基于规则置信度调整的关联文本分类*)

林 堃 白清源 谢丽聪 谢伙生 张 莹
(福州大学数学与计算机科学学院 福州 350002)

摘 要 基于关联规则的文本分类方法 ARC-BC 是目前已知的分类效果最好的关联规则分类算法。本文提出了利用 ARC-BC 分类器的封闭测试的结果对分类器进行调整规则置信度的算法 RCA(Rules Confidence Adjustment),参与正确分类行为次数多于参与错误分类行为次数(即"威信"较高)的规则应该拥有更高的置信度,反之,就赋予更低的置信度。实验结果表明,经过 RCA 算法调整的关联文本分类器的分类效果得到显著提高。

关键词 文本分类,关联规则,置信度,调整

Association Text Classification Based on Adjustment of Confidence of Rules

LIN Kun BAI Qing-Yuan XIE Li-Cong XIE Huo-Sheng ZHAGN Ying
(College of Mathematics and Computer Science , Fuzhou University, Fuzhou 350002)

Abstract The ARC-BC algorithm based on association rules is the best known classification algorithm based on association text classification algorithm. In this paper, we propose an algorithm RCA(Rule Confidence Adjustment) to adjust confidence of rules in classifier according to results of close test of the ARC-BC algorithm: the more actions of classifying correctly , the higher confidence of the rule, And vice versa . The experiments results show that the classification results can be improved significantly by adjusting confidence of rules of the ARC-BC.

Keywords Text classification, Association rules, Confidence

1 引言

文本分类是根据一个文本的内容将它分到一个或者多个预定义好的类,应用于邮件分类、垃圾邮件的过滤,以及网页的分类和搜索。最早的关联规则[3]文本分类是由 Liu ,Hsu 和 Ma 在 1998 年提出的关联分类方法 CBA[1]。CBA[1]将关联规则挖掘过程应用到分类过程中,取得比同样基于规则的决策树分类算法 C4.5 更好的分类效果。由于 CBA[1]分类算法的不足,后来提出了各种改进方法,其中比较著名的有 ARC-BC[2]和 CMAR[3]。

本文提出利用 ARC-BC[2]封闭测试的结果对分类器规则的置信度进行调整,经过若干次的迭代调整后,使得 ARC-BC[2]的分类查准率和分类查全率等都得到显著的提高。

本文内容组织如下:第 2 部分简要介绍一下关联规则挖掘和 ARC-BC[2]分类算法;第 3 部分介绍本文提出的根据 ARC-BC[2]分类器封闭测试结果进行规则的置信度调整算法 RCA(Rules Confidence Adjustment);第 4 部分采用实验数据来比较分类效果;最后对全文作一个简单的总结。

2 关联文本分类的基本思想

现有的基于关联规则的分类方法 ARC-BC 的基本思想是利用现有的关联规则挖掘算法 Apriori[4]产生局部类别中的频繁项,即频繁出现的特征词或特征词项集,然后再用这些频繁项集为规则前件,以类别名为后件构造分类规则,并以此规则集构成分类器对测试样本进行分类:测试样本匹配某类别频繁项集的置信度最高,则判定测试样本属于该类别。

2.1 特征选取

文档的特征提取是对每个类别的文档构造若干个最有区别能力的短语。常用的特征项选择方法是利用评估函数,计算每个特征函数下的该函数值,并将该值作为特征值,然后按特征值高低选取分值较高的特征作为被选中的特征结果。本文采用信息增益作为特征选择的依据。

2.2 文档的表示

现有的文本分类算法大多采用向量空间模型来表示文档。文本分类系统大多采用"词袋法"来表示文本,该方法假设文本中词条出现的次序是无关紧要的,把文本看作一系列无序词条的集合。假设有 m 个无序的特征项,则一个文档可以用以下的特征项向量来表示:

$D_j=(W_{j1},W_{j2},\cdots,W_{jm})$

通常有两种方法来表示:

