Rabin 指纹去重算法在搜索引擎中的应用①

贺建英

(四川文理学院 计算机学院, 达州 635000)

摘　要: 针对搜索引擎在海量数据中搜索速度慢, 占用存储空间大, 对重复的网页去重性差的现状, 提出一种基于 Rabin 指纹算法的去重方法, 不仅对搜索到的 URL 地址进行去重, 还对非重复 URL 地址对应的网页内容进行相似和相同的去重, 试验表明能有效地提高搜索速度、节省存储空间, 增强搜索的精度.

关键词: Rabin 指纹方法; 搜索引擎; 去重; URL; 海量数据

Application of Duplication Removal Method of Rabin Fingerprint in Search Engine

HE Jian-Ying

(College of Computer, Sichuan University of Arts and Science, Dazhou 635000, China)

Abstract: The existing search engine of massive data takes up large memory, needs much time and provides results of great duplication rate. To overcome these disadvantages, this paper proposes a duplication removal method based on the Rabin Fingerprint method, which cannot only remove the duplicated URL, but also remove the same even similar website content on different URL so that it can speed up the searching speed, save the memory capability and improve the accuracy of the research.

Key words: Rabin fingerprinting method; search engine; duplicate removal; URL; massive data

近年来由于信息技术的迅速发展, 互联网中的信息增长非常迅速, 几乎每 18 个月就翻一倍. 在海量的信息中想要找到需要的信息, 搜索引擎体现着它越来越重要的价值, 如大家熟悉的百度搜索、谷歌搜索等, 另外还有其他大大小小的搜索引擎. 在当前信息成倍增长的大数据时代, 提高搜索引擎中信息的采集速度和更新速度是非常有必要的. 搜索引擎的信息采集器通过网页的连接地址来搜索, 通常有 4 种抓取策略[1]: 1)深度优先; 2)宽度优先; 3)权重优先; 4)重访抓取.

然而在信息采集过程中可能会出现相同或相似的网页, 据统计在互联网上完全相同的网页占总网页数的 22%, 30%的页面与另外 70%的页面近似或者完全重复. 这样在采集网页时对页面的抓取、搜索和索引等将会大大消耗服务器的资源. 提高采集效率, 避免重复采集相同的页面是当务之急. 它需要一定的策略, 如判断某个页面如果是一个新页面, 则把该页面添加到采集列表中, 否则把该页面直接丢弃掉, 不再对该页面进行处理. 搜索引擎中去重的一般框架如图 1 所示.

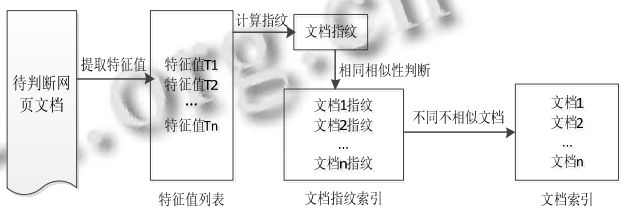


图 1　通用搜索引擎去重框架

该框架中通过抽取待判断网页文档的特征值, 进一步计算该文档的指纹, 把该文档的指纹与已有文档指纹索引中的指纹进行相同或相似性比较, 如果指纹不是相同或者相似则把该文档加入到文档索引列表中, 且把文档的文档指纹文件添加到指纹列表中. 文档的指纹计算方法有多种, 本文为提高搜索引擎的速度, 减少服务器负荷, 并增强系统的可扩展性, 特提出一

① 基金项目:国家档案局项目(2014-X-65)
收稿时间:2014-11-04;收到修改稿时间:2014-12-08

种适合搜索引擎的基于拉宾指纹算法的去重算法. 根据搜索引擎的去重原理知道, 对于搜索引擎采集器搜索到内容其实包含两部分: 1)找到搜索网页的 URL 地址; 2)通过 URL 地址找到搜索的网页. 所以在去重的过程中应当分两部分考虑: 首先查看搜索到的URL地址是否相同, 如果 URL 地址相同, 则直接不考虑重复的 URL 地址, 直接删掉; 如果 URL 地址不同, 则进一步判断该网页的内容跟已经搜索到的网页的内容是否相同, 如果相同也不再考虑该文档直接删掉, 否则加入搜索列表. 在文献[2]中只考虑到了对URL地址的去重, 而没有进一步考虑网页内容的去重.

