ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.17, No.5, May 2006, pp.1017−1025
DOI: 10.1360/jos171017

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于分类规则树的频繁模式文本分类*

陈晓云[1,2+], 陈 祎[1], 王 雷[1], 李荣陆[1], 胡运发[1]

[1](复旦大学 计算机与信息技术系,上海 200433)

[2](福州大学 数学与计算机科学学院,福建 福州 350002)

Text Categorization Based on Classification Rules Tree by Frequent Patterns

CHEN Xiao-Yun[1,2+], CHEN Yi[1], WANG Lei[1], LI Rong-Lu[1], HU Yun-Fa[1]

[1](Department of Computer and Information Technology, Fudan University, Shanghai 200433, China)

[2](School of Mathematics and Computer Science, Fuzhou University, Fuzhou 350002, China)

+ Corresponding author: Phn: +86-591-83733907, E-mail: c_xiaoyun@21cn.com

Chen XY, Chen Y, Wang L, Li RL, Hu YF. Text categorization based on classification rules tree by frequent patterns. *Journal of Software*, 2006,17(5):1017−1025. http://www.jos.org.cn/1000-9825/17/1017.htm

Abstract: Association categorization approach based on frequent patterns has been recently presented, which builds the classification rules according to frequent patterns in various categories and classifies the new text employing these rules. But there are two shortages when the method is applied to classify text data: one is that the method ignores the information about word's frequency in a text; another is that the rule pruning to improve the classification efficiency will lead to obvious descending of accuracy when mass rules are generated. Therefore, a text categorization algorithm based on frequent patterns with term frequency is presented. This study illuminates that the word frequency is helpful for improving the accuracy of the association categorization and the classification rule tree can improve the efficiency of the association classification. The result of experiments shows the performance of association classification is better than three typical text classification methods Bayes, kNN (*k* nearest neighbor) and SVM (support vector machines), so it is a promising text classification method.

Key words: frequent pattern; text categorization; term frequency; association rule; classification rule

摘 要: 基于频繁模式的关联分类是近年来出现的一种分类方法,该方法利用各类别频繁出现的模式构造分类规则,并对新文本进行分类.但现有关联分类方法应用于文本分类时存在两方面不足:一方面,用以构造分类规则的频繁模式仅考虑特征词在文本中出现与否,从而忽视了出现频度;另一方面,当产生的规则数量较多时,为提高分类效率需要进行规则修剪,修剪后的分类准确性明显降低.为此,提出了基于分类规则树的带词频的频繁模式文本分类方法.研究结果表明,词频的引入可以提高关联分类的准确率;而采用分类规则树可使分类时间明显加快又确保不降低分类质量.这两方面的措施弥补了现有关联分类应用于文本分类的不足.与 3 种典型文本分类方法比较后发现,在低维特征空间中,关联分类的性能优于 Bayes,kNN(*k* nearest neighbor)和 SVM(support vector machines),因此是一种很有应用前景的文本分类方法.

* Supported by the National Natural Science Foundation of China under Grant No.60173027 (国家自然科学基金); the Science and Technology Foundation of Education Office of Fujian Province of China under Grant No.JB02069 (福建省教育厅科技基金)

Received 2004-04-15; Accepted 2005-05-08

关键词: 频繁模式;文本分类;词频;关联规则;分类规则
中图法分类号: TP18 文献标识码: A

随着 WWW 应用的普及,在线文本信息迅速增加,文本信息的分类组织是管理海量文本信息的关键.现有的文本分类主要基于统计理论和机器学习方法,其中比较著名的有 Bayes[1],kNN(*k* nearest neighbor)[2],LLSF(linear least squares fit)[2],Nnet(neural network)[3]以及 SVM(support vector machines)[4].Yang[2,5]使用英文标准分类语料对这些分类方法进行充分比较,认为 SVM,kNN 在分类准确性和稳定性方面优于其他分类方法.kNN 是一种懒惰学习方法,保存所有训练样本直至测试样本需要分类时,通过计算样本间距离确定分类,因此其分类时间是非线性的,当训练文本数或特征数增加时,分类时间急剧增加.SVM 目前被认为是分类准确性最好的分类器,当训练样本分布不均匀时,其分类质量比 kNN 还要好.但由于 SVM 本质上是两类分类器,对两类分类问题而言,SVM 是线性分类器.但是,当用 SVM 实现多类分类时,必须构造多个 SVM 分类器,即将多类分类问题转化成两类分类问题,一般使用一对剩余方法(one-vs-best)[4]进行多类分类.对于 k 类分类问题,需要构造 k 个分类器,当类别数增多时,其分类时间明显增加.SVM 的另一个不足就是训练样本数目较大时,内存开销很大,训练时间长.

