基于专有名词优先的快速中文分词

梁卓明，陈炬桦
（中山大学 信息科学学院 计算机系，广东 广州 510275）

摘 要：中文分词是中文信息处理系统中的一个重要部分。主题信息检索系统对分词的速度和准确率有特殊的要求。文中回答了词库建立的词条来源和存储结构两大问题，提出了一种基于专有名词优先的快速中文分词方法：利用首字哈希、按字数分层存储、二分查找的机制，通过优先切分专有名词，将句子切分成碎片，再对碎片进行正反两次机械切分，最后通过快速有效的评价函数选出最佳结果并作调整。实验证明，该分词方法对主题信息文献的分词速度达92万字每秒，准确率为96%，表明该分词方法在主题信息文献的分词处理中具有较高性能。

关键词：中文分词；专有名词；词典机制
中图分类号：TP391 **文献标识码**：A **文章编号**：1673-629X(2008)03-0024-04

A Rapid Chinese Word Segmentation Method Based on Priority Special Names

LIANG Zhuo-ming, CHEN Ju-hua
(Dept. of Computer Sci., Sch. of Info. Sci., Zhongshan Univ., Guangzhou 510275, China)

Abstract: Chinese word segmentation is a key component of Chinese information processing systems. The topic information retrieval system has special requirement for both speed and veracity. Answer two important questions for building dictionary: how to get word items and how to organize them, and design a rapid Chinese word segmentation algorithm based on dictionary based on special name. Use "first character Hash, store the items according to the word length, and binary search mechanism, cut the sentences by special name, then bi-direction maximum match to segment the rest, use an easy but effective scoring function to select the best, and adjust at last. The experiment result shows this segmentation method can reach a speed of 920 000 words per second, and the correctness rate can reach 96%, which proves that this method has high efficiency.

Key words: Chinese word segmentation; special name; dictionary mechanism

0 引 言

信息检索系统一般以词为单位进行索引或查询。词是最小的、有意义的、能独立活动的语言成分[1]。英语的文本是在小字符集上的已经以空格分隔开的词串，汉语的文本则是在大字符集上的连续字串，词与词之间并没有明显的分隔标记。因此，在中文信息处理系统中，中文分词是必不可少的环节。

中文分词，也称切词(Segmentation)，就是把中文的汉字序列切分成有意义的词。目前分词算法非常多，大致可以归纳为三大类：基于字符串匹配的分词方法、基于理解的分词方法、基于统计的分词方法。

基于字符串匹配的分词方法，又叫机械分词法[2]，是按照一定的策略将待切分的字串与一个词典中的最大的词条进行匹配，若在词典中找到某个字符串，则匹配成功，进行切分，否则不予切分。机械分词法的优点是速度快，易于维护更新，缺点是不能进行智能的歧义识别和新词识别。

基于理解的分词方法，又叫基于语法和规则的分词法[3]，或者叫专家系统分词法，将自动分词过程看作知识推理过程，通过计算机模拟人类对句子的结构的理解，进行词的识别。该方法需要考虑知识表示、知识库的逻辑结构和知识库的维护，在实际应用中工作量巨大，因此目前尚处在实验阶段。

基于统计的分词方法，又叫无词典分词法[4,5]，根据字符串在语料库中出现的统计频率来决定其是否构成词。优点是比较有效地实现上下文识别生词、自动消除歧义。缺点是时空开销大，训练时间长，对长词识别能力差，不能根本消歧。

每种分词方法都需要以一种对应的形式将汉语词

收稿日期：2007-06-21
作者简介：梁卓明(1982-)，男，硕士研究生，研究方向为搜索引擎、中文分词、聚类检索；陈炬桦，博士，副教授，研究方向为并行计算、信息管理系统。

语知识组织一个字库或词库。词库的组织形式直接影响该分词方法的最终效果。文献[6]讨论了三种典型的分词词典机制:基于整词二分的词典机制、基于TRIE索引树的词典机制、基于逐字二分的词典机制。实验表明,基于逐字二分的词典机制是一种简洁高效的词典组织形式。

中文分词算法有两个关键指标:速度和准确率。不同分词算法有不同的侧重点。根据实际应用采用合适的算法十分重要。在开发手机上的WAP主题信息搜索系统时,使用了Lucene作为检索子系统。开源的全文检索工具Lucene被广泛嵌入到各种应用中实现针对应用的信息索引、检索功能。由于Lucene对中文支持不足,需要开发一个针对主题检索和Lucene的中文分词模块,要求它在较高准确率下具有尽可能快的分词速度。手机上的汉字输入比较麻烦,人们倾向于输入较短的带明显主题导向的查询字符串,例如"广州天河湘菜",甚至是"天河湘菜"。另外,主题信息文献里的专有名词出现频率高。因此,文中提出了一种基于专有名词优先的快速中文分词方法,旨在兼顾高速度和高准确率,提高Lucene的索引效率。