1 Rabin指纹计算方法

在 1981 年美国哈弗大学教授 Michael O.Rabin 提出了 Rabin 指纹计算方法, 该方法在文件相似性检测[3-5]中应用非常广泛, 其思想如下:

假定 $X=([b_1, b_2, ...b_n])$包含了 n 个二进制符号串, 并且 $b_1=1$, 那么构造一个(n-1)阶的多项式 $X(f)$为式(1)所示:

$$X(f)=b_1f^{n-1}+b_2f^{n-2}+...+b_{n-1}f+b_n \quad (1)$$

再给定一个阶为 h 的多项式 $Y(f)$为式(2)所示:

$$Y(f)=a_1f^{h}+a_2f^{h-1}+...+a_{h-1}f+a_h \quad (2)$$

则当 Y 确定后, X 的拉宾指纹可方便的定义为式(3)所示:

$$R(X)=X(f)\bmod Y(f) \quad (3)$$

对于拉宾指纹算法, 如果字符串 X 的指纹不同于 Y 的指纹, 那么 X, Y 的字符串则不同, 即如公式(4)所示, 故可以根据此性质判断两个字符串是否相等.

$$R_{Rabin}(X)\neq R_{Rabin}(Y)\Rightarrow X\neq Y \quad (4)$$

假定字符串 X 和 Y 是不同的, 则其拉宾指纹冲突概率非常低, 即如式(5)所示:

$$P(R_{Rabin}(X)=R_{Rabin}(Y)\mid X\neq Y)<<1 \quad (5)$$

故不同的字符串, 其拉宾指纹相同的概率远远小于1, 可由此判断不同的字符串有不同的拉宾指纹, 另外如果两个字符串连在一起的拉宾指纹等于他们各自字符串的拉宾指纹之和, 即式 (6)所示:

$$R_{Rabin}(X+Y)=R_{Rabin}(X)+R_{Rabin}(Y) \quad (6)$$

当然拉宾指纹算法还有其他的性质, 在此就不再列举. 借助于上面的性质用于在搜索引擎去重中, 是本文的研究重点.

2 基于Rabin指纹算法的去重算法

2.1 基本思想

本算法的基本思想是分两步进行去重, 第一步通过拉宾指纹算法计算搜索引擎搜索到的网页 URL 地址的指纹, 建立一个 URL 地址指纹索引文件和一个URL地址索引文件, 把非重复的URL地址指纹存储到指纹列表中, 对搜索到的URL地址的指纹与地址指纹索引文件中的指纹进行比较, 利用拉宾指纹算法性质(4)判断地址字符串是否为相同的地址, 如果该地址指纹与地址指纹索引文件中的地址指纹不同, 则把该地址指纹加入地址指纹索引文件中, 把URL地址加入到URL 地址索引文件中. 再建立对应的向量, 使向量中下标的编号、指纹编号及URL地址索引编号一致, 该向量用来表示非重复的 URL 地址和重复地址及网页文档出现的频率. 第二步是考虑到不同的 URL 地址, 也可能出现相同内容的网页文档: 利用拉宾指纹计算这些文档的指纹文件, 删除重复网页文档. 对非重复的继续对网页内容进行分块计算其指纹, 根据两个网页文件的块指纹相同度在 70%, 则认为是相似文件, 则直接删除该网页.

2.2 实现过程

设计两个向量 A、B, 即 $A=(a_1, a_2, ...a_n)$, 其中 $a_i=\{0, 1\}$, i 为整数, i 在[0, +∞), 将它们的初值设置为 0. 向量 $B=(b_1, b_2, ...b_n)$, 其中 $b_i\in[1, +\infty)$, i 为整数, i 仍然在区间[0, +∞)上, 设置它们的初值为 0. 向量 A 和向量 B 一一对应. 在向量 A 中, 当 $a_i=0$ 时表示这个地址将不会出现在搜索的最终结果中, 则对应的URL地址指纹文件列表中的值也将设置为 NULL, 同时 URL 地址索引文件中对应位置上的值也设置为 NULL, 将不在进行一步的去重判断. 在向量 B 中对应位上的 b_i 将记录重复的网页或 URL 出现的次数. 建立的 URL 地址指纹索引列表和 URL 地址索引文件的索引号和向量 A、B 的下标号一一对应. 根据 Rabin 指纹算法性质(4)可知, 不同的指纹对应不同的字符串, 设置向量 A 中相同指纹的第一个下标对应的值为 1, 且在向量 B 的对应下标位置的值进行累加操作, 重复一次则 $b_i=b_i+1$. 如图 2 所示.

