一种快速的多模式串匹配算法及其在实时汉语文本分类系统中的应用

张鑫 程学旗 谭建龙 王映
中国科学院计算技术研究所 软件研究室 北京 100080
Email: zhangx@software.ict.ac.cn

摘 要：本文提出了一种快速的多模式串匹配算法，并且将它应用在实时汉语文本分类系统的文本向量化中。本文对比了匹配算法和传统的分词方法这两种文本向量化方法，衡量了使用这两种方法生成向量的相似度和所需时间，并且分析了产生差异的原因。实验结果说明使用多模式串匹配算法能够极大的缩短生成文本向量所需时间，并且使用向量的夹角余弦值衡量两种方法生成的向量有平均 97.4%的相似度。

关键词：字符串匹配，模式串匹配，文本分类，向量空间

A Fast Multi-pattern Matching Algorithm and Its Application In Real-time Chinese Text Classifying System

Zhang Xin Cheng Xueqi Tan Jianlong Wang Ying
Software Division, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080
Email: zhangx@software.ict.ac.cn

Abstract: We present a fast multi-pattern matching algorithm. At the same time, we use this algorithm to create text vector in real-time Chinese text classifying system. To compare the new vector-creating method against the conventional way based on word partition, we measured the vector similarity and processing time in both methods, and analyzed the reasons of difference in the creating vectors. The results show that we can greatly reduce the time for creating text vectors by using the multi-pattern matching algorithm, and the vectors created in both methods are 97.4% similar in average, measured by cosine value of vector angular.

Keywords: string match, pattern match, text classifying, vector space

1 引言

制约实时文本分类系统的应用的关键因素是实时文本分类系统的处理能力。大多数的文

本分类系统都选择使用“词”作为特征项，构成分类器的特征空间或概率空间。而由于汉语文本的书写表示特殊性，文本分类系统对汉语进行分类前，必须要对汉语文本进行分词处理，再根据训练产生的特征集进行筛选，将文本从有序的符号序列转化成无序的〈特征项，出现次数〉二元组的集合。这也就是文本的向量化处理过程。同分类器对文本向量进行分类的过程相比，根据处理文本的长度和所选分类器的不同，文本向量化过程所消耗的计算时间是分类过程的数倍甚至上百倍不等。而在实时汉语文本分类系统中，由于需要分类的文本都是原始文本，对文本的分类所需时间是文本向量化过程和分类计算时间的总合，所以提高文本向量化过程的处理速度是提高实时文本分类系统的处理能力的决定因素。

多模式串匹配算法处理问题是根据建立在一个确定的“符号集合”Σ上的“模式串集合”P 以及“数据符号序列”T，找出模式串集合 P 中每一个模式串在数据符号序列中的所有出现位置。目前主要应用在实时信息检索系统和网络入侵检测系统，对大量时效非索引信息进行快速检索，或在用户的工作事件序列或网络传输信息的数据序列中检测可能出现的入侵行为或可疑信息的关键词。

我们发现汉语文本分类系统对分类文本进行向量化的过程中，构成向量空间的特征都是由训练过程选择的实义名词或动词。这些词语同英文词语相比具有绝对的“稳定性”，即词语不会随着所在语句的位置，时态、语态等语法因素而又任何形式上的变化。而多模式串匹配算法能过快速准确的统计模式串出现的次数(或所有出现位置)。因此，我们在实时汉语分类系统中使用多模式串匹配算法来进行文本向量化的工作，取代原来的分词加选择的向量化工作。在我们的实验中，新的向量化方法的处理时间是原方法的千分之一甚至更少。

本文的后续部分安排如下：第二章给出我们在 WuManber 多模式串匹配算法的基础上改进算法；第三章描述了我们做的文本向量化对比实验的过程和结果，并分析了两种方法产生向量的差异及原因；最后提出我们的实验结论和后续工作的安排。