1 基于专有名词优先的快速中文分词

1.1 基本思想

根据专有名词词典,优先在待分析字符串中识别和切分出特殊的专有名词,以这些词作为断点标记,将原字符串分为较小的串或称为碎片;再根据通用词典,对碎片进行正向反向的两次快速机械分词,使用评价函数选择最优结果,最后根据一些基本语法进行调整,从而在利用机械分词速度快的优势上,进一步提高切分的准确率。

1.2 词库组织

词库的组织要解决两个问题:词条来源和存储结构。本算法所用的词库由四个部分组成:人名词库、地名词库、特殊名词词库和通用词库。由前三个词库组成一个专有名词库,独立于通用词库。

对于词条的来源问题,笔者从中外名人录中提取出一万多个的名字,组成人名词库,一些新兴的热门人名,如"李宇春",可以通过对查询日志的分析,识别出来,再把它添加到人名词库中。地名词库由中国和世界各地的城市名称和主要道路名称组成。笔者发现,道路名称在不同城市的重复出现,如"人民路"、"中山路"、"环市路"。因此笔者搜集了具有代表性的北京、天津、上海、成都、广州、佛山六个城市的所有道路名、桥名和景点名称。特殊名词词库,是指与日常生活、吃喝玩乐相关的词汇,从阿里巴巴等电子商务网站上提取。通用词库从中科院语言研究所的ICTCLAS[7]分词系统中的词库提取出七万多条后,再添加部分从训练语料中发现的词,最后过滤出存在于专有名词词库中的词条。

考虑到如何有效地组织词库的问题时,笔者对基于逐字二分的词典机制的作了一些改进,采用"首字哈希+按字符数分层存储"的方式组织词库。词库分两部分索引信息表、词条表。索引信息存储词条首字的索引信息,通过它来定位以它为首字的词在词条表中的位置。该词库组织方式的数据结构如下:

```
struct IndexChSP{
    int start2;int end2; //2字词
    int start3;int end3; //3字词
    int start4;int end4; //4字词
    int start5;int end5; //5字词
    int start6;int end6; //6字词
    int start7;int end7; //7字词
};//索引专有名词词典
```

其中,start2和end2分别表示以该字为首字的2字词在2字词库中的存储起止位置。

常用的汉字均是GB2312,以GB2312汉字为首字成词的机会也远远大于非GB2312字,所以首先将词库划分出GB2312词和数量极少的非GB2312词,再对GB2312词按字数分为二字词库、三字词库、四字词库、五字词库、六字词库、七字词库。词库中不存储单字词。专有名词库中总共有35 948个词,99 446个汉字,平均每个词含有2.76个汉字。通用词库词条最长为四字。通用词库中总共有108 616个词,292 806个汉字,,平均每个词含有2.70个汉字。

查询词典过程如图1,对首字使用哈希函数,求出对应的GB2312值,如果在0到6 767的范围内(GB2312共有6 768个汉字),则判断为GB2312词,以它的GB2312值作为下标找到它的索引信息表,再通过索引位置信息,在词条表中进行二分查找,最终确定查询词条在词库中是否存在。

对首字使用哈希函数,利用了汉字在GB2312汉字编码表中的对应关系:每个汉字的范围从0xA1 - 0xF7,共87种,第二个字节的范围从0xA1 - 0xFE,共94种,利用这两个字节共可定义87 * 94 = 8178个汉字,实际共有6763个GB2312码的汉字。这样做的好处是,索引字共6763个,远远小于词典中词的条目,在匹配的过程中只要查找以索引字建立的词典,效率得到数量级的提高。首字哈希函数计算公式如下,其中,Index表示某汉字在编码中的位置,c1、c2代表汉字的内码。

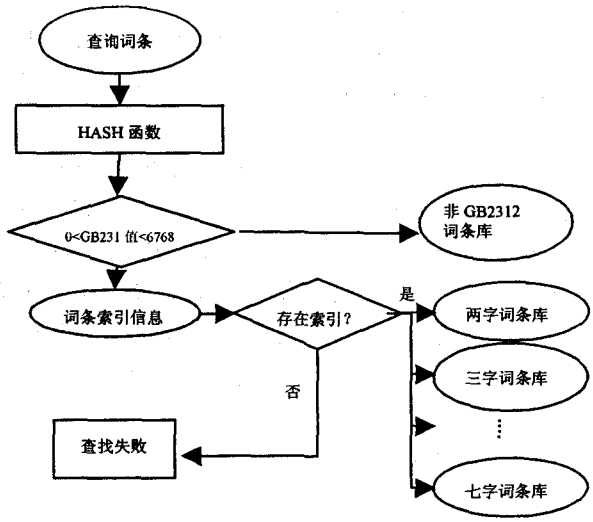

图 1 词典查询流程图

$$Index_{clc2} = (c1 - 0 \times B0) \times 94 + (c2 - 0 \times A1) \quad (1)$$

最后,将词库里的词条按其内码排序,然后直接将词库以硬代码的形式写入程序中,省去了动态装载字典的过程,整个词库以全局变量的形式存在,在分词和查字典的过程中,免去大量的 IO 操作,大大提高了运行效率。最终编译出来的组件文件也只有 1.2M 的大小。当字典需要更改时,只需要重新编译分词程序即可,编译时间不足 2 秒。