(1)基于文档统计。若第 i 个特征项在第 j 个文档中出现,则 W_{ji} 为 1,否则为 0 。

(2)基于词频统计。若第 i 个特征项在第 j 个文档中出现 N 次,则 W_{ji} 为 N。

2.3 关联规则的挖掘

在各种关联规则挖掘算法中,最经典、最广泛使用的就是 Agrawal 等设计的 Apriori[4] 算法和 Han 等设计的 FP-Growth[5]算法。

Apriori[4]算法在关联规则挖掘过程中将每个文档当成一个事务,把每个文档的特征项当成事务项目,然后从这些事务中挖掘出支持度大于用户设定的最小支持度 minsup 的频繁项集。

2.4 规则的剪枝

由于利用 Apriori[4]算法挖掘的频繁模式的规模可能相

*)由福州大学科技发展基金(2005-XQ-13、2006-XQ-22、XRC-0511)、福建省教育厅(JB06023)资助。林 堃 硕士研究生,主要研究方向:文本挖掘 。白清源 副教授,研究方向:数据库技术和数据挖掘。

当大，所以需要对挖掘得到的关联规则进行规则剪枝，ARC-BC[2]的剪枝策略是只保留规则前件更一般且规则置信度更高的规则，并利用数据库覆盖方法剪掉不能覆盖至少一个事务的规则。例如：

$R_1: t_1, t_2 \rightarrow C_1$ [confidence：40%]

$R_2: t_1 \rightarrow C_1$ [confidence:50%]

R_2就是那些需要保留下来的规则，而 R_1 则要被剪枝。经过剪枝，则剪去了那些对分类没有帮助的，冗余的分类规则。

2.5 ARC-BC 算法的基本思想

假设文本集 D 中有 n 个类别，分别为 $D_1, D_2, \cdots, D_n$，文档表示文档集 D 第 i 类子集的第 j 个文档，其表示成一个 m（m 为特征空间的大小）维向量

$X_{ij}=\{W_{i1}, W_{i2}, \cdots, W_{im}\}$

当特征空间中的第 k 个特征词出现在该文档 D_i 时，则 W_{ik} 为 1，否则为 0。

ARC-BC[2]算法主要包括两个步骤：

(1)生成分类器阶段

ARC-BC[2]分类算法是应用 Apriori[3]算法分别挖掘每个类别样本文档集中支持度[3]大于最小支持度阈值的频繁模式集；以该类的频繁模式为前件 T，以该类别的类名 C_i 为后件，由此构成一条关联分类规则 $T \Rightarrow C_i$；将所有生成的关联分类规则集合起来就构成了关联规则分类器。

表 1 构成分类器的关联规则例子

病毒，电脑⇒计算机
装备，直升机⇒军事
学生，学校⇒教育
总统，记者，直升机⇒政治
装备，记者⇒军事

关联规则的支持度是 C_i 类样本中匹配频繁模式 T 的样本数与 C_i 类中样本总数的比值。

关联规则的置信度是 C_i 类中样本匹配频繁模式 T 的样本数与整个训练样本集中匹配频繁模式 T 的样本数的比值。

(2)分类预测阶段

由(1)中生成的分类器对测试样本进行分类预测。预测时，用分类规则的前件跟被预测样本 d 匹配（即规则的前件出现在预测样本 d 中），将分类器中与被预测样本 d 匹配的规则按照分类规则的后件（即类别名）分成 $S_1, S_2, \cdots, S_n$。然后分别计算 $S_1, S_2, \cdots, S_n$ 中规则集置信度的总和，则将被预测样本 d 分到总和最大的那个类别中去。

3 调整规则置信度

本文提出的 RCA 调整算法是一种根据分类器封闭测试的结果对分类器中分类规则的置信度进行调整的策略，以到达更好的分类效果。

本文的 RCA 调整算法基于以下的考虑：如果把每一条规则都视为一个“裁判”的话，那么很直观的想法就是，对于那些判断正确次数更多错误判断次数更少的“裁判”应该赋予更高的“威信”，即规则的置信度应该更高，反之，亦然。

3.1 调整因子定义以及度量

定义 1 训练样本 d 在封闭测试时被分类器分类到其应属的类别，这个过程称之为正确分类行为；反之，称为错误分类行为。

设规则 R_i 参与的正确分类行为为 N_i，参与的错误分类行为为 M_i。例如，若某训练样本 d 被分类器判定属于 C_k，匹配了的规则$\{R_1, R_5, R_7\}$：