1998 年,Liu, Hsu 和 Ma 最先提出关联分类方法 CBA(classification based on associations)[6].CBA 集成分类规则挖掘过程和关联规则挖掘过程,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.此后,陆续有人针对 CBA 的不足提出各种改进方法,典型的有 CMAR(classification based on multiple association rules)[7]和 ARC(associative rule-based classifier)[8].这些方法的基本思想是利用现有关联规则挖掘算法[9,10]产生各类别中频繁出现的特征词或特征词项集,利用频繁特征词项集构造分类规则对测试样本进行分类.测试样本包含某类频繁特征词项的数量越多、置信度越高,则认为测试样本属于该类别的可能性就越大.

关联分类方法的分类时间仅与规则数量和测试样本集的规模有关.分类规则一旦确定,除非训练集发生变化,否则对不同测试集进行分类时不必重新训练.但是,现有的关联分类方法还存在以下问题:

(1) 应用于文本分类时,需要考虑特征词在文本中多次出现的情况,目前的关联分类方法都是基于不考虑词频的方法;

(2) 随着规则数的增加,在分类新文本时,需要多次扫描文本,从而降低了分类效率.关联分类算法 CBA,CMAR 和 ARC 通过规则修剪技术对冗余规则进行修剪,但修剪后的分类精度出现不同程度的下降,下降幅度与使用的修剪策略有关[7,8].如果不对规则进行修剪,虽然可以保证分类的准确性,但分类时间难以避免地增加了.

为此,我们提出带词频的关联规则文本分类算法,并利用分类规则树(classification rules tree,简称 CR-tree)存储规则.分类时,通过对 CR-tree 进行深度优先搜索,可以有效地减少每篇待分类文本所需考察的规则数,以确保分类快速又不降低分类精度.

算法主要步骤如下:① 将训练文本表示成特征向量的形式;② 利用频繁模式挖掘算法找出各类别的频繁模式集,每个频繁模式与其所属类的类标号组成该类的一条分类规则,并计算规则置信度;③ 找出所有与测试样本匹配的分类规则,按类别对各类匹配规则的置信度求和,测试样本被分类到置信度之和最高的类别.

1 文本的特征向量表示

定义 **1**(项). 文本的内容特征常用它所含有的特征词来表示,这些词被称为项(term).一个或一个以上的项组成项集.若考虑项在文本中出现的次数(即词频),项表示为 $t_i(w_i)$.后面所提到的项都是指带词频的项.

定义 **2**(文本的向量表示). 给定一个文本 $d=(t_1(w_1),t_2(w_2),\ldots,t_n(w_n))$,当暂时不考虑 t_k 在文本中的先后顺序并要求 $t_k(1\le k\le n)$互异时,可以把 $t_1,t_2,\ldots,t_n$ 看成一个 n 维坐标系,而 $w_1,w_2,\ldots,w_n$ 为相应的坐标值,因而$(w_1,w_2,\ldots,w_n)$可看成是 n 维空间中的一个向量,称$(w_1,w_2,\ldots,w_n)$为文本 d 的向量表示.

可以利用 N-gram 特征提取技术和χ^2 统计技术选取特征词[11],并将训练集中 m 个类别 $C_1,C_2,\ldots,C_m$ 中的文本表示成 m 个特征矩阵.实验中发现,同一类文本的特征词词频大多集中在某一范围内,但也会出现个别高频词.为避免这些高频词产生过多冗余规则,我们定义项 t_k 的最大词频阈值为 $M(t_k)$,当 t_k 的词频低于该阈值时,用实际

词频作为 t_k 的词频;而当词频大于该阈值时,以该阈值作为 t_k 的词频.

例 1:已知训练文本集 C_i 类有 5 篇文本,其向量表示如下:

	I_1	I_2	I_3	I_4	I_5
d_1	3	2	0	0	1
d_2	0	1	4	0	2
d_3	4	5	0	3	0
d_4	2	0	1	0	1
d_5	3	4	2	0	0

2 带词频的频繁模式发现

在训练阶段,利用关联规则发现算法挖掘各类别训练样本子集的频繁模式,并构造分类规则.为便于描述,我们在定义 3 和定义 4 中引入若干记号.

定义 **3**. 项的相似、相等及包含.

