7 ±0.021 8	0.226 7 ±0.028 4	0.230 0 ±0.022 9
	coverage ↓	6.181 6 ±0.223 8	6.196 9 ±0.225 0	6.168 8 ±0.222 6	6.184 1 ±0.218 7
	ranking loss ↓	0.163 3 ±0.012 8	0.167 2 ±0.013 9	0.164 3 ±0.013 7	0.165 8 ±0.012 7
	average precision ↑	0.764 5 ±0.020 6	0.765 4 ±0.020 2	0.768 2 ±0.020 4	0.767 9 ±0.020 5
Enron	hamming loss ↓	0.052 5 ±0.001 9	0.052 5 ±0.001 8	0.052 0 ±0.002 5	0.052 2 ±0.001 8
	one-error ↓	0.306 7 ±0.034 2	0.322 6 ±0.029 1	0.307 3 ±0.028 2	0.320 2 ±0.028 0
	coverage ↓	13.081 4 ±1.003 0	13.063 8 ±0.983 6	13.040 3 ±0.983 6	13.047 9 ±0.979 9
	ranking loss ↓	0.093 1 ±0.009 5	0.092 4 ±0.009 3	0.092 0 ±0.009 3	0.092 3 ±0.009 2
	average precision ↑	0.629 9 ±0.022 1	0.627 4 ±0.022 0	0.633 7 ±0.019 9	0.629 5 ±0.021 6

表 4 各算法在 Emotions 数据集上的实验结果比较

评价指标	CML-KNN	ML-KNN	AdaboostMH	RAkEL	BPMLL
hamming loss ↓	0.188 4 ±0.019 8	0.191 8 ±0.028 0	0.304 1 ±0.001 8	0.218 1 ±0.025 2	0.198 2 ±0.031 3
one-error ↓	0.258 3 ±0.097 6	0.265 2 ±0.079 6	0.554 9 ±0.041 6	0.313 7 ±0.073 6	0.298 5 ±0.067 3
coverage ↓	1.705 8 ±0.189 2	1.743 0 ±0.157 7	3.255 4 ±0.260 9	1.941 4 ±0.169 8	1.723 7 ±0.212 5
ranking loss ↓	0.148 2 ±0.033 4	0.153 4 ±0.023 3	0.423 4 ±0.039 2	0.189 3 ±0.035 7	0.156 2 ±0.212 5
average precision ↑	0.816 1 ±0.043 4	0.809 3 ±0.041 6	0.567 9 ±0.029 6	0.775 5 ±0.040 7	0.802 2 ±0.043 3

表 5 各算法在 Yeast 数据集上的实验结果比较

评价指标	CML-KNN	ML-KNN	AdaboostMH	RAkEL	BPMLL
hamming loss ↓	0.190 8 ±0.010 4	0.192 1 ±0.012 1	0.231 8 ±0.008 8	0.225 8 ±0.010 4	0.223 0 ±0.011 5
one-error ↓	0.226 7 ±0.028 4	0.229 6 ±0.028 5	0.248 7 ±0.033 6	0.292 5 ±0.033 0	0.240 0 ±0.029 0
coverage ↓	6.168 8 ±0.222 6	6.181 6 ±0.223 8	9.097 6 ±0.221 1	7.488 4 ±0.187 6	6.532 8 ±0.221 2
ranking loss ↓	0.164 3 ±0.013 7	0.163 3 ±0.012 8	0.371 2 ±0.024 2	0.213 6 ±0.011 0	0.181 2 ±0.014 1
average precision ↑	0.768 2 ±0.020 4	0.764 5 ±0.020 6	0.598 ±0.020 9	0.715 5 ±0.019 5	0.751 8 ±0.019 5