(1)样本 d 被正确分类到 C_k，则 R_1, R_5, R_7 都分别参与了一次正确分类行为，记 R_1, R_5, R_7 对应的 N_1, N_5, N_7 分别加 1。

(2)样本 d 被错误分类到 C_k，则 R_1, R_5, R_7 都分别参与了一次错误分类行为，记 R_1, R_5, R_7 对应的 M_1, M_5, M_7 分别加 1。

封闭测试后，统计分类器中每个规则参与的正确分类行为次数 N_i 和错误分类行为次数 M_i。

有了本文以上的定义，下面就可以量化地定义规则的“威信”了。

定义 2 定义规则置信度调整因子为：

$$W_i=\frac{2N_i}{N_i+M_i}\ (1\leqslant i\leqslant K) \tag{1}$$

当 $N_i>M_i$ 时，第 i 条规则参与了更多的正确分类行为，则其 W_i 的值大于 1（即“威信”高），则该规则的置信被提高。

当 $N_i<M_i$ 时，第 i 条规则参与了更多的错误分类行为，则其 W_i 的值小于 1（即“威信”低），则该规则的置信度被降低。

当 $N_i=M_i$ 时，第 i 条规则参与的正确分类行为和错误分类行为同样多，W_i 值等于 1，设该规则的置信度保持不变。

公式 1 的构造保证了 W_i 的值控制在 0～2 之间，防止发生规则的置信度“调整过度”的现象。

3.2 调整过程

在本文的 RCA 算法中，分类规则置信度的调整是一个多次迭代的过程：

(1)生成 ARC-BC 分类器，并设初值 $W_i=1(1\leqslant i\leqslant K, K$ 为分类器规则总数)，且应满足 $\sum_{i=1}^{K} W_i=K$；

(2)利用上一步生成的分类器对训练样本进行封闭测试，并根据封闭测试结果统计所有分类规则的 N_i 值和 M_i 值。

(3)根据公式 1 计算每个分类规则的调整因子。$t+1$ 次迭代的规则置信度

$$Conf_i^{t+1}=Conf_i^t W_i \tag{2}$$

由于在调整的过程中要保持 $\sum_{i=1}^{K} W_i=K$，所以对公式 2 进行归一化：

$$Conf_i^{t+1}=\frac{Conf_i^t W_i K}{\sum_{j=1}^{K} W_j} \tag{3}$$

将由公式 3 得到的 $Conf_i^{t+1}$ 作为 $t+1$ 次迭代的规则置信度，再次进行封闭测试。当分类准确率不再明显增长或者达到指定的迭代次数时算法结束。

3.3 RCA 算法过程

下面简要地描述一下本文提出的 RCA 算法。

算法说明：RCA 算法根据规则的“威信”调整分类器的分类规则的置信度，达到更好的分类效果。

输入：由 ARC-BC[2]产生的分类器（即分类规则集 Rule_Set）；训练样本集合 Doc_Train；

输出：经过规则置信度调整的分类器；

算法过程：

```
(1)foreach doument Di in Doc_Train do{
(2)     if Di 被正确分类到 Ck then
(3)     foreach R in matchRule(Di,CK)do
(4)          R.N ++;
(5)     else if Di 被错误分类到 Ck then
(6)     foreach R in matchRule(Di,CK)do
(7)               R.M ++;
(8)}
(9)CountRuleWeight(Rule_Set);
(10)do{
```

(11) Adjustment();
(12) CloseTest();
(13)}while(迭代结束条件成立)

RCA算法第(3)和第(6)行的matchRule(D_i,C_K)指的是文档D_i,匹配类C_K中规则的集合。算法第(9)行CountRuleWeight(Rule_Set)是计算整个分类器中所有分类规则的W值,计算方法见3.1公式1。算法第(11)行的Adjustment()是RCA算法的调整过程,具体如3.2所示。算法第(12)行CloseTest()指ARC-BC算法利用训练样本进行封闭测试,得出的分类结果用来作为下一轮调整的输入。算法第(13)行,迭代结束条件是当分类准确率不再明显增长或者达到指定的迭代次数。

