第32卷 第19期 计 算 机 工 程 2006年10月
Vol.32 № 19 Computer Engineering October 2006
·软件技术与数据库· 文章编号：1000—3428(2006)19—0076—03 文献标识码：A 中图分类号：TP393

基于文本分类TFIDF方法的改进与应用

张玉芳[1]，彭时名[1]，吕 佳[2]
(1. 重庆大学计算机学院，重庆 400045；2. 重庆师范大学数学与计算机科学学院，重庆 400047)

摘 要：TFIDF是文档特征权值表示常用方法。该方法简单易行，但低估了在一个类中频繁出现的词条，该词条是能够代表这个类的文本特征的，应该赋予其较高的权重。通过修改TFIDF中IDF的表达式，来增加那些在一个类中频繁出现的词条的权重，用改进的TFIDF选择特征词条、用遗传算法训练分类器来验证其有效性。该方法优于其它算法，实验表明了改进的策略是可行的。

关键词：文本分类；特征选择；TFIDF；类别区分

Improvement and Application of TFIDF Method Based on Text Classification

ZHANG Yufang[1], PENG Shiming[1], LV Jia[2]
(1. Department of Computer Science, Chongqing University, Chongqing 400045;
2. College of Mathematics and Computer Science, Chongqing Normal University, Chongqing 400047)

【Abstract】TFIDF is a kind of common methods used to measure the terms in a document. The method is easy but it undervalues these terms that frequently appear in the documents belonging to the same class, while those terms can represent the characteristic of the documents of this class, so higher weight is entrusted to them. The expression of IDF in TFIDF is modified to increase the weight of those terms mentioned, then is applied to the experiment to validate it. In the experiment, the improved TFIDF is used to select feature and genetic algorithm is used to train the classifier. The method is better than others and proves that the improved TFIDF method is feasible.

【Key words】Text classification; Feature selection; TFIDF; Class discrimination

文本自动分类的任务是：对未知类别的文档进行自动处理，判断它所属预定义类别集中一个或多个类别。随着各种电子形式的文本文档以指数级的速度增长，有效的信息检索、内容管理及信息过滤等应用变得越来越重要和困难。文本自动分类是一个有效的解决办法，已成为一项具有实用价值的关键技术。近年来，多种统计理论和机器学习方法被用来进行文本的自动分类，掀起了文本自动分类研究和应用的热潮。本文通过研究发现传统的文本特征权值表示方法TFIDF的不足：存在着低估某一个类中频繁出现的词条的缺陷。其实在一个类中频繁出现的词条恰好能够代表此类文本的特性，是应该给它们较高的权重的。本文对此进行了改进，在为每个类训练分类器的时候，用到了遗传算法的寻优策略，最后通过实验证明改进的TFIDF方法是可行的。

1 文本分类的步骤

文本分类的步骤如下：

(1)分词处理。分词技术是文本分类的基础。简单地说，就是用分词算法，把文本切分成字、词和短语。对于英文文本、单词用空格分隔，因此英文文本直接可以用空格进行切分，然而，中文句子是以连续的字符串形式出现的，词与词之间没有间隔，对于中文文本的分词，需要进行特殊的处理。目前比较常用的方法有最大匹配法(Maximum Match Method)、反向最大匹配法(Reverse Direction Maximum Matching Method)、二次扫描法和联想－回溯法[1]等。

(2)特征词条选择。因为文本一般来说比较长并且包含一些无意义的虚词，所以不能够也不必要把文本中的全部词用来分类。常常根据某种筛选策略，选出那些对分类贡献大的词条来代表文本进行分类。常用的筛选策略有TFIDF、文档频率方法、信息增益方法、互信息方法、CHI方法、期望交叉熵[2]等。借助于这些策略来选择对文本分类贡献大的特征词条。

(3)文本表示。在对文档进行分类之前，必须把文档表示成为计算机可以处理的形式。空间向量模型(SVM)[3,4]是常用的有效的方法之一。空间向量模型的主要思想是：把文本看作一个多维向量，把从文本选出来的一个特征词条当作向量的一维。

(4)分类。常用的文本分类算法有K-最近邻分类法(K_Nearest_Neighbor)[5]、朴素贝叶斯(Naïve Bayes)[6]、决策树(Decision Tree)、神经网络(Neural Net)[7]。