2 多模式串匹配算法

2.1 多模式串匹配算法处理问题和现状

多模式串匹配算法处理问题可描述如下：

已知条件：

*符号集合 Σ

*模式符号串集合 $P=\{p_i\}$ ， $p_i= p_i^{(1)}.. p_i^{(m_i)}$ $p_i^{(x)}\in \Sigma$

*数据符号串 $T=t^{(1)}.. t^{(n)}$ $t^{(x)} \in \Sigma$

求解问题：

出现位置集合$\{o_i\}$，使得 $\forall j\ t^{(oi+x)} = p_j^{(x)}$ $x=1.. m_j$

目前常用的多模式串匹配算法有 Aho Corasick 算法[2]，Wu Manber[3] 算法和 BOM（Backward Oracle Matching）算法[4]。其中 Wu Manber 算法在符号集合的字符数大于 8，模式串长度小于 20 个字符的情况下(自然语言文本)，具有最快的处理速度[6]。

Wu-Manber[3]算法是 Boyer-Moore[1]算法处理多模式串问题的派生形式，是一种快速实用的多模式串匹配算法。它采用了 Boyer-Moore 算法的框架，使用块字符(**block character**)来计算的 bad-character shift 距离表。此外，在进行匹配的时候，它使用散列表选择模式串集合中的一个子集与当前文本进行匹配，减少无谓的匹配运算。Wu-Manber 方法的执行时间不会随

着模式串集的增加而成比例增长，而且要远少于使用每一个模式串和 Boyer-Moore 算法对文本进行匹配的时间总和。Wu-Manber[2]算法的时间复杂度在最好的情况能达到 O(B*n/m)。(B 是块字符的长度，是算法在每一个入口点计算块字符的时间)

2.2 改进的 Wu Manber 算法

我们发现 Wu Manber 算法的优越之处在于它能够过滤掉不可能的匹配的入口点。因为进行模式串匹配工作是固定的和最为耗时的，匹配入口点过多，进行模式串匹配的次数越多，算法消耗的时间就越长。我们对 Wu-Manber 算法进行改进的意图就是放大这种优势，主要采用了两种方法：计算精确的不良字符转移距离和引入弱化的良好后缀转移距离。

2.2.1 精确的不良字符转移

Wu-Manber 算法在选择匹配入口点的问题中使用的 Shift[]表记录了基于块字符的不良字符移动距离。计算公式如下：

$$bad\_character\_shift(Bc)=\begin{cases} m-B+1 & Bc \notin substr(pattern) \\ \min\{m-B-k \mid \forall j\ pattern_j[k+i]=Bc[i]\ (0\le i<B)\} & others \end{cases}$$

我们引入了精确的不良字符转移表 Shift[]计算方法。

$$bad\_character\_shift(Bc)=\min\{length_j-B-occurence\_pos(Bc,pattern_j)\mid \forall j\}$$

$$occurence\_pos(Bc,pattern)=\max\{pos \mid \underset{0\le i<B}{\wedge}((pos+i<0)\vee(pattern[pos+i]=Bc[i]))$$

$$(-B\le pos\le length-B)\ \}$$

新的不良字符转移函数的值域 0≤y≤m (m 是模式串的最小长度)，而不是原来的 0≤y≤m-B+1。

2.2.2 弱化的良好后缀转移

Wu-Manber 算法中没有使用 Boyer-Moore[1]算法中使用的良好后缀移动方法，因此它在进入匹配计算模块后，无论匹配的结果如何，下一个匹配开始位置都固定的向右前进一位。因此我们引入了一种弱化了的良好后缀转移方法。

Wu-Manber 算法进入匹配阶段的充要条件是当前匹配入口点的块字符对应的 Bad-character Shift 偏移值为 0，即当前的块字符恰好是某个(组)模式串的最后一个块字符。所以无论匹配结果如何，块字符都是固定出现的。因此我们引入 GBSShift[]表，该表纪录了每一个模式串的长度为 B 的后缀(最后一个块字符)在所有模式串中的所有非后缀出现位置与相应模式串词尾的距离的最小值。

$$good\_B\_suffix\_shift(Bc)=\min\{length_j-B-occurence\_pos(Bc,pattern_j)\mid \forall j\}$$

$$occurence\_pos(Bc,pattern)=\max\{pos \mid \underset{0\le i<B}{\wedge}((pos+i<0)\vee(pattern[pos+i]=Bc[i]))$$

