文章编号:1001 - 9081(2005)07 - 1716 - 03

汉字型姓名转换为首音码的重码智能处理

涂金德,李永平
(温州职业技术学院 计算机系,浙江 温州 325035)
(jdebox@126.com)

摘 要:在 GBK 汉字库范围内,通过分析姓氏汉字的读音和多音字用作名字时的习惯读音,去掉了在姓名中很少使用的读音,从而大幅度降低了首音重码;然后对仍有重码的汉字进行智能处理,进一步减少了重码选择;最后生成了首音转换码表,并设计转换算法,能够实现首音码的高效自动转换,使重码选择率从 14.4% 降为 3.7%。

关键词:首音码;重码;姓名;汉字;智能
中图分类号: TP391.1 **文献标识码**:A

Intelligent treatment of converting Chinese name to the first letter of spelling

TU Jin-de, LI Yong-ping
(*Computer Department, Wenzhou Vocational and Technical College, Wenzhou Zhejiang* 325035, *China*)

Abstract: The pronunciation of family name and the conventional pronunciation of the polyphone in GBK standard Chinese library were analysed. Then these spellings not common use in name were removed, that reduced the coincident codes greately. The next, the left coincident codes were treated intelligently, which could reduce the operation of selecting coincident codes further. At last, converting code table of the first letter of spelling was created and the converting algorithm was designed. The code table and algorithm can convert efficiently the first letter of spelling and the rate of coincident code reduces from 14.4 percent to 3.7 percent.

Key words: the first letter of spelling; coincident code; name; Chinese characters; intelligence

0 引言

首音码(汉字拼音首字母)作为特定汉字信息的输入或查询已经在各个领域得到广泛应用,如在医院管理信息系统中使用药名的首音码来输入药方。输入特定汉字信息后自动转换为首音码的技术也得到普遍使用,如图书馆管理系统在图书登记时自动生成作者姓名、书名的首音码。由于汉字存在大量多音字,使得转换时需要频繁进行首音重码的人工选择,降低了转换速度,影响相关信息的快速录入。因此,降低首音重码和减少首音码转换的人工干预显得十分必要。

目前关于首音码应用的研究较多[1,2],但是,有关首音重码问题的研究甚少,未见有降低首音重码研究的公开发表文献。

本文主要对姓名的首音码转换技术进行研究,以大幅度降低其首音重码。针对目前普遍使用的 GBK 汉字库,对汉字用作姓名时的习惯读音进行全面分析,对首音重码的汉字进行智能处理,生成首音转换码表,并设计相应的转换算法。这将在涉及姓名首音码自动转换的各个领域中具有很大的应用价值,如在手机电话薄中可利用姓名的首音码方便地实现电话号码的快速查找。

1 GBK 汉字拼音表的获取

Windows 2000/xp 自带的全拼输入法码表文件(WINPY.MB)含有全部 GBK 汉字及其拼音。该码表中包含的多音字数目很大(共有 5543 个),即使去掉首音相同的汉字,首音不同的汉字还有 4569 个。经分析,该码表中许多读音很罕见,而且大量读音在用作姓名的汉字中是不使用的。本文将 GBK 汉字分为两部分:GB 汉字(属于 GB 2312-80 的汉字)和不属于 GB 的汉字(这里称为扩充汉字)。GB 汉字及拼音使用 UCDOS7.0 自带的拼音输入法中的拼音码表,该码表中的拼音更为常用,而且能够满足下面分析的要求。扩充汉字及拼音就来自码表 WINPY.MB。

使用 Delphi 语言编程处理生成两个 Paradox 类型的数据表,即 GB 汉字拼音表 GBPY.DB 和扩充汉字拼音表 KCPY.DB。这两个数据表的结构如表 1(除字段 SYM 外)所示,每个汉字占用一条记录。表 GBPY.DB 中有 6 763 个汉字,表 KCPY.DB 中有 14 139 个汉字,两表共有 20 902 个汉字,这与 GBK 标准[3]中规定的汉字总数相符。