2 TFIDF

2.1 传统的TFIDF

TFIDF的主要思想是：如果某个词或短语在一篇文章中出现的频率TF高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适合用来分类。TFIDF实际上是：TF×IDF，TF词频(Term Frequency)，IDF反文档频率(Inverse Document Frequency)。TF表示词条t在文档d中出现的频率。

$$\mathrm{IDF}=\log\frac{N}{n}$$

作者简介：张玉芳(1965－)，女，副教授、博士，主研方向：数据挖掘；彭时名、吕 佳，硕士生

收稿日期：2005-12-25 **E-mail：**peng_sn@163.com

其中，N 为全部文档数，n 表示包含词条 t 的文档数量。

IDF的主要思想是：如果包含词条t的文档越少，也就是n越小，IDF越大，则说明词条t具有很好的类别区分能力。如果某一类C_i中包含词条t的文档数为m，而其它类包含t的文档总数为k，显然所有包含t的文档数n=m+k，当m大的时候，n也大，按照IDF公式得到的IDF的值会小，就说明该词条t类别区分能力不强。但是实际上，如果一个词条在一个类的文档中频繁出现，则说明该词条能够很好代表这个类的文本的特征，这样的词条应该给它们赋予较高的权重，并选来作为该类文本的特征词以区别与其它类文档。这就是IDF的不足之处。

2.2 改进的 TFIDF

针对IDF提出改进意见，设总的文档数为N，包含词条t的文档数为n，其中某一类C_i中包含词条t的文档数为m，则t在C_i类中为

$$IDF=\log(\frac{m}{n}\times N)$$

如果在某一类C_i中包含词条t的文档数量大，而在其它类中包含词条t的文档数量小的话，则t能够代表C_i类的文本的特征，具有很好的类别区分能力。如果除C_i类外，包含词条t的文档数为k，则公式的变形形式为

$$IDF=\log(\frac{m}{m+k})\times N$$

IDF 的值和 m，k 的关系如下：

设 $f(m)=\frac{m}{m+k}$，$m_1>m_2$

则 $f(m_1)-f(m_2)=\frac{m_1}{m_1+k}-\frac{m_2}{m_2+k}=\frac{(m_1-m_2)k}{(m_1+k)(m_2+k)}$。

因为 $m_1>0$，$m_2>0$，$k>0$，所以 $f(m_1)-f(m_2)>0$，$f(m)=\frac{m}{m+k}$是随m增大而增大的。显然，$f(k)=\frac{m}{m+k}$是随k增大而减小的。也就是说，IDF的值是随m增大而增大，随k增加而减小的。刚好能够体现改进的思想：如果某一个类C_i中包含词条t的文档数量大，而在其它类中包含词条t的文档数量小的话，则t能够代表C_i类的特征。

3 分类模型

在图 1 中，每一个类都有一个分类器，叫做类分类器，要判断一个文档的类别，每个类分类器判断此文档是否属于该分类器所代表的类。

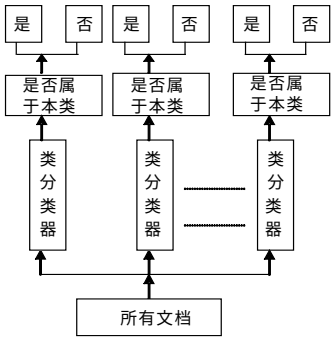

图 1 分类模型

首先在每个类的训练文本中，用遗传算法来为该类训练出一个类分类器(类文本向量)，遗传算法具有很好的寻优能力，借用遗传算法这一特点，在训练文档中训练出一个能够代表这个类的文本作为类分类器。当类分类器训练好后，要看测试文档是否属于该类，只需要比较测试文档和类分类器的相似程度。

4 分类策略

4.1 文档表示

用改进的 TFIDF 来衡量每个类中训练文档的词条的权重，并按照权重大小排序，从每个文档中选出 K 个权重最大的词条。

将从每个训练文本选出的词条除去重复项后放到一个集合 A 中，A 中的所有词条可以形成这样的字符串：$a_1,a_2,a_3,...,a_i,...,a_n$，其中n为集合A中的元素个数，$a_i$表示A中的第i个元素。

每个文档也可以表示成这样的字符串形式，如果a_i在文档中出现则a_i的对应位置用 1 来代替，否则用 0 来代替，这样文档便可以表示成：101…1…0 这样的字符串形式。人们习惯把字符串的每一位当作一个向量的维，而叫字符串为文本向量。