$$(-B\le pos<length-B)\ \}$$

2.3 算法性能的对比实验

为了验证改进算法的性能，我们分别使用 Aho Corasick，BOM,Wu Manber 和 ImprovedWM[7] 四种算法在中英文语料上进行了对比实验。实验中采用的英文文本数据集与 SumKim1999[5]采用了相同的数据集：Gutenberg 项目中的 King James Bible(4,845,164 bytes)，实验中使用的中文文本数据使用了人民日报分类语料[8]中的一部分(5,242,880)。我们的实验平台是一台奔腾 IV 1.4GHz 处理器，384M SDRAM 内存的个人电脑。实验中我们使用的英文模式串集合是 1000 个从英文词典中随机抽取的最小长度为 4 的英文单词，中文模式串集合是

1000 个构成向量特征空间的汉语单词。

	Aho Corasick		BOM		Wu Manber		Improved WM	
语料类别	处理时间	处理速度	处理时间	处理速度	处理时间	处理速度	处理时间	处理速度
英文	0.163	28.31	0.1175	39.33	0.1135	40.71	0.0771	59.91
中文	0.254	19.68	0.2353	21.25	0.1355	36.89	0.1021	48.96

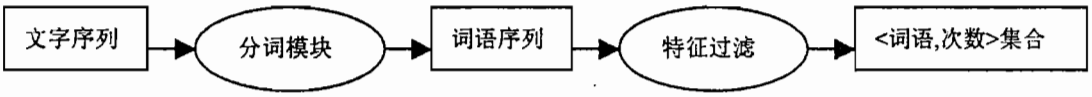
实验中的处理时间是我们重复 30 次匹配过程的取得的处理时间的平均值，单位是秒，处理速度的单位时兆字节/秒。

3 实时文本向量化

3.1 对比实验

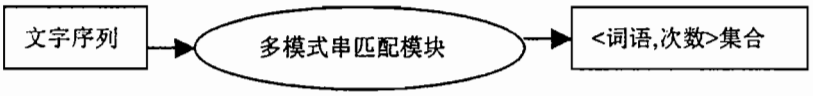
我们将多模式串匹配算法应用到实时文本分类系统的文本向量化过程中，并和传统的方法进行了对比实验。实验中我们使用的语料是人民日报分类语料[8]，共含有 47 个分类。在训练阶段，文本数据分词后共产生 45884 个词语。我们使用互信息量作为特征提取的标准，为每一个分类选定了互信息量值最大的 1000 个词语作为该分类的特征。

传统的文本向量化方法处理流程如下[8][9][10]：



实验中我们使用的分词模块是中科院计算所张华平博士设计的概率统计词典 ICTCLAS.DLL，特征过滤阶段是使用二分法检索特征模式集，检查当前词语是否属于特征项。

模式串匹配文本向量化方法处理流程如下：



实验中使用的多模式串匹配模块是我们的前文说明的改进算法。

我们使用向量的夹角余弦值比较了两种方法生成向量的相似度。取每个分类集合内所有的向量相似度的平均值作为对应分类内生成向量的相似度。下表给出了相似度最高的和相似度最低的 10 个分类的相似度，以及两种向量化方法使用时间。

相似度最高的 10 个分类情况概述如下：

分类名称	文件数目	文件总长度	平均相似度	传统方法 t1	匹配方法 t2
C21-CheIndustry	23	33,846	99.2%	0.662	0.00059
C-Biology	9	7,747	99.1%	0.16	0.0001
C24-nMetallurgy	12	25,876	99.0%	0.511	0.00044
C-Law	88	243,299	98.9%	6.067	0.004237
C10-nChemistry	19	33,539	98.8%	0.702	0.00057
C15-Energy	40	60,958	98.7%	1.162	0.0001471
C16-Electrony	33	59,172	98.7%	1.22	0.001154
C-New6	44	128,224	98.7%	2.533	0.003154
C28-Material	10	12,589	98.6%	0.26	0.00019
C19-Computer	42	84,676	98.6%	1.66	0.001702