表 1 数据表结构

字段名	类型	宽度	说明
BH	整型		编号:存放每个汉字的顺序号
HZ	字符串	2	汉字:存放单个汉字
PY	字符串	40	拼音:存放对应汉字的拼音,多个拼音用空格分隔
PYS	整型		拼音个数:记录一个汉字的读音总数
SYM	字符串	5	首音码:记录对应汉字的首音码

下面创建的数据表若没有特殊说明,均使用表 1 所示的数据表结构。

2 汉字型姓名转换为首音码的重码智能分析

只有一种读音的汉字显然首音码唯一,而对于多音字,若其所有拼音的首音相同,则首音码也唯一。如多音字“和 he

收稿日期:2005 - 03 - 09

作者简介:涂金德(1968 -),男,浙江温州人,讲师,主要研究方向:管理信息系统、计算机安全; 李永平(1955 -),男,浙江温州人,副教授,主要研究方向:管理信息系统.

hu huo”首音码为“h”。多音字的首音不同,才有首音重码问题,下面主要针对这种情况进行分析。

2.1　GB 汉字的首音重码分析

2.1.1　百家姓汉字的首音重码分析

百家姓中共有 444 个单姓,60 个复姓[4]。除姓“卻”(xi)外,其他姓氏汉字均属于 GB 汉字。本文创建百家姓汉字表 BJXHZ. DB(仅有 BH 和 HZ 两个字段),将文献[4]中的百家姓汉字录入到该表中;然后利用该表直接查询 GB 汉字拼音表(GBPY. DB)中的百家姓汉字拼音,与文献[4]的百家姓汉字读音进行对照分析,同时也考虑姓氏汉字用作名字时的读音习惯,并查阅金山词霸 2005;最后确定仍存在首音重码的姓氏汉字如表 2 所示。

表 2　存在首音重码的姓氏汉字

姓氏	拼 音	首音码	姓氏	拼 音	首音码
万	Wan mo	wm	翟	di zhai	zd
乐	le yue	yl	曾	zeng ceng	zc
强	qiang jiang	qj	查	cha zha	zc
盛	sheng cheng	cs	俟	qi si	qs
解	jie xie	xj	澹	tan dan	td
单	dan chan shan	scd	长	chang zhang	zc
仇	chou qiu	qc	车	che ju	cj
莘	shen xin	sx	佴	nai er	ne

由文献[4]和表 2 可知,所有复姓汉字中只有“万俟”(mo qi)、“单于”(chan yu)、“子车”(zi ju)存在首音重码。由于“万俟”、“单于”、“子车”复姓很罕见,为简化首音码表,这里暂时不考虑它们。若在实际中遇到这些复姓,也可以进行人工编辑。经上述简化,“万”(w)、“俟”(s)、“车”(c)只有一种首音码,“单”(s d)只有两种首音码。

在表 2 中,单姓汉字(解、单、仇、莘、翟、曾、查、澹、佴)虽均有两个首音码,但其用作姓氏的读音与用作名字的读音是相互排斥的,如“单”,用作姓时读“shan”,而用在名字中读“dan”。要解决这些汉字的首音重码问题,可以通过如下智能处理:将用作姓的首音码放在前面,用作名字的首音码放在后面,若该汉字出现在姓名的首位(即用作姓氏)则取前首音码,否则取后首音码。

而单姓汉字(乐、强、盛、长)用作姓氏时,首音码可以唯一确定,但用作名字时,存在首音重码。为了减少这些汉字的重码选择,可以作如下智能处理:将用作姓的首音码放在前面,另一个首音码放在后面,若该汉字出现在姓名的首位则取前首音码,否则再进行人工重码选择。

经上述分析,确定了所有百家姓汉字的首音码,然后生成百家姓的首音码表 BJXSYM. DB。由于百家姓单姓与复姓中的汉字有重复,如单姓“单”与复姓“单于”中都有“单”,将表 BJXSYM. DB 按 HZ 字段进行排序,删除有重复汉字的记录。