(1) 特征词相同而词频相同或不同的两个项 $t_i(w_i)$和 $t_i(w_j)$是相似的,记为$\mu(t_i,t_i)$;

(2) 如果相似的两个项其词频也相同,即 $w_i=w_j$,那么称 $t_i(w_i)$与 $t_i(w_j)$相等,记为 $t_i(w_i)=t_i(w_j)$;

(3) 如果 $w_i>w_j$,称 $t_i(w_i)$包含 $t_i(w_j)$,记为 $t_i(w_i)\supset_w t_i(w_j)$;

(4) 如果 $w_i<w_j$,称 $t_i(w_i)$属于 $t_i(w_j)$,记为 $t_i(w_i)\subset_w t_i(w_j)$.

例 2:{computer(5)}$\supset_w${computer(2)};{computer(5)}={computer(5)}.

定义 **4**(项发生数). 项 t 的发生数是指 t 在文本 d 或项集 X 中出现的次数,记为 $R(t,d)$或 $R(t,X)$.

定义 **5**(支持数). 某文本 d 对项集 X 的支持 $\hat{\varphi}$ 定义为 X 中的各项在 d 中的最小发生数,即

$$\hat{\varphi}(X,d)=\min\left(\left\lfloor\frac{R(x_i\mid x_i\in X,d)}{R(x_i,X)}\right\rfloor_{i=1}^{|X|}\right) \tag{1}$$

其中,$R(x_i\mid x_i\in X,d)$表示项集 X 中的任意一项 x_i 在文本 d 中的发生数,也是 x_i 在文本 d 中的词频;$R(x_i,X)$是项 x_i 在项集 X 中的发生数.

项集支持定义了一篇文本对某项集的支持计算公式,每篇文本对带词频项集的支持各不相同.文本集中所有文本对项集的支持和,就是项集在该文本集中的支持数,即

$$support\_count(X)=\sum_{i=1}^{|D|}\hat{\varphi}(X,d_i) \tag{2}$$

每篇文本对不带词频的项集的支持均为 1,不带词频的项集在文本集中的支持数是包含该项集的文本数.

例 3:在例 1 给出的训练集中,项集$\{I_1(2),I_2(2)\}$的支持数计算如下:

$\hat{\varphi}(I_1(2)I_2(2),d_1)+\hat{\varphi}(I_1(2)I_2(2),d_2)+\hat{\varphi}(I_1(2)I_2(2),d_3)+\hat{\varphi}(I_1(2)I_2(2),d_4)+\hat{\varphi}(I_1(2)I_2(2),d_5)$ =1+0+2+0+1=4.

定义 **6**(支持度). 已知文本集 D,其中每一文本的文本向量为 $d_j=(t_1(w_1),t_2(w_2),\ldots,t_n(w_n))$,项集 X 在文本集 D 中的支持度是 X 在 D 中的支持数与 D 中文本数之比,形式化表示如下:

$$\varphi(X,D)=\frac{\sum_{i=1}^{|D|}\hat{\varphi}(X,d_i)}{|D|} \tag{3}$$

定义 **7**(频繁模式). 给定最小支持度阈值ε,若项集 X 的支持度大等于ε(即$\varphi(X,D)\geq\varepsilon$),称项集 X 在 D 中是频繁的,频繁项集也称频繁模式.

引入词频后频繁模式仍然具有向下封闭的性质,即性质 1.

性质 **1**. 若项集 $t_{i1}(w_{i1}),\ldots,t_{ij}(w_{ij})$是非频繁的,其超集也是非频繁的.

该性质的证明与不含词频的情况类似,因篇幅所限,这里不再给出其详细证明.这里要说明的是,项集 $t_{i1}(w_{i1}),\ldots,t_{ij}(w_{ij})$的超集是指根据定义 3 所定义的包含关系,所有包含 $t_{i1}(w_{i1}),\ldots,t_{ij}(w_{ij})$的项集都是其超集.如 $t_1(3)$

是 $t_1(2)$与 $t_1(1)$的超集;$t_1(2)t_2(2)$是 $t_1(1),t_2(1),t_1(2),t_2(2),t_1(1)t_2(1),t_1(1)t_2(1)$和 $t_1(2)t_2(1)$的超集.由此,在发现带词频的频繁模式集过程中,可利用性质 1 对候选项集进行修剪,减少所需考察的项集数量.

算法 **1**. 发现带词频的频繁模式.

Input: C_i 类训练文本向量集 D_i;最小支持度阈值ε;

Output: C_i 类频繁项集 F.