4 实验结果与分析

4.1 样本集的处理

本文使用VC++开发工具,实验数据是从中文自然语言处理开放平台网站获取李荣陆[7]收集的新华社的新闻样本。从中抽取1400个样本,7个类别,分别为计算机,交通,教育,经济,军事,体育,政治;利用信息增益和基于文档统计取50个特征词。为了避免因为类别间样本个数的差异导致对实验的干扰,采取以下策略:每个类别各取200个样本。另外为了使得开放测试的分类效果不会因为测试样本的规模受到影响,本文采取从每个类的200个样本中选100个作为训练样本,另外100个作为测试样本。

4.2 实验度量标准

使用目前常用的度量标准Precision(P)和Recall(R),F1[6]以及Macro-avg(宏平均)[6]和Micro-avg(微平均)[6]。

4.3 ARC-BC与其它分类方法的对比

本文在实验中分别使用ARC-BC算法,KNN算法和SVM算法在实验数据进行分类测试,以下是实验结果。

表2 三种分类算法的分类效果对比

F1	ARC-BC	KNN	SVM
计算机	64.14	76.95	77.47
交通	76.19	83.24	88.36
教育	34.53	87.47	92
经济	72.96	65.62	81.89
军事	28.57	47.35	46.34
体育	78.26	71	83.14
政治	42.52	63.47	79.90
Micro-avg	62.26	70	79.90
Macro-avg	59.57	70.73	78.44

从表2可以看出,采用本文的训练数据和测试数据,ARC-BC的分类效果最差,SVM分类效果最好。

4.4 规则置信度调整的实验结果

在我们的实验中,规则置信度在多次调整后,分类效果有了明显的提高(第0次代表未经调整的ARC-BC分类效果)。

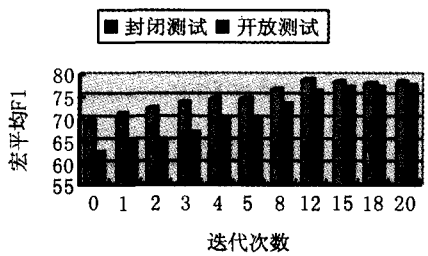

图1 封闭测试和开放测试的宏平均F1

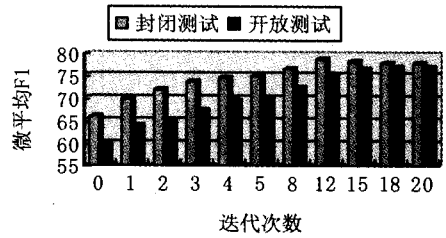

图2 封闭测试和开放测试的微平均F1

在本文的实验中,在迭代了12次之后,分类效果就基本趋于稳定了,RCA算法可以结束。由图1和图2可以看出经过RCA规则置信度调整后的ARC-BC的封闭测试的宏平均F1和微平均F1都较未经调整前的分类结果有显著提高:封闭测试的微平均F1和宏平均F1均提高了17%左右,而开放测试的宏平均F1和微平均F1却提高了近30%。

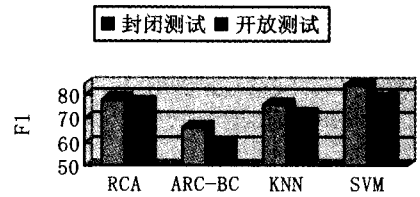

图3 调整后的分类效果比较

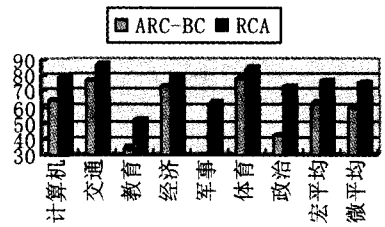

图4 调整前后的各类分类效果(F1)对比

为了简便,本文这里采取F1作为度量单位。在本文的实验中,在迭代了12次之后,分类效果就基本趋于稳定了,RCA算法可以结束。由图1和图2可以看出经过RCA规则置信度调整后的ARC-BC的封闭测试的宏平均F1和微平均F1都较未经调整前的分类结果有显著提高:封闭测试的微平均F1和宏平均F1均提高了17%左右,而开放测试的宏平均F1和微平均F1却提高了近30%,达到了本文根据封闭测试结果调整置信度以提高开放测试分类效果的目的。