图 2 URL 去重操作图

在图 2 中, 假设搜索到的第 i 个和第 j 个 URL 地址指纹与 URL1 的地址指纹相同, 此时向量 A 中的 a_1=1, 则指纹向量 B 中的 b_1 累加 2 次, 得到 b_1=2. URL 去重算法描述如下:

1) 初始化向量A、B, 设置向量A、B中的值均为 0;

2) 通过 Rabin 指纹算法计算搜索到的 URL 地址的指纹, 利用 Rabin 性质(4)判断是否有相同的 URL 地址指纹, 如没有, 则把 URL 地址指纹添加到 URL 地址指纹列表中, 把 URL 地址加入到 URL 地址索引列表中;

3) 通过第 2 步, 对搜索到的每个 URL 地址进行去重判断, 对有重复的 URL 指纹, 在重复的地址指纹中找到索引号, 通过索引号把向量 A 中对应位置上的值设置为 1, 并在向量 B 的对应位置实施累加操作;

4) 遍历向量 B 中的值, 使用选择排序, 每趟选择最大的值放在第一个, 记住每趟最大值所在的下标, 在向量 A 和 URL 指纹索引文件中及 URL 地址文件索引做同步操作, 使排序后相同下标值对应的地址没改变;

5) 按 URL 地址出现的频率从高到低显示搜索到的非重复 URL 地址;

URL 地址去重后, 搜索到的结果仍然不一定是最优的. 在文献[1]中提到网页内容的完全相同或者相似率也非常高. 如果能减少这一部分重复, 则对搜索结果将会有更优解. 为了达到网页文档的去重, 在 URL 去重的基础上, 利用 Rabin 指纹计算方法完成对非重复 URL 指纹索引列表中地址所对应的网页文件的指纹计算, 形成一个网页文档的指纹索引列表. 如图 3 所示.

图 3　计算非重复 URL 指纹的网页文档指纹

根据得到的网页文档指纹利用 Rabin 性质(4)判断, 如果有完全相同的网页则保留第一个相同的网页文档, 同步把网页文档指纹列表中的重复的文档指纹设置为 NULL, 并在向量 A 和 B 中的对应下标处修改值, 即向量 A 中的值改为 0, 向量 B 中的值实现累加操作. 如假定索引为第 i 个和第 j 个 URL 对应的网页文档与第(i-m)个网页文档相同, 则在向量 A 中设置 a_i=0, a_j=0, 向量 B 中对应的 $b_{(i-m)}$的值+2; 其算法过程描述如下:

1) 对非重复的 URL 指纹列表中对应的网页文档使用 Rabin 指纹计算方法, 计算出每个网页文档的指纹, 并把它们放在网页文档指纹列表中;

2) 对网页文档指纹列表中的文件级指纹根据 Rabin 性质(4)进行文件级去重判断, 如果有重复指纹, 则重复指纹的第一个文档指纹保留, 把其余重复的文档指纹设置为 NULL, 并把对应位置上非重复 URL 指纹的也设置为 NULL;

3) 同步修改向量 A 中对应下标的值, 即 ai=0; 对向量 B 中对应下标的值做累加操作;

4) 遍历向量 B 中的值, 使用选择排序, 每趟选择最大的值放在第一个, 记住每趟最大值所在的下标, 在向量 A 和 URL 指纹索引文件中做同步操作, 使排序后相同下标值对应的地址没改变;

5) 按 URL 地址出现的频率从高到低显示搜索到的非重复 URL 且非重复的网页文档的 URL 地址指纹;

对网页文档进行文件级的去重后, 对不完全相同的网页文档进行定长分块计算其块级指纹, 如图 4 所示. 再对各个网页文档的分块指纹进行去重操作, 如果两个网页文档分块指纹的重复率达到 70%, 则这两个页面为相似页面, 在本算法中认为是重复页面, 则把对应的 URL 地址指纹中除第一个以外的其他相似页面的地址指纹设置为 NULL, 并同步操作向量 A、B, 方式雷同文件级去重操作. 最后经过对向量 B 的排序和向量 A 中值的同步操作, 在 URL 指纹索引列表中得到最优的地址指纹索引.

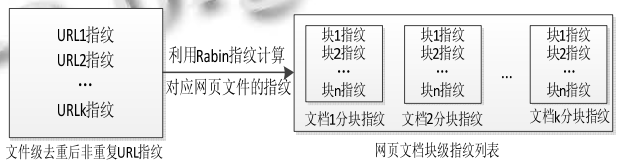