2.1.2　其他汉字的首音重码分析

去掉 GB 汉字拼音表 GBPY. DB 中的百家姓汉字(即在表 BJXSYM. DB 中的汉字)对应的记录,生成其他汉字的拼音表 GBQTPY. DB。经统计,GBQTPY. DB 表中多音字(即字段 PYS >1)共有 421 个,去掉首音相同的多音字(141 个),如“差 cha chai”、“大 da dai”、“着 zhe zhao zhuo”等,生成首音不同的汉字表 BTSYM. DB(共 280 个汉字)。表 BTSYM. DB 的 PY 字段中,放在开头的拼音是最常用的读音,分析时可以作为参考。

查阅金山词霸 2005,对首音不同的汉字表 BTSYM. DB 中各汉字进行分析,去掉在姓名中不使用的读音。仍存在首音重码的汉字主要有 12 个,如表 3 所示。如“朝”,可以有“梁朝(chao)伟”、“陈朝(zhao)霞”等,这些汉字需要人工选择首音重码。将上述分析确定的首音码存入 BTSYM. DB 表的 SYM 字段中。

表 3　存在首音重码的其他汉字

汉字	拼 音	首音码	汉字	拼 音	首音码
便	bian pian	bp	会	hui kuai	hk
参	can cen shen	cs	降	jiang xiang	jx
朝	chao zhao	cz	奇	qi ji	qj
传	chuan zhuan	cz	系	xi ji	xj
调	diao tiao	dt	行	xing hang heng	xh
恶	e wu	ew	重	zhong chong	zc

2.2　扩充汉字的首音重码分析

由于扩充汉字使用率很低,取其中一种常用首音码即可,这样就不存在首音重码选择。

经统计分析,扩充汉字拼音表 KCPY. DB 中首音不同的汉字共2868 个,其中有部分是繁体字。分离其简体字和繁体字,生成简体字表 QTJSYM. DB(2 198 个汉字)和繁体字表 QTFSYM. DB(670 个汉字)。对简体字表 QTJSYM. DB 中每个汉字,查阅金山词霸 2005,以国际标准汉字大字典为标准,均采用较常用的一种读音,将该首音码存入 QTJSYM. DB 表的 SYM 字段中。另外,繁体字表 QTFSYM. DB 中汉字首音码采用其对应的简体字首音码,对存在多种首音码的,也查阅金山词霸 2005,取常用的一种,同样将该首音码存入 QTFSYM. DB 表的 SYM 字段中。

3　汉字型姓名智能转换为首音码的码表生成

首先,生成 GB 汉字的首音码表 GBSYM. DB。合并首音不同的汉字首音码表 BTSYM. DB 和百家姓的首音码表 BJXSYM. DB 生成表 GBSYM. DB。然后向该表添加表 GBQTPY. DB 中首音码唯一的其他汉字(不在表 BTSYM. DB 和 BJXSYM. DB 中的汉字)及首音码。

其次,生成扩充汉字的首音码表 KCSYM. DB。合并简体字表 QTJSYM. DB 和繁体字表 QTFSYM. DB 生成表 KCSYM. DB。然后向该表添加扩充汉字拼音表 KCPY. DB 中首音码唯一的其他汉字(不在表 QTJSYM. DB 或 QTFSYM. DB 中的汉字)及首音码。

最后,生成 GBK 汉字的首音码表 GBKSYM. DB。合并 GB 汉字的首音码表 GBSYM. DB 和扩充汉字的首音码表 KCSYM. DB 即可生成表 GBKSYM. DB。

表 4　表 GBKSYM2. DB 中的记录

HZ	SYM	HZ	SYM	HZ	SYM	HZ	SYM
便	b * p	会	h * k	长	z * c	解	xj
参	c * s	降	j * x	重	z * c	查	zc
盛	c * s	奇	q * j	佴	ne	曾	zc
朝	c * z	强	q * j	仇	qc	翟	zd
传	c * z	行	x * h	单	sd		
调	d * t	系	x * j	莘	sx		
恶	e * w	乐	y * l	澹	td		