1.3 算法描述

分词过程如图 2 所示。

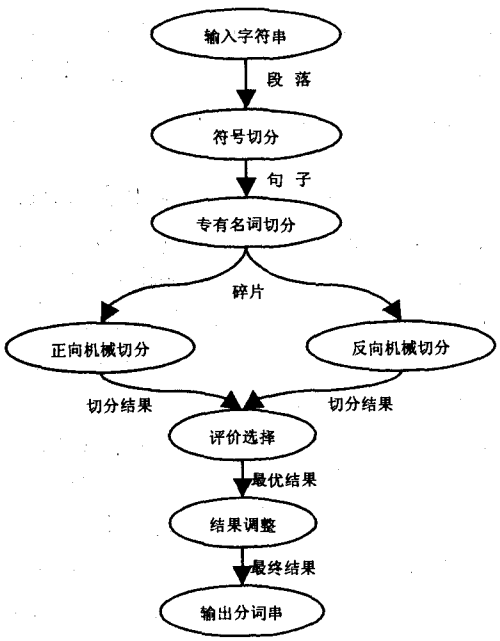

图 2 分词处理流程图

第一步,对输入的字符串进行符号切分,将英文单词、数字串、半角和全角的标点串作为切分标记,剔除掉没意义的其它符号,剩下中日韩三国文字。如“使用 Java 语言”,切分后变成“使用”+“Java”+“语言”三个字串,中间的英文字串直接进入结果,剩下的字串进入下一步的处理。

第二步,对提交的句子进行专有名词切分,即使用专有名词词库,对句子进行最大匹配的字长为七个字的快速正向最大匹配切分,提取出句子中的专有名词,留下碎片进入下一步的双向机械切分。

第三步,对碎片进行双向机械切分,即利用通用词库,对碎片进行正向和反向两次的机械切分,记录下切分的位置、切分的词条数和词条字长。从表 1 看出,最大匹配分词方法由于其机械性,是速度最快的分词方法。但最大匹配分词方法不能发现交叉切分歧义。解决这个问题的方法是再进行一次逆向最大匹配分词,两者结合起来发现交叉歧义,利用下面的评价函数消歧。

第四步,对双向切分结果进行评价选择,公式描述如下:给定一个分词原子序列 S,S 的某个可能的分词结果记为 $W = (w1, w2, \cdots, wn)$,记

$$T = \sum_{i=1}^{i=n} \text{length}(w_i)^2 \quad (2)$$

在 $W1, W2, \cdots, Wk$ 中,对应的词数为 $C1, C2, \cdots, Ck$,记 W' 为最终切分结果,使得此时 $C' = \text{Max}(C1, C2, \cdots, Ck)$,并且,$T' = \text{Max}(Ti, Tj, \cdots, Tq)$,$Ci = Cj = \cdots = Cq$。即以词条数最少优先,采用字长大于 2 的字符串的总数最少的切分结果,如果切分词条数相等,则以短词优先,采用各词条的字长的平方之和最小的切分结果。

第五步,查看选择的切分结果中相邻的单字串的最长组合,如果全部属于外国译名字符集,则判断其为一个外国译名的专有名词。

1.4 算法分析

1.4.1 空间复杂度

(1)词表:GB2312 词中由于使用首字哈希,不存储词的首字,实际存储的汉字数为 63 513(专有名词库)+184 214(通用词库)=247 727 个,共存储了 144 564 个词,占用空间为 2 * 247 727=495 454 字节。

(2)索引表:专有名词库中共占空间 4 * (7 - 1) * 2 * 6768 =324 864 字节;通用词库共占空间 4 * (4 - 1) * 2 * 6768=81 216 字节,索引表总占空间 324 864 +81 216=406 080 字节。

所以分词系统运行时只占用 495 454 + 406 080 = 901 534 字节,即约 880kB。再加上待切分字串和其它一些参数,只执行一个分词任务时,占用内存不足 900kB。