表 6 各算法在 Enron 数据集上的实验结果比较

评价指标	CML-KNN	ML-KNN	AdaboostMH	RAkEL	BPMLL
hamming loss ↓	0.052 0 ±0.002 5	0.052 5 ±0.001 9	0.062 4 ±0.001 8	0.048 5 ±0.002 3	0.239 0 ±0.059 5
one-error ↓	0.307 3 ±0.028 2	0.306 7 ±0.034 2	0.463 6 ±0.040 9	0.281 5 ±0.052 3	0.743 8 ±0.120 7
coverage ↓	13.040 3 ±0.983 6	13.081 4 ±1.003 0	28.493 6 ±0.700 1	25.097 8 ±1.332 0	16.457 2 ±2.320 7
ranking loss ↓	0.092 0 ±0.009 3	0.093 1 ±0.009 5	0.239 6 ±0.012 0	0.200 5 ±0.028 0	0.155 8 ±0.042 9
average precision ↑	0.633 7 ±0.019 9	0.629 9 ±0.022 1	0.454 3 ±0.061 8	0.615 7 ±0.028 0	0.368 2 ±0.091 1

表 7 基于 ML-KNN 改进的各算法在 Yeast 数据集上的实验结果比较

评价指标	CML-KNN	ML-KNN	GML-KNN	IMLLA
hamming loss ↓	0.190 8 ±0.010 4	0.192 1 ±0.012 1	0.198 0 ±0.011 5	0.191 0 ±0.010
one-error ↓	0.226 7 ±0.028 4	0.229 6 ±0.028 5	0.228 0 ±0.023 5	0.227 0 ±0.038
coverage ↓	6.168 8 ±0.222 6	6.181 6 ±0.223 8	6.328 0 ±0.221 9	6.167 0 ±0.232
ranking loss ↓	0.164 3 ±0.013 7	0.163 3 ±0.012 8	0.168 0 ±0.013 1	0.162 0 ±0.014
average precision ↑	0.768 2 ±0.020 4	0.764 5 ±0.020 6	0.764 0 ±0.020 2	0.770 0 ±0.021

4 结语

在多标签分类中,标签间的相关性是一个不可忽略的因素,针对已有的 ML-KNN 算法无法有效利用标签之间相关性的缺陷,提出了一种基于标签相关性的 K 近邻多标签分类算法——CML-KNN。该算法在对某个未见示例的标签进行预测时,利用标签间的条件概率来描述标签对的关系,并对条件概率进行排序,找到标签间的最强相关性,包含了蕴含于其他标签间的相关信息。实验结果表明,CML-KNN 算法的性能优于 ML-KNN 算法以及其他一些常用的多标签分类算法。在今后的工作中,将融合 IMMLA 中近邻间相关信息进行深入研究,从而得到更好的分类效果。

参考文献:

- [1] TSOU MAKAS G, KATAKIS I. Multi-label classification: an overview[J]. Database Technologies Concepts Methodologies Tools and Applications, 2007, 2007(3): 1-13.
- [2] BOUTELL M R, LOU J, SHEN X, *et al.* Learning multi-label scene classification[J]. Pattern Recognition, 2004, 37(9): 1757-1771.
- [3] HUANG S-J, ZHOU Z-H. Multi-label learning by exploiting label correlations locally[C]// AAAI 2012: Proceedings of the 26th AAAI Conference on Artificial Intelligence. Menlo Park: AAAI Press, 2012: 949-955.
- [4] ZHANG M, ZHANG K. Multi-label learning by exploiting label dependency[C]// KDD 2010: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2010: 999-1008.
- [5] ZHANG M, ZHOU Z. Multi-label neural networks with applications to functional genomics and text categorization[J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(10): 1338-1351.
- [6] SCHAPIRE R E, SINGER Y. BoostTexter: a boosting-based system for text categorization[J]. Machine Learning, 2000, 39(2/3): 135-168.
- [7] ZHANG M-L, ZHOU Z-H. ML-KNN: a lazy learning approach to multi-label learning[J]. Pattern Recognition, 2007, 40(7): 2038-2048.
- [8] TSOU MAKAS G, VLAHAVAS I. Random k-Labelsets: an ensemble method for multi-label classification[C]// ECML 2007: Proceedings of the 18th European Conference on Machine Learning, LNCS 4701. Berlin: Springer-Verlag, 2007: 406-417.
- [9] READ J, PFAHRINGER B, HOLMES G, *et al.* Classifier chains for multi-label classification[C]// ECML PKDD 2009: Proceedings of the 2009 European Conference on Machine Learning and Knowledge Discovery in Databases, LNCS 5782. Berlin: Springer-Verlag, 2009: 254-269.
- [10] ZHANG W, WANG C, LIU Z. *et al.* A multi-label classification algorithm based on random walk model[J]. Chinese Journal of Computers, 2010, 33(8): 1418-1426. (郑伟, 王朝坤, 刘璋, 等. 一种基于随机游走模型的多标签分类算法[J]. 计算机学报, 2010, 33(8): 1418-1426.)
- [11] ZHANG M. An improved multi-label lazy learning approach[J]. Chinese Journal of Computer Research and Development, 2012, 49(11): 2271-2282. (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.)
- [12] GAO W, ZHOU Z. On the consistency of multi-label learning[J]. Artificial Intelligence, 2013, 199(3): 22-44.
- [13] CHEN X, WU T, GAO Z. K-nearest neighbor multi-label learning algorithm based on granular computing[J]. Computer Engineering, 2012, 38(22): 167-171. (陈小波, 吴涛, 高正龙. 基于粒计算的K近邻多标签学习算法[J]. 计算机工程, 2012, 38(22): 167-171.)