相似度最低的 10 个分类情况概述如下：

分类名称	文件数目	文件总长度	平均相似度	传统方法 t1	匹配方法 t2
C-Sport	202	224，789	94.5%	4.047	0.013754
C5-Education	89	25,747	95.2%	5.317	0.014366
C-Other	33	67,813	96.1%	1.343	0.001261
C3-Art	98	193,978	96.3%	3.837	0.002894
C7-History	60	171,664	96.5%	3.856	0.002813
C-New2	181	357,058	96.7%	6.728	0.007502
C-Medical	58	61,304	96.8%	1.121	0.001162
C4-Literate	41	146,043	97.0%	3.022	0.012475
C1-Language	45	9,573	97.1%	1.983	0.001733
C29-Transport	77	121,002	97.1%	2.342	0.002153

表中文件长度的单位为字节（byte），两种方法所需时间的单位为秒，两种方法所需数据都已载入至内存中，所计时间不包含文件读取时间。传统方法所需时间通过标准 C++中的 clock()函数测得，匹配算法所需时间是重复 1000 次匹配过程时间的平均值。

3.2 向量化差异分析

我们分析了两种向量化方法产生向量存在差异的因素，主要有特征词包含和分词误差。

特征词包含是指在选定的特征词集合中，有一些特征词包含另一个(些)特征词。比如：在 C12-Earth 分类中，地质、地质学、地质学家作为第 33、15、23 个特征词被选中。由于多模式串匹配算法会准确的找出每个模式串的每次出现，所以对于“地质学家”这个序列，如果将地质、地质学、地质学家作为模式串，则模式串匹配算法会作出地质、地质学、地质学家各出现一次的判断，而分词模块只会作出地质出现一次的判断。这种情况可根据特征词集合预先计算出一个规则集合。如：

地质出现次数 = 地质匹配次数 – 地质学匹配次数；

地质学出现次数 = 地质学匹配次数 – 地质学家匹配次数；

分词误差是指分词过程中单位不一致或分词错误。我们认为在这种情况下，向量差异为有益差异，能更好的代表文本。比如在 C-Sport 分类 A114.txt 中两句关于苏联队的分词结果：

“苏联队 夺得 世界 冰球 A 组 锦标赛 冠军”，

“新华社 日内瓦 5月 2日 电 （ 记者 施 光耀 ） 苏 联队 今天 在 瑞士

首都 伯尔尼 以 5 ： 0 战胜 最后 一个 对手”。

同一个词语“苏联队”在文中的不同出现分别被分成“苏联队”和“苏” “ 联队”，在这种情况下，有匹配算法生成的向量更能分映文本的真实内容。

4 结论和展望

我们利用多模式串匹配算法来代替传统的文本向量化方法，作为实时汉语文本分类系统中产生向量的方法，极大的缩减了文本向量化过程所需的时间，从而提高了文本分类系统的处理能力。

我们的后续工作会致力于根据英文单词的构词规律，将多模式串匹配算法应用在英文文本的向量化过程中。

参 考 文 献

[1] Robert Stephen Boyer, J Strother Moore: "A fast string searching algorithm", Communications of the ACM20, 1977. 762-772

[2] A.V.Aho M.J.Corasick: "Efficient string matching: an aid to bibliographic search", Communications of the ACM, 1975

[3] Sun Wu,Udi Manber: "A Fast Algorithm For Multi-pattern Searching", The Computer Science Department The University of Arizona, 1994

[4] C.Allauzen M.Raffinot: "Factor oracle of a set of words", Technical report 99-11, Institute Gaspard Monge, University de Marne-la-vallee,1999

[5] Sum Kim: "A Fast Multiple String-Pattern Matching Algorithm", San Diego CA : 17th AoM/IAoM International Conference on Computer Science,August,1999

[6] Gonzalo Navarro, Mathieu Raffinot: "Flexible Pattern Matching In Strings", 2002 CAMBRIDGE PRESS.

[7] 张鑫 谭建龙 程学旗: "一种改进的 Wu-Manber 多关键词匹配算法", 《计算机应用》, 2003

[8] 卜东波: "聚类/分类理论研究及其在文本挖掘中的应用", 中科院计算所博士论文, 2001

[9] 易靖: "基于信息粒度原理的文本分类方法的研究", 北京工业大学硕士论文, 2001

[10] 庞剑峰: "基于向量空间模型的自反馈的文本分类系统的研究与实现", 中科院计算所硕士论文, 2001