由图4中可以看出,各个类别的分类效果(F1)都有了显著提高,特别是原先分类效果较为不好的类别提高的幅度较大,使得总体的分类效果得到大幅度的提高。

表3 分类效果与时间消耗的对比

	ARC-BC		RCA	
分类效果	微平均F1	宏平均F1	微平均F1	宏平均F1
	59.57%	62.26%	75.28%	76.09%
时间消耗	训练时间	测试时间	训练时间	测试时间
	159	3	198	3

表3列出了分类效果的提高与时间消耗的对比。分类效果的提高是明显的,同时也带来了时间上的消耗,但本文的RCA调整算法只是增加了训练时间,测试时间(也就是对新样本的分类时间)并没有增加。ARC-BC挖掘频繁规则时采用的Apriori[3]算法,Apriori[3]算法需要多次扫描事务数据库的特点导致了训练时间较长,并且ARC-BC算法的规则剪枝

过程也消耗了不少时间。本文的 RCA 算法多出的训练时间是由多次迭代封闭测试以及调整过程所消耗的。虽然调整后训练时间增加了，但能大幅度地提高分类器地分类性能。

结束语 本文提出了根据 ARC-BC 封闭测试的分类结果来调整分类器规则置信度的算法 RCA(Rule Confidence Adjustment)，实验结果证明了本文提出的 RCA 算法是有效的，能大幅度地提高分类效果。接下来的工作我们将考虑如何进一步改善关联规则分类器的生成质量，提高挖掘频繁项集的效率。

参 考 文 献

1 Liu B, Hsu W, Ma Y. Integrating Classification and Association Rule Ming . SIGKDD, New York , 1998

2 Aaiane O R, ntonie M. Classifying text documents by associating terms with text categories . Thirteenth Australasian Database Conference(ADC'02) , Melbourne , Australia , 2002

3 Li W, Han J, Pei J. CMAR: Accurate and efficient classification based on multiple classification rules. ICDM, San Jose , California, 2001

4 Rakesh Agrawal Ramakrishnan Srikant. Fast Algorithms for Mining Association Rules. In: Proceedings of the 20th VLDB Conference Santiago, Chile, 1994

5 HAN Jiawei, PEI Jian, YIN Yiwen. Mining Frequent Patterns Without Candidate Generation. ACM-SIGMOD, Dallas, 2000

6 Sebastiani F. Machine Learning in Automated Text Categorization. ACM Computing Surveys, 2002, 34(1): 1～47

7 http://www.nlp.org.cn/docs/download.php? doc_id=1023

(上接第 35 页)

网卡驱动程序在两种情况下对网卡进行发送操作：其一是当 INTAB 发送中断，其二是主机产生了新的发送包。

发生第一种情况时，令 MFS=0；如果 MFW=1，则将 F_W 转为 F_S，并令 MFW=0，MFS=1；如果主机中有待发帧，则将最高优先的帧写入 RTL8139，并令 MFW=1。

发生第二种情况时，如果 MFW=0，则将最高优先的帧写入 RTL8139，并令 MFW=1；如果 MFS=0，则执行第一种情况的操作。其它条件时都不做进一步操作。

参照 Realtek 公司提供的 RTL8139 设计 Demo 方案，利用 Altera 公司的 EPLD 芯片 EPM7032 对 INTAB 中断信号实行数字延时，图 2 是满足间隙整形要求的网卡原理图，图 3 是中断数字延时硬件原理图。