1.4.2 时间复杂度

本算法对输入的待分词字符串实际扫描了四次,

由于查字典的时间可看作为常数时间 log(N),N 为每个字为首字成词的平均词数,平均切分时间为 4log(N),所以总的算法复杂度为 $O(n)$。

2 实验分析

用纯C语言实现该算法与反向最大匹配切分算法,并下载了中科院免费版的 ICTCLAS,将三种算法的函数封装成同样接口的 DLL 文件,在 Java 中调用测试,测试机器为 AMD2600 + /1024M PC,首先进行速度测试,对从《人民日报》上随机摘取的 55 195kB 语料进行切分速度测试,测试结果如表1。接着,对国内最权威的著名美食网站 Poco 上随机挑选一百篇文章进行切分准确性测试,平均文章长度为 1 000 汉字,测试结果如表2。

表1 切分速度对比测试

算法	反向最大匹配切分	ICTCLAS	专有名词优先切分
总耗费时间	18.322s	1839.926s	29.932s
平均切分速度	3 012 kB/s	30kB/s	1844 kB/s
平均每秒切分字数	1 500 000 字/s	15 000 字/s	922 000 字/s

表2 切分准确率对比测试

算法	反向最大匹配切分	ICTCLAS	专有名词优先切分
准确率	82%	88%	96%

最后,对 152 930 条数据进行索引测试,测试了使用本分词组件前后索引文件大小的变化,并抽取了十个词进行检索测试,得出前后不同的平均检索时间,测试结果如表3。

表3 Lucene 索引检索测试

项目	不使用分词组件	使用本分词组件
索引文件大小	26 260kB	19 510kB
检索速度	0.10s	0.07s

从上面的实验数据证明,本分词算法的速度极快,准确率较高,适应对时间响应快应用要求。与 Lucene 配合使用后,能使 Lucene 的索引文件体积减少 30%,检索速度提高 20%,显著提高 Lucene 的索引检索效率。另外,本组件运行时占用内存少,支持多用户同时使用,满足基于 Lucene 的搜索引擎的应用需要。

3 结束语

该分词方法的主要优势在于速度极快,能达到 92 万字每秒的切分速度。对于商旅信息分词的准确率可达到 96%。不足之处在于它不能进行自动的新词识别。解决的办法是,根据实际需要,人工录入,完善词库,这样同时可有效控制词库的大小。本方法也不能进行真正的歧义识别,真正的歧义识别必须根据中文的语法和词性分析,这样一难以覆盖全部语法,二耗费过多时间和系统资源,得不偿失,有待进一步研究突破。因此,本算法有效满足主题信息处理的应用要求。

参考文献:

[1] 朱德熙.语法讲义[M].北京:商务印书馆,1982.

[2] 孙茂松,邹嘉彦. 汉语自动分词中的若干理论问题[J]. 语言文字应用,1995, 4(4):40-47.

[3] Palmer D. A trainable rule-based algorithm for word segmentation[C]// The 35th Annual Meeting of the Association for Computational Linguistics (ACL'97). Madrid:[s.n.],1997.

[4] Choi A, Cheng C H, Ko Y L. Word extraction from Chinese documents by occurrence counts[C]//1988 International Conference on Computer Processing of Chinese and Oriental Languages. Toronto, Canada:[s.n.],1988:488-491.

[5] Fan C K, Tsai W H. Automatic word identification in Chinese sentences by the relaxation technique[J]. Computer Processing of Chinese and Oritental Languages, 1988,4(1):33-56.

[6] 孙茂松, 左正平,黄昌宁.汉语自动分词词典机制的实验研究[J]. 中文信息学报,2000,14(1):1-6.

[7] 刘 群,张华平,俞鸿魁,等. 基于层叠隐马模型的汉语词法分析[J].计算机研究与发展,2004,41(8):1421-1429.

(上接第23页)

性,以便识别更广泛,可用性更好。

参考文献:

[1] Bezdek J C, Pal S K. Fuzzy Models for Pattern Recognition [M]. [s.l.]: IEEE Press, 1992.

[2] Kimura T D, Apte A, Van Vo. Recognizing Multistroke Geometric Shapes: An Experimental Evaluation[C]//In Proceedings of the ACM Conference on User Interface and Software Technology(UIST'93). Atlanta, GA: ACM Press, 1993: 121-128.

[3] Ulgen A F F, Akamatsu N. Geometric shape recognition with fuzzy filtered input to a backpropagation neural network[J]. IEEE Trans. Inf. Syst., 1995, E788-D(2): 174-183.

[4] O'Rourke J. Computational geometry in C[M]. 2nd edition. Cambridge: Cambridge University Press, 1998.

[5] Freeman H, Shapira R. Determining the minimum-area encasing rectangle for an arbitrary closed curve[J]. Communications of the ACM, 1975, 18(7): 409-413.

[6] Fonseca M J, Jorge J A. Using Fuzzy Logic to Recognize Geometric Shapes Interactively[C]//Proceedings of the 9th Int. Conference on Fuzzy Systems(FUZZ-IEEE)2000. San Antonio, USA:[s.n.],2000.