(上接第2760页)

4 结语

考虑到 ELM 网络中,输出权重广义逆时间复杂度和空间复杂度过大导致的训练速度问题,本文设计了一种基于共轭梯度算法的 CG-RLM 网络。共轭梯度法占用内存小,且在很少的迭代步数内终止,保证了 CG-ELM 网络在保持精度的同时取得了较 ELM 更快的训练和测试速度,仿真实验也验证了本文方法的有效性,尤其是对大型数据,CG-ELM 更为适用。本文同时也说明了 ELM 算法快速的原因并不在于利用广义逆对输出权重的计算,因为本文的共轭梯度迭代算法就比广义逆运算更快速,而在于输入权重和隐含层偏置的随机选择,因为这种随机的赋权避免了反复迭代调整的时间,从而节约了训练时间。

参考文献:

- [1] TOH K A. Deterministic neural classification[J]. Neural Computing, 2008, 20(6): 1565-1595.
- [2] RUMELHART D E, McCLELLAND J L. Parallel distributed processing: explorations in the microstructure of cognition[M]. Cambridge: MIT Press, 1986.
- [3] SIEW C-K, HUANG G-B, ZHU Q-Y. Extreme learning machine: theory and applications[J]. Neurocomputing, 2006, 70: 489-501.
- [4] EMILIO S O, JUAN G S, MARTIN J D, *et al.* BELM: Bayesian extreme learning machine[J]. IEEE Transactions on Neural Networks, 2011, 22(3): 505-509.
- [5] FRENAY B, VERLEYSEN M. Using SVMs with randomised feature spaces: an extreme learning approach[EB/OL]. [2014-10-10]. <http://bfrenay.files.wordpress.com/2013/11/slides3.pdf>.
- [6] MOHAMMED A A, MINHAS R, JONATHAN Q M, *et al.* Human face recognition based in multidimensional PCA and extreme learning machine[J]. Pattern recognition, 2011, 44(10/11): 2588-2597.
- [7] TOH K A. Deterministic neural classification[J]. Neural Computing, 2008, 20(6): 1565-1595.
- [8] HUANG G, ZHOU H, DING X, *et al.* Extreme learning machine for regression and multi-class classification[J]. IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics, 2012, 42(2): 513-529.
- [9] HESTENES M R, STIEFEL E L. Methods of conjugate gradients for solving linear systems[J]. Journal of Research National Bureau Standards, 1952, 49(6): 99-147.
- [10] HU M. Matrix calculation and application[M]. Beijing: Science Press, 2008: 409-446. (胡茂林. 矩阵计算与应用[M]. 北京: 科学出版社, 2008: 409-436.)
- [11] SERRE D. Matrices: theory and applications[M]. Berlin: Springer-Verlag, 2002.
- [12] XUE Y. Numerical analysis and scientific computing[M]. Beijing: Science Press, 2011: 226-230. (薛毅. 数值分析与科学计算[M]. 北京: 科学出版社, 2011: 226-230.)