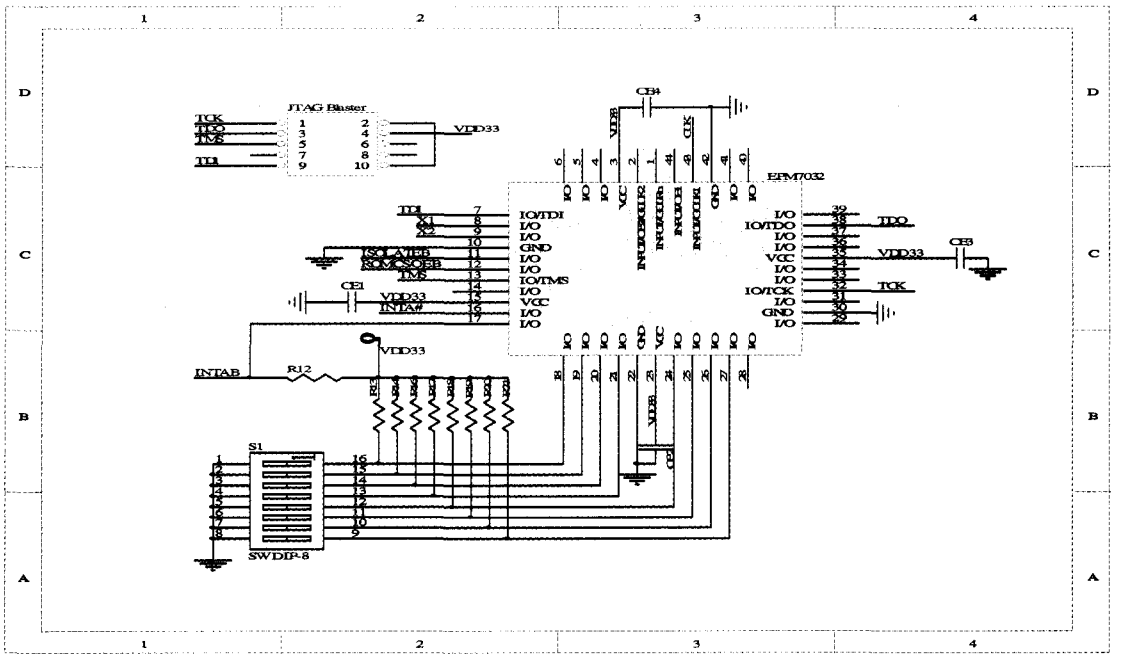

图 3 中断信号处理原理图

实际网络中，下游节点处理包的服务时间 t_S，与下游节点机器有关，不是一个确定不变的常数值，间隙整形需要下游节点，要标明 t_S 的实际值。由于 t_S 随节点而变，显然网卡对 RTL8139 的中断信号 INTAB 延时也应该是可变的。为了调试使用方便，在网卡上设置了一个配置多选择开关 S1，共有 8 个独立输入，分别为 S(1)、S(2)、…、S(8)。当 S(8)有效时，不延时 INTAB 信号；当 S(7)有效时，INTAB 信号不能到达 PCI 总线上，即永久延时；当 S(7)和 S(8)都无效时，由 S(1)、S(2)、…、S(6)组合可选择 64 种预先设定的延时值之一，实现对 INTAB 延时。

5 实现与调试结果

为了证明间隙调整设计的有效性，我们直接利用 RTL8139 网卡原有驱动程序，使用计算机网络 ping 命令进行测试。

首先断开 S(8)，实测传输延时时间与普通 RTL8139 网卡一致。

第二步，接通 S(8)，断开 S(7)，ping 命令超时，符合设计的功能。

第三步，断开 S(7)和 S(8)，S(1)接通，其它的都断开，预设延时值为 0x8000=32768，测得 ping 延时为 1.0ms，而普通网卡小于 1.0ms，说明延时有效。

第四步，断开 S(7)和 S(8)，其它的也全断开，即预设值为 0xFC00=64512，约为第三步预设值的两倍，测得 ping 延时为 1.9ms，延时也约为第三步的两倍。

上述实验结果表明，设计完全达到要求。实际使用时，如果要求的时间更长些，通过修改程序中的预设可选值即可。

参 考 文 献

1 曾华燊. 现代网络通信技术[M]. 成都：西南交通大学出版社，2004. 450～466

2 曾家智，徐洁，吴跃，等. 服务元网络体系结构和微通信元系统构架. 电子学报，2004，32(5)：745～749

3 曾家智，赵继东，蓝勇钢. 微通信元系统架构和 QoS. 计算机应用，2004，24(6)：10～12