机读词典快速检索杂凑算法的实现

唐家益 张玉琴 (陕西师范大学)

摘要:本文按照参考文献 *1* 给出的词汇 *T-Y* 码,*T-Z* 码和机读词典 *Hash* 索引文件的定义以及在 *Hash* 索引文件中冲突次数的计算公式和处理冲突的策略,为一个实用的机读词典建立了各级索引文件,试验结果具有参考价值。

一、机读词典的建立

作者在探索使用计算机测定英语语篇难度(见参考文献 3),英语语篇词汇量以及个人词汇量的研究过程中,需要有一个足以复盖一般英语语篇中绝大多数词汇的机读词典,以便查阅每一词汇的词频。为此我们从参考文献 2 所列 84761 种类型的词汇中整理了 21000 多个不同的词汇,并按照词频从高到低的顺序,划分为 21 个等级,每级 1000 个词汇,存入 IBM PC 计算机磁盘作为机读词典。由于在机读词典中,词汇是按照其词频的高低顺序排列的,故必须为它建立索引文件,以便实现随机查找。

二、一级 Hash 索引文件的构造

首先根据词典中词汇的数目确定一级 Hash 索引文件的大小,本例中根据参考文献 1 的结论取目录项数为较大素数 499997,约为词典中词汇数 21000 的 23.8 倍;然后再按照参考文献 1 中给出的杂凑函数 T-Z 的定义,计算词典中每个词汇的 T-Z 函数值,并用作该词汇在此索引文件中索引项的序号,这样就建立一级 Hash 索引文件 T。所占磁盘空间约为 1MB。

考虑到 Hash 函数 T-Z 值不唯一,故应有解决冲突的溢出处理策略。由于在机读词典中,词汇是按词频的顺序排列的,于是当两个词汇的 T-Z 函数值发生冲突时,只要将词频较高的记号汇的记录号留在一级 Hash 索引文件 T 中,而将词频较低的词汇的记录号放进溢出区,这样就可以保证绝大多数常用词汇能一次检出,即在一级 Hash 索引文件 T 中找到其在机读词典中的记录号。

初始化时,将一级 Hash 索引文件 T 的每一个记录中均置初值 0。由于词典中没有一个词汇的记录号为 0,因此如果某个词汇对应的索引项的内容为 0,立即可以断定词典中无此词汇,无需与词典中的词汇进行比较,更不用查阅二级或三级索引文件。这就极大地提高了检索的速度。于是除少数发生碰撞的词汇外,一次就可以定位或确定词典中没有该词汇。当然,由于在词典中只存放规则名词的单数和规则动词的原形,故名词复数,动词过去式,单数第三人称及分词等有规则变化的形式,在一级 Hash 索引文件 T 中查不到,还不能断定词典中无此词汇,必须恢复成其原形后再计算 1 次 T-Z 函数值,故要查阅 2 次一级 Hash 索引文件 T。

三、溢出区的结构

溢出区实际上也是与一级 Hash 索引文件 T 结构相同的索引文件,称为二级 Hash 索引文件 Ta,因而可以用类似的方法建立。不过因为其目录项数远小于一级 Hash 索引文件 T 的目录项数,故二级 Hash 索引文件 Ta 的体积也小得多。设二级 Hash 索引文件 Ta 的目录项数为 M,则可定义任一词汇 A 的另一个 Hash 函数,类似的 T-Z 码:

$$tz_1(A) = tz(A) \bmod M$$

本例中取二级 Hash 索引文件 Ta 目录项数 M 为 99991,其初始化方法与一级 Hash 索引文件 T 的相同。类似地,如果二级 Hash 索引文件 Ta 中也有冲突,可进行二次溢出处理,即建立三级 Hash 索引文件 Tb,等等。

表 1 列出了当一级 Hash 索引文件 T 的目录项数为 499979 时,词典中词汇的 Hash 函数值发生冲突的情形。表中所列预测值系按参考文献 1 中公式(1)计算结果,实际发生数则是统计结果。该表列出了词汇数从

1000 到 21000 时,在一级 Hash 索引文件 T 中冲突数随着增加的状况。

表 1

词汇数	冲突数	
	预测值	实际发生数
1000	1.0	0
2000	4.0	2
3000	9.0	6
4000	16.0	15
5000	24.9	23
6000	35.9	34
7000	48.8	51
8000	63.7	58
9000	80.5	81
10000	99.3	96
11000	120.1	114
12000	142.8	134
13000	167.5	158
14000	194.2	181
15000	222.8	207
16000	253.3	238
17000	285.7	272
18000	320.1	308
19000	356.5	338
20000	394.7	371
21000	434.9	421

四、实验及结果

在为该机读词典建立一级 Hash 索引文件 T 的过程中,实际发生冲突 421, 而根据参考文献[1]中公式计算:

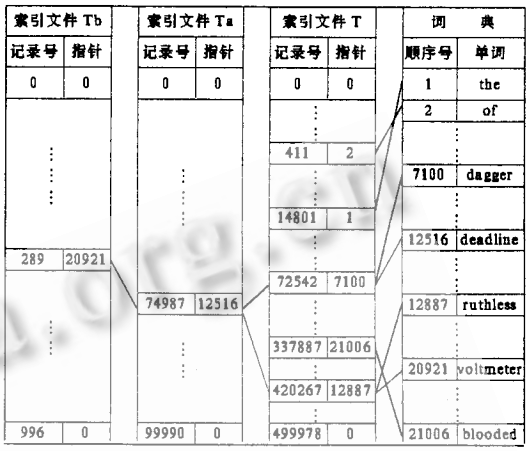
$$21006-499979(1-(1-1/499979)^{N})\approx 434.9$$

从表 1 中可以看出,两者基本相符,且常用词汇发生冲突的极少。例如,前 5000 词汇中只有 23 个,占 0.46%;前 10000 个词汇中只有 96 个占 0.96%。

实验结果还说明,如果一级 Hash 索引文件的目录项数接近或超过词典中词汇数的 25 倍,则需要查阅二级 Hash 索引文件的单词不到 2%,需要查阅三级 Hash 索引文件的更少,而且完全可以进行预测并控制发生冲突的次数。

机读词典,Hash 索引文件结构及冲突处理策略示于表 2。

表 2



从表 2 可以看出,词典中第 7100 个词汇 dagger 与第 12516 个词汇 deadline 在 Hash 索引文件 T 中的 T–Z 码相同,都是 72452,但 dagger 的词频高,故留在 Hash 索引文件 T 中,而将 deadline 放到 Hash 索引文件 Ta 中。类似地,对于词典中的 ruthless 和 voltmeter,词频低的 voltmeter 也和 deadline 一样被放到 Hash 索引文件 Ta 中。在 Hash 索引文件 Ta 中,这两个词汇的 T–Z 码一样,都是 74987,因而也发生了冲突,按同样的规则, voltmeter 被送入 Hash 索引文件 Tb,从这里也可以看出 T–Z 码的性质。通过对近 100 篇英语语篇(大部分为初中到大学及英语专业硕士研究生英语教材的整篇课文)的难度语篇词汇量的测试,在 286 微机上分析 1 篇课文的词汇量或难度平均不到半分钟,速度可以满足要求。

参考文献:

[1]唐家益,一种快速检索机读词典的杂凑算法,现代电子技术,1995 年第 1 期

[2]Carroletal.《Word Frequency Book》 Houphton Mifflin Company 1971

[3]唐家益,英语语篇难度的计算机测定,《外语教学》,1992 年第 2 期

[4]D.E.Knuth The Art of Computer Programming Volume3/ Sorting and Searching 1973