4.2 两向量的夹角公式

要判断两个文本的相似度，一般是判断两文本向量的相似度。向量的相似度，可以采用两向量的夹角的余弦大小来衡量。其值越大相似度越高。设两文本向量$d_i(a^i_1,a^i_2,\ldots,a^i_n)$和$d_j(a^j_1,a^j_2,\ldots,a^j_n)$则两向量夹角的余弦[8]为

$$\cos(d_i,d_j)=\frac{\sum_{k=1}^{n}(a_k^i*a_k^j)}{\sqrt{\sum_{k=1}^{n}(a_k^i)^2}\times\sqrt{\sum_{k=1}^{n}(a_k^j)^2}} \quad (1)$$

如果$\cos(d_i,d_j)>\partial$，则认为d_i和d_j相似，否则不相似，其中，∂是一个相似度阈值，也是一个经验值。

5 用遗传算法来训练每个类分类器

5.1 遗传算法

遗传算法是一种集效率与效果于一身的优化搜索方法。它利用结构化的随机信息交换技术组合群体中各个结构中最好的生存因素，从而复制出最佳代码串，并使之代代地进化，最终得到满意的优化结果[9]。运用遗传算法的这一特征，来寻求每个类的最优分类器。

5.2 遗传算法的初始种群

每个类的训练文本形成一个初始种群，在每个类的训练文本中训练一个类文本向量作为类分类器，一旦训练出了类文本向量，测试文档便根据其与类文本向量的夹角余弦大小来判断是否属于此类，当余弦大于∂，文本属于此类，否则不属于此类。

5.3 编码

设有文本d，a_i是集合 A中的词条，当a_i 在文本d中出现时，则a_i对应的$a_1,a_2,a_3,\ldots,a_i,\ldots,a_n$串的对应位上的值为 1，否则为 0，这样根据A中的词条在d出现与否，d便可表示成1010001….0101，这样的二进制编码串。串长度为n，即A的元素个数。

5.4 适应度函数

在训练类分类器的时候，采用所有类的训练文本来训练，根据类文本向量是否能够正确识别本类文本和准确区分其他类文本作为类文本向量的适应度。假设C_i类中一文本向量d，则d的适应度可以表示成：

$$Fit(d)=\frac{T_+\times F_-}{T_-\times F_+} \quad (2)$$

其中，T_+表示属于C_i并且能够用d正确分类到C_i 的文本数量，T_-表示属于C_i但是没有正确分类到C_i的文本数量，F_+表示不属于C_i但是分类到C_i的文本数量，F_-表示不属于C_i没有分类C_i的文本数量。

6 试验及其结果分析

6.1 评估指标

对于文本分类系统的性能评估测试，国际上有通用的评估指标，包括查全率(Recall)，查准率(Precision)和F_1评估值 3 项主要指标[10]。对应的公式分别如下：

(1)查全率

$$R_i = \frac{N_{cpi}}{N_{ci}} \tag{3}$$

(2)查准率

$$P_i = \frac{N_{cpi}}{N_{pi}} \tag{4}$$

(3)F1 评估值

$$F_1 = \frac{2R_iP_i}{R_i + P_i} \tag{5}$$

其中，N_{ci}是实际属于C_i类的文档数；N_{pi}是分类器预测为C_i类的文档数，N_{cpi}是分类器正确分类的文档数。

6.2 试验结果分析

试验中采用了复旦大学计算机信息与技术系国际数据库中心自然语言处理小组整理的训练和测试语料库，从中选取了 5 个类，其中训练文档 5 000 篇、测试文档 5 000 篇，每个类分别有 1 000 篇训练文档和 1 000 篇测试文档。试验中，从每个训练文本选出词条 K40，相似度阈值取 0.8。把用改进的 TFIDF 结合遗传算法分类效果和两种分类策略比较：(1)传统的 TFIDF 结合 KNN 分类效果；(2)传统的 TFIDF 结合遗传算法的分类效果。比较结果见表 1。

表 1 结构分析比较

	NewTFIDFAND Genetic			TFIDF AND KNN			TFIDF AND Genetic		
	R(%)	P(%)	F(%)	R(%)	P(%)	F(%)	R(%)	P(%)	F(%)
体育	81.37	87.37	84.26	78.64	86.57	82.41	67.72	73.63	70.55
政治	76.12	88.06	81.67	68.35	76.49	72.19	70.87	78.94	74.69
经济	65.48	79.70	70.80	53.69	79.45	64.08	57.36	70.29	63.17
农业	78.94	82.69	80.77	70.83	79.67	74.99	68.46	74.32	71.27
环境	68.95	83.86	75.68	65.29	68.95	67.07	56.48	67.38	61.69

New TFIDF AND Genetic 是指改进的 TFIDF 结合遗传算法；TFIDF AND KNN 是指 TFIDF 结合 KNN 分类算法；TFIDF AND Genetic 是指 TIDF 结合遗传算法。R、P、F 分别指查全率、查准率、F1 评估值。