Method:

(1) For each term $t\in D_i$ do begin

(2) $M[t]=\mathrm{Max}(R(t,d_i)\big|_{i=1}^{|D_i|})$; //$M[t]$是 t 的最大发生数

(3) j=1;

(4) While (j=1) or ($j<M[t]$ and $t(j-1)\in F_1$)

(5) $t(j).support\leftarrow\varphi(t(j),D_i)$;

(6) $F_1=\{t(j)|t(j).support\geq\varepsilon\}$; //$F_1$ 是频繁 1-项集

(7) $j=j+1$;

(8) End

(9) End

(10) For (i=2; $F_{i-1}\neq\varnothing$; i++) do begin

(11) $T_i=(F_{i-1}\bowtie F_{i-1})$; //联接 F_{i-1} 和 F_{i-1} 生成候选 i-项集

(12) $T_i=T_i-\{X|(i-1)\text{term-set of } X\notin F_{i-1}\}$;

(13) For each X in T_i do begin

(14) $X.support=\varphi(X,D_i)$

(15) End

(16) $F_i=\{X\in T_i|X.support\geq\varepsilon\}$; //$F_i$ 为频繁 i-项集

(17) End

(18) $F=\bigcup_i\{X\in F_i\mid i>1\}$ //F 是由所有频繁项集组成的集合

文中的 i-项集是指由 i 个不同关键词组成的项集,如 ab(3)和 cd(1)均是 2-项集.

算法第(1)~(9)行求出所有频繁 1-项集.与不含词频的情况不同,这里,项 t 的频繁 1-项集可能有多个,如 t(1),t(2),t(3)等;并且,若 t(2)不频繁,则 $t(k)$(k>2)一定不频繁(性质 1).

算法第(10)~(18)行产生所有支持度大等于阈值ε的频繁项集 F.其中,通过联接 F_{i-1} 和 F_{i-1} 生成候选 i-项集的方法与 Apriori 算法[9]类似,但联接条件不同;含有词频时的联接条件是:两项集间有且仅有一个关键字不相同的项,且其他关键字相同的项其相应词频必须相同.

例 4:已知例 1 中的 C_i 类训练文本集,最小支持数阈值设为 4,则 C_i 类的频繁模式集 $F=\{I_1(1),I_2(1),I_3(1),I_5(1),I_1(2),I_2(2),I_1(1)I_2(1),I_1(1)I_2(2),I_2(1)I_1(2),I_1(2)I_2(2)\}$.

对于有 m 个类别 $C_1,C_2,\ldots,C_m$ 的训练集 D,D_i 是属于 C_i 类的文本子集,且 $D=\bigcup_{i=1}^{m}D_i$. 由各类别样本集产生的频繁模式集互不相同,但可能存在交集,相交程度由训练样本的分布情况决定.若两个类别比较接近,频繁模式集相交的程度就会大些.显然,出现在交集中的频繁模式,其类别区分度不如出现在非交集中的频繁模式.我们用置信度来描述频繁模式属于某一类的程度.

定义 **8**(分类规则). C_i(i=1,2,…,m)类的分类规则是形如 $X\Rightarrow C_i$ 的蕴含式.其中项集 X 在 D_i 中频繁,X 为规则的条件,C_i 为规则的结论.

定义 **9**(规则置信度). 规则 $X\Rightarrow C_i$ 的置信度定义为

$$\sigma(X\Rightarrow C_i)=\frac{\varphi(X,D_i)\,|D_i|}{\varphi(X,D)\,|D|} \tag{4}$$

规则 $X\Rightarrow C_i$ 的置信度越高,表示模式 X 出现在 C_i 类的程度越高,包含 X 的文本属于 C_i 类的可能性也就越高.若置信度为 100%,则模式 X 只出现在 C_i 类训练样本中,此时模式 X 具有最好的类别区分度.

由例 4 产生的频繁模式集构造出 C_i 类的分类关联规则为

$$I_1(1)\Rightarrow C_i \quad I_2(1)\Rightarrow C_i \quad I_3(1)\Rightarrow C_i \quad I_5(1)\Rightarrow C_i \quad I_1(2)\Rightarrow C_i \quad I_2(2)\Rightarrow C_i$$
$$I_1(1)\wedge I_2(1)\Rightarrow C_i \quad I_1(1)\wedge I_2(2)\Rightarrow C_i \quad I_2(1)\wedge I_1(2)\Rightarrow C_i \quad I_1(2)\wedge I_2(2)\Rightarrow C_i$$

3 用分类规则树分类新文本

3.1 分类器的构造

待分类文本 d 可能包含不同类别的频繁模式,即与不同类别的分类规则相匹配.该如何根据这些匹配规则确定待分类文本的类别呢?直观的做法是将 d 分类至匹配规则中置信度最高的规则所指向的类.CBA[6]采用的就是这种方法.这种方法忽略了其他匹配规则的影响,在某些情况下分类精度较低[7].我们综合考虑所有的匹配规则,通过各类匹配规则的置信度之和来确定 d 的类别.

定义 **10**(类置信度). 定义与文本 d 匹配的所有指向类别 C 的分类规则置信度之和为类别 C 的类置信度,即

$$\Omega(d,C)=\sum_{j=1}^{l} confidence(T_j\Rightarrow C) \tag{5}$$

其中 l 是以类别 C 为目标类的文本 d 的匹配规则数,$T_j\Rightarrow C$ 是与文本 d 匹配的 C 类规则.

定义 **11**(类差异因子). 若有 n 条规则与文本 d 匹配,其中有 k_1 条规则指向 C_1 类,k_2 条规则指向 C_2 类,…,k_m 条规则指向 C_m 类,则对于文本 d,C_j 类的差异因子定义为

$$\delta(d,C_j)=\frac{\underset{i}{\mathrm{Max}}\left(\Omega(d,C_i)\right)-\Omega(d,C_j)}{\underset{i}{\mathrm{Max}}\left(\Omega(d,C_i)\right)}(i,\ j=1,2,...,m) \tag{6}$$

差异因子反映了所有类与最高类置信度之间的差异.在进行兼类分类时,选择差异小于给定阈值 δ 的类作为该文本 d 的分类;若进行单类分类,则 Ω 值最高的类就是文本 d 的分类.

具体分类步骤如下:

步骤 1. 在分类规则集中找出所有与待分类文本 d 匹配的规则(即规则条件模式中的所有项均在文本 d 中出现).

步骤 2. 将匹配规则按类标号分组,同组规则具有相同类标号,计算各组规则的置信度之和,即类置信度 Ω.

步骤 3. 将所有类别按 Ω 降序排序并计算类差异因子.d 被分到类差异因子小于给定阈值的类别中.显然,排在第一位的类具有最大类置信度,其差异因子为 0%.

步骤 1 是分类过程中最耗时的一步,因此,如何减少该步骤所花费的时间是实现快速分类的关键.前面提到,现有关联分类都是通过减少规则数量来提高分类速度的,但因为规则修剪会使分类准确性降低,因此,该方法并不可行.我们利用分类规则树来查找匹配规则,可以极大地提高查找速度又不影响分类质量.

3.2 基于**CR-tree**的匹配规则查找

首先,我们给出以下性质:

性质 **2**. (1) 若规则 R_1:$T_1\Rightarrow C$ 不匹配待分类文本 d,则其超规则也不匹配文本 d.

(2) 若项集 T_1 不匹配待分类文本 d,则其超集也不匹配文本 d.

利用性质 2 可以减少分类过程所需考察的规则数.例如,如果规则 $a(3)\wedge b(1)\Rightarrow C_1$ 不匹配文本 d,那么其超规则 $a(3)\wedge b(1)\wedge e(2)\Rightarrow C_2$ 一定也不匹配 d,分类时无须考虑.

为此,我们借鉴 FP-tree[10]思想,将分类规则用分类规则树存储.分类规则树具有以下特点:

(1) 树仅有一个根节点 R;

(2) 树中任一非根节点到根节点 R 的路径表示一个模式 P,非根节点称为模式节点,记为 P;

(3) 若树中模式节点 P 所代表的模式是规则 R:$P\Rightarrow C$ 的条件,则节点 P 称为规则节点,该节点需记录规则指向类的类标号 C 及规则置信度;

(4) 树中任一节点 P 的子节点 P'所表示的项集是该节点所表示的项集的超集,即 $P\subset_w P'$.

例 5:假设训练阶段发现的分类规则集 R 见表 1.

先将出现在规则中的所有项按出现频率降序排序,结果集记为 L:{b(1):5,a(2):4,c(1):2,d(1):1}.排序后的规则集 R'见表 2.

Table 1 A simple classification rule set R

表 **1** 简单的分类规则集 R

Rule ID	Rule	σ(%)
1	$a(2)\Rightarrow C_1$	80
2	$a(2)b(1)\Rightarrow C_1$	70
3	$a(2)b(1)\Rightarrow C_2$	60
4	$a(2)b(1)c(1)\Rightarrow C_3$	60
5	$b(1)d(1)\Rightarrow C_1$	70
6	$b(1)c(1)\Rightarrow C_3$	60

Table 2 An ordered classification rule set R'

表 **2** 项排序后的分类规则集 R'

Rule ID	Rule	σ(%)
1	$a(2)\Rightarrow C_1$	80
2	$b(1)a(2)\Rightarrow C_1$	70
3	$b(1)a(2)\Rightarrow C_2$	60
4	$b(1)a(2)c(1)\Rightarrow C_3$	60
5	$b(1)d(1)\Rightarrow C_1$	70
6	$b(1)c(1)\Rightarrow C_3$	60

由 R'构造的分类规则树如图 1 所示.

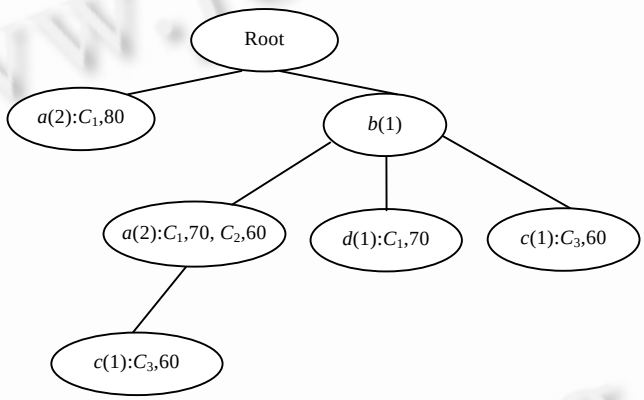

Fig.1 CR-Tree for the rule set R

图 1 规则集 R 的分类规则树

CR-tree 是分类规则集的压缩存储结构.由于在 CR-tree 中,子模式与超模式共享前缀节点,因此可以节约大量存储空间.

根据性质 2,若 CR-tree 中某节点所代表的模式(或规则)与待分类文本不匹配,那么,该节点的所有子节点表示的模式(或规则)也与该文本不匹配.因此,在查找与新文本匹配的规则时,对 CR-tree 树进行深度优先搜索,无须考察不匹配节点的子树,直接考察其右兄弟节点,若不存在右兄弟节点,再回溯到父节点,考察父节点的右兄弟节点.通过 CR-tree 及性质 2 能够有效地减少每篇待分类文本所需考察的规则数.

4 实验结果与算法分析

我们在 PIII 900,256M 内存计算机上用 C++实现了本文的算法.实验所需的数据集主要由从新华网等网站上收集到的新闻网页组成.所有文本均由人工预先整理,并分成政治、经济、军事、交通、环境、计算机、体育、教育、医药、艺术 10 类,其中,训练文本有 2 000 篇,每类 200 篇用来对分类器进行训练;测试文本有 2 315 篇,用来评价分类器性能.分类器性能评价指标使用国际上常用的微平均 F_1 值.

为了对比本文的方法(associative rule-based classifier with CR-tree and recurrent term,简称 TRARC)与其他分类方法,我们在实验数据集上分别应用本文的方法,ARC,kNN,Bayes 和 SVM 分类方法对测试文本进行分类.

在实验中,使用 N-gram 技术和χ^2特征选择方法取得文本特征,特征数均取 60;带词频的关联分类最大词频阈值取 5,最小支持度为 20%;Bayes 使用多项式模型;在 kNN 分类中,K=5;SVM 方法使用多项式核函数,利用一对剩余方法进行多类分类.

对本文的方法 TRARC,从表 3 可以得出以下结论:

- 在关联分类中,以词频为文本特征值优于以布尔值为文本特征值.
- 词频的引入使规则数增加,但 TRARC 的分类时间比 ARC 更快速.从下一个实验中可以看到,这主要归功于 CR-tree 的使用.
- TRARC 在分类准确性和分类效率方面均优于 ARC,但其训练时间比 ARC 要长,这是因为引入词频后 TRARC 所要考察的候选项集数量增加了.

Table 3 Comparison of different text categorization methods

表 **3** 不同文本分类方法的比较

	TRARC (Our approach)	ARC	kNN	Bayes	SVM
Training time (s)	196	51	0.04	0.06	18
Classification time (s)	19	46	60	92	14
The number of rules	5 834	1 641	—	—	—
Micro-average-F_1 (%)	88.9	84.86	80.57	75.4	80.3

关联分类方法的准确性优于 Bayes,kNN 和 SVM.当然,我们取的特征数较低,仅为 60,但这更说明关联分类在低特征数的情况下就可以达到较高的分类质量.当特征数为 1 000 时,kNN 分类的 Micro-average-F_1 值为 89.4%,分类时间为 94s,虽然精确度略高于关联分类方法,但分类速度明显低于 TRARC 和 ARC.

为了考察分类规则树对分类准确性和分类时间的影响,分别对使用规则树和不使用规则树,以及使用规则修剪和未使用规则修剪的情况进行比较测试.

由表 4 可以得出以下结论:

- 使用 CR-tree 明显提高了分类效率.在未经规则修剪的情况下,未使用 CR-tree 的分类时间约为每篇 0.024s;而使用 CR-tree 后,分类时间约为每篇 0.008s.
- 超规则修剪虽然可以加快分类速度,但其分类准确率下降了.这与文献[8]的实验结果一致.

Table 4 The effect on F_1 value and classification time by CR-tree and super rules pruning

表 **4** CR-Tree 及超规则修剪对 F_1 值及分类时间产生的不同影响

	Without CR-tree		With CR-tree	
	With pruning super rules	Without pruning super rules	With pruning super rules	Without pruning super rules
The number of rules	434	5 834	434	5834
Classification time (s)	40	52	5.5	19
Micro-average-F_1 (%)	80.4	88.9	80.4	88.9

图 2~图 4 显示了最大词频阈值变化对 F_1 值、训练时间及分类时间的影响.其中,图 2 说明随着最大词频的增加,F_1 值呈上升趋势,进一步说明引入词频可以有效地提高关联分类器的性能;图 3 则说明随着最大词频阈值的增加,训练时间也增加;但由图 4 可以看出,虽然由于词频的引入使规则数及训练时间增加,但 TRARC 分类时间却保持稳定,而未使用 CR-tree 的 RARC(associative rule-based classifier with recurrent term)分类时间明显增加.

图 5~图 7 显示了最小支持度阈值变化对 F_1 值、训练时间及分类时间的影响.从图 5~图 7 可以看出,随着支持度阈值的增大,训练时间与分类时间均明显下降,而且 TRARC 算法的分类速度明显快于 RARC 算法.

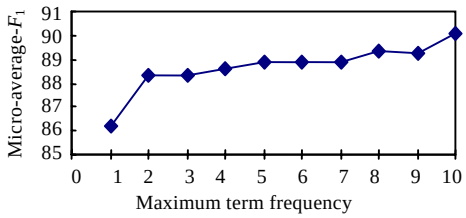

Fig.2 The effect of maximum term frequency on F_1 value

图 2 最大词频变化对 F_1 值的影响

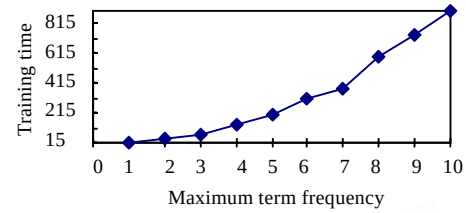

Fig.3 The effect of maximum term frequency on training time (minsup=20%)

图 3 最大词频变化对训练时间的影响(minsup=20%)

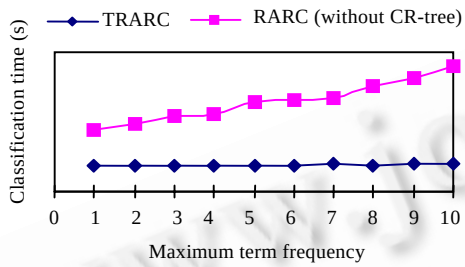

Fig.4 The effect of maximum term frequency on classify time

图 4 最大词频变化对分类时间的影响

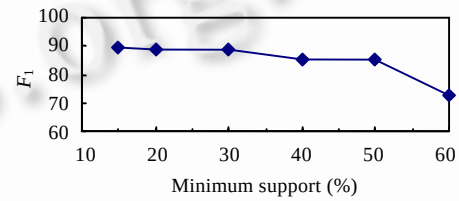

Fig.5 The effect of support threshold on F_1 value (the maximum term frequency is 5)

图 5 支持度阈值变化对 F_1 值的影响(最大词频为 5)