由于绝大部分 GBK 汉字的首音码唯一,为了减少首音码表的存储容量,将 GBK 汉字的首音码表分成两个表。首音码唯一的汉字存放在表 GBKSYM1. DB 中,有首音重码的汉字存放在表 GBKSYM2. DB 中。表 GBKSYM1. DB、GBKSYM2. DB

均仅有两个字段,即 HZ 和 SYM 字段,字段 HZ 存放单个汉字,字段 SYM 存放对应汉字的首音码。在表 GBKSYM1. DB 中,字段 SYM 宽度为 1 个字符即可,在表 GBKSYM2. DB 中,该字段宽度为 3 个字符。表 GBKSYM2. DB 的存储记录如表 4 所示,字段 SYM 中两首音码中间带“*”,表示对应汉字的首音码不能自动确定,需要人工选择。

由于 GBK 中 GB 汉字使用频度较高,为了提高查询速度,在表 GBKSYM1. DB 中 GB 汉字排序在前,扩充汉字排序在后。

4 汉字型姓名智能转换为首音码的实现

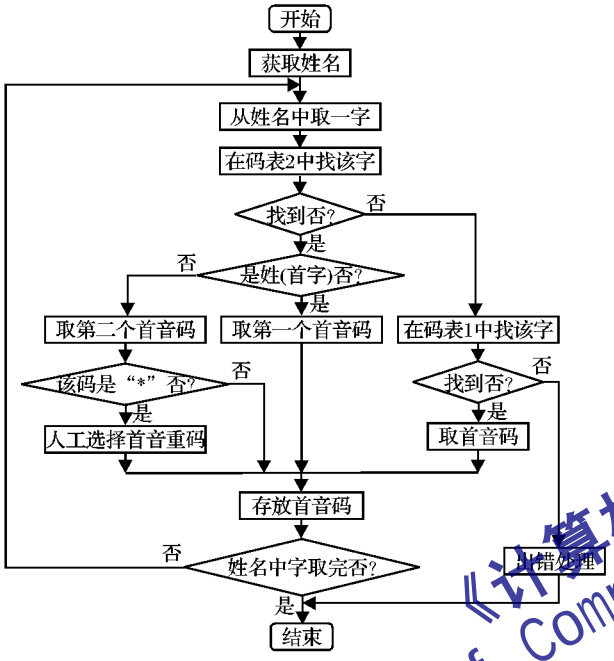

图 1 汉字型姓名智能转换为首音码的算法流程

使用上述分析生成的首音码表,并应用如图 1 所示的转换算法就可以方便地将输入的姓名自动转换成其对应的首音码。图 1 中的码表 1 和码表 2 分别指表 GBKSYM1. DB 和表 GBKSYM2. DB。由于表 GBKSYM2. DB 中汉字很少,为了提高查询速度,先查询该表,若找不到再查询表 GBKSYM1. DB。

5 结语

通过对 64000 多个无重复姓名进行测试,直接使用 GBK 拼音表首音重码选择率为 20%,去掉首音相同汉字后首音重码选择率降为 14.4%,而采用本文设计的码表及转换算法首音重码选择率降至 3.7%。可见,使用本码表及转换算法可以大幅度降低首音重码选择率,显著提高转换效率。该码表及算法已经在温州职业技术学院分院图书馆、温州菜篮子集团蔬菜种子批发公司中得到应用,效果很好。若对偶尔遇到的重码选择感到不便,也可以去掉图 1 中的“人工选择首音重码”功能,由系统自动组合各首音重码,就可以实现全自动转换。但这将增加一定的数据冗余,若想要去掉冗余数据,可以在转换后再进行人工编辑。当然,该码表可能忽略了一些汉字的首音重码,在使用时只要按上述规则添加即可。虽然本文仅对用作姓名的汉字的音码重码进行分析,但其分析方法也可以为解决汉字的其他方面(如药名、书名、歌曲名等)的首音重码问题提供参考。

参考文献:

[1] 李琦. 基于汉字拼音首字母的信息查询法的分析与实现[J]. 四川轻化工学院学报, 2003, 4(3): 71 - 74.