从表 1 可以看出，改进的 TFIDF 结合遗传算法的分类效果，无论是查全率、查准率还是 F1 评估值都要优于传统 TFIDF 结合 KNN 或者遗传算法的分类效果。

7 结束语

本文对 IDF 进行了改进，并采用遗传算法来为每个类训练分类器，把改进 TFIDF 结合遗传算法的分类效果分别与传统 TFIDF 结合 KNN 的分类效果和传统 TFIDF 结合遗传算法的分类效果作了比较。实验结果表明，传统 TFIDF 结合 KNN 的分类效果要比传统 TFIDF 结合遗传算法的分类效果好，但是比改进 TFIDF 结合遗传算法的分类效果差，因此改进的 TFIDF 是有效的且可行的。

参考文献

1 刘 源, 谭 强. 信息处理用现代汉语分词规范及自动分词算法[M]. 北京: 清华大学出版社, 1994: 36-51.

2 Mnic D, Grobelnik M. Feature Selection for Unbalanced Class Distribution and Naïve Bayees[C]. Proceedings of the 6th International Conference on Machine Learning. Blrf: Morgan Kaufmann, 1999: 258-267.

3 Rocchio J. Relevance Feedback in Information Retrieval[C]. Proc. of SMART Retrieval System: Experiments in Automatic Doc., NJ, USA: Prentice-hall, 1971: 313-323.

4 Salton G, Wong A, Yang C. A Vector Space Model for Automatic Indexing[J]. Communications of ACM, 1975, 18(11): 613-620.

5 刘 斌, 黄铁军. 一种新的基于统计的自动文本分类方法[J]. 中文信息学报, 2002, 16(6): 18-24.

6 范 焱, 郑 诚. 用 Naïve Bayes 方法协调分类 Web 网页[J]. 软件学报, 2001, 12(9): 1386-1392.

7 梁久祯, 兰东俊. 基于先验知识的网页特征压缩与线性分类器设计[C]. 第十二届全国神经计算学术大会讨论文集. 北京: 人民邮电出版社, 2002: 494-501.

8 邹 涛, 王继成, 朱华宇. WWW 上的信息挖掘技术及实现[J]. 计算机研究与发展, 1999, 36(8): 1019-1024.

9 Rudolph G. Convergence Properties of Canonical Genetic Algorithms[J]. IEEE Trans. on Neural Networks, 1994, 5(1): 96-101.

10 Yiming Y. An Evaluation of Statistic Approaches to Text Categorization[J]. Information Retrieval, 1999, 1(1/2): 69-90.

（上接第 75 页）

生成其他的判别树T_{c3},T_{c4},…,T_{cb}的过程如图 1 所示。

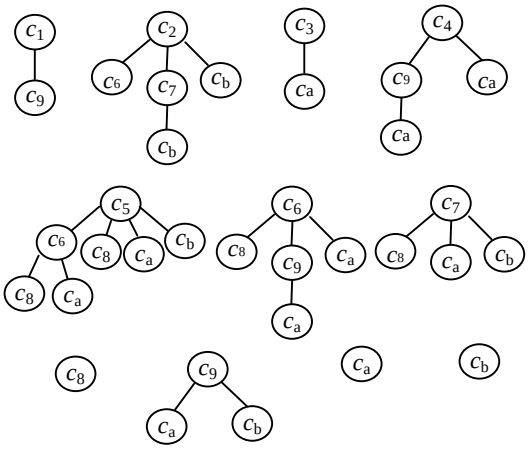

图 1 判别树 Tc1 至 Tcb

3 结束语

本文提出了一种微指令设计中互斥微操作命令的查找算法，并对其正确性进行了理论证明，然后通过实例作了进一步分析，利用该算法可编程实现互斥微操作命令的自动化查找。

参考文献

1 Hwang K. Advanced Computer Architecture: Parallelism, Scalability, Programmability[M]. 北京: 机械工业出版社, 1993.

2 Park I C, Hong S K, Kyung C M. Two Complementary Approches for Microcode Bit Optimization[J]. IEEE Transaction on Computers, 1994, 42(2): 234-239.

3 朱 霞, 高德远, 樊晓桠等. 优化微程序控制器设计[J]. 西安: 西北工业大学学报, 2003, 21(2): 176-179.

4 殷人昆, 陶永雷, 谢若阳等. 数据结构: 用面向对象方法与 C++描述[M]. 北京: 清华大学出版社, 1999.