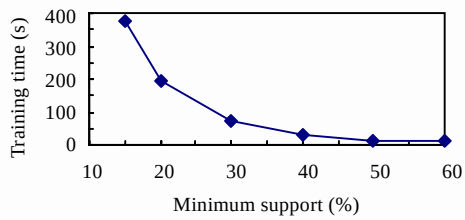

Fig.6 The effect of support threshold on training time

图 6 支持度阈值变化对训练时间的影响

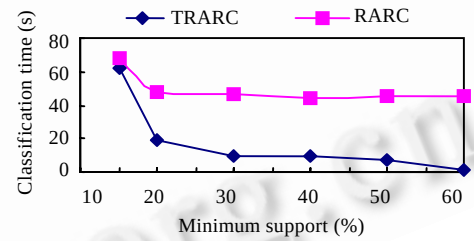

Fig.7 The effect of support threshold on classification time

图 7 支持度阈值变化对分类时间的影响

5 结 论

根据现有关联分类方法应用于文本数据时存在的问题,我们提出了基于分类规则树的带词频的频繁模式分类方法.实验结果表明,当利用频繁模式分类文本集时,考虑词频可以提高分类的准确性;而利用分类规则树可使分类时间明显加快.这两方面的措施弥补了现有关联分类应用于文本分类的不足.与 3 种典型的文本分类方法比较后发现,在低维特征空间中,关联分类的性能优于 Bayes,kNN 和 SVM,因此是一种很有应用前景的文本分类方法.

当然,关联分类也存在训练时间长的不足.这是由频繁模式发现算法决定的,可以直接利用数据挖掘领域相关研究的最新成果不断改进;同时,由于分类规则可以反复使用,分类不同测试集时不必重新训练,在一定程度上缓解了训练时间长所带来的问题.

References:

[1] Lewis DD. Naive (Bayes) at forty: The independence assumption in information retrieval. In: Proc. of the 10th European Conf. on Machine Learning (ECML'98). Heidelberg: Springer-Verlag, 1998. 4−15. http://citeseer.ist.psu.edu/lewis98naive.html

[2] Yang YM, Liu X. A re-examination of text categorization methods. In: Proc. of the 22nd Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. New York: ACM Press, 1999. 42−49.

[3] Wiener E, Pedersen JO, Weigend. AS. A neural network approach to topic spotting. In: Proc. of the 4th Annual Symp. on Document Analysis and Information Retrieval (SDAIR'95). Las Vegas, 1995. 317−332. http://citeseer.ist.psu.edu/wiener95neural.html

[4] Joachims T. Text categorization with support vector Machines: Learning with Many Relevant Features. Technical Report, LS-8 Report 23, Dortmund: University of Dortmund Computer Science Department, 1998.

[5] Yang YM. An evaluation of statistical approaches to text categorization. Journal of Information Retrieval, 1999,1(1):67−88.

[6] Liu B, Hsu W, Ma YM. Integrating classification and association rule mining. In: ACM Int'l Conf. on Knowledge Discovery and Data Mining (SIGKDD'98). New York: ACM Press, 1998. 80−86.

[7] Li W, Han J, Pei J. CMAR: Accurate and efficient classification based on multiple classification rules. In: Cercone N, ed. Proc. of the 2001 IEEE Int'l Conf. on Data Mining (ICDM 2001). California: IEEE Press, 2001. 369−376.

[8] Zaïane OR, Antonie ML. Classifying text documents by associating terms with text categories. In: Zhou XF, ed. Proc. of the 13th Australasian Database Conf. (ADC 2002). Melbourne, Australian Computer Society, 2002. 215−222.

[9] Agrawal R, Srikant R. Fast algorithms for mining association rules. In: Bocca JB, Jarke M, Zaniolo C, eds. Proc. of the 1994 Int'l Conf. on Vary Large Data Bases. Santiago, 1994. 487−499.

[10] Han J, Pei J, Yin YW. Mining frequent patterns without candidate generation. Data Mining and Knowledge Discovery, 2004,8(1): 53−87.

[11] Zhou SG, Guan JH, Hu YF, Zhou AY. A Chinese document categorization system without dictionary support and segmentation processing. Journal of Computer Research and Development, 2001,38(7):839−844 (in Chinese with English abstract).

附中文参考文献:

[11] 周水庚,关佶红,胡运发,周傲英.一个无需词典支持和切词处理的中文文本分类系统.计算机研究与发展,2001,38(7):839−844.

陈晓云(1970－),女,福建晋江人,博士,副教授,主要研究领域为数据挖掘,信息检索,机器学习.

李荣陆(1976－),男,博士,主要研究领域为自然语言处理,信息检索.

陈祎(1982－),男,硕士生,主要研究领域为数据挖掘,数据库.

胡运发(1940－),男,教授,博士生导师,主要研究领域为数据工程,知识工程.

王雷(1982–),男,硕士生,主要研究领域为数据挖掘,数据库.