[2] 张春生, 廉洁, 包图雅. 拼音缩写码在医药行业管理中的应用[J]. 内蒙古民族大学学报, 2003, 18(2): 121 - 122.

[3] GBK, 汉字内码扩展规范[S], 1995.

[4] (宋)佚名, 王应麟, [梁]周兴嗣, (清)李毓秀, 木子. 百家姓 三字经 千字文 弟子规[M]. 乌鲁木齐: 新疆青少年出版社, 1996.

[5] 金山词霸 2005 专业版[CP/DK]. 金山公司, 2004.

(上接第 1712 页)

链接放在 PageRank 较低的页面,造成的 PageRank 损失较小。

任何一个网站都几乎不可能没有出站链接,但不巧的是,所有的“正常”链接都会泄漏 PageRank 值。但还有些“特别”的链接方式不用泄漏。PageRank 泄漏与否依赖于 Google 能否识别出链接,这样可以使用 Google 不能识别或是不考虑的链接,包括表单处理(form action)和包含 JavaScript 代码的链接[3]。

表单的 action 属性不一定是处理表单脚本的 url,它可以指向任何网站的任何一个页面。

例子:

```
<form name = "myform" action =
  "http://cs.scu.edu.cn/somepage.html" >
<a href = "javascript: document.myform.submit()" >
四川大学计算机学院 </a >
```

此外,action 属性甚至可以不必位于 form 表单而在 JavaScript 代码中,而 JavaScript 代码可以位于存储路径的 js 目录下,而该目录一般 Google 的 spider 程序都不访问。

3 总结及 PageRank 改进

PageRank 值由网络链接结构决定,与具体的检索内容无关,因而检索期间消耗很小,优于早期的 HITS 算法。在不考虑网页内容具体需要的情况下,提出的优化策略有利于提高网站在基于 PageRank 算法排名的搜索引擎搜索结果中的排名。这种效应也许短时间内尚不明显,但随着页面的增加和网站间链接的逐渐增多,最终的效果还是可观的。

同时,由于 PageRank 算法的检索无关性,也可能导致一些不利的结果,例如对一些词汇在特定的上下文中有特定的含义,或是一些专业词汇,仅仅依靠 PageRank 排名的结果可能不太令人满意,比如同样是查找“结构”这个词,在建筑学的上下文中,和芯片制造的上下文中,用户希望得到的检索结果必然不尽相同。但由于 PageRank 是网页的固定属性,可能就达不到期望的效果了。如果将整个互联网看成一个维度,那么 PageRank 则是该维度上的一个矢量,针对以上的缺陷,可以考虑建立这类矢量的一个矢量集。换句话说,可以针对某些指定的主题词计算出多个 PageRank 值,然后根据检索内容匹配相应主题词的网页 PageRank 值[4]。当然,在结果排序时用到的 PageRank 值仍然是唯一的。这种改进在检索期间的消耗上有所增加,但在结果排序上却有大大提高。

参考文献:

[1] BRIN S, PAGE L. The anatomy of a large-scale hypertextual web search engine[A]. Proceedings of the Seventh International World Wide Web Conference[C], 1998.

[2] BABA H, 马场肇. Google の秘密 - PageRank 彻底解说[EB/OL]. http://www.kusastro.kyoto-u.ac.jp/~baba/wais/pagerank.html, 2003.

[3] JEH G, WIDOM J. Scaling personalized web search[R]. Stanford University, 2002.

[4] HAVELIWALA TH. Topic-Sensitive PageRank[A]. Proceedings of the Eleventh International World Wide Web Conference[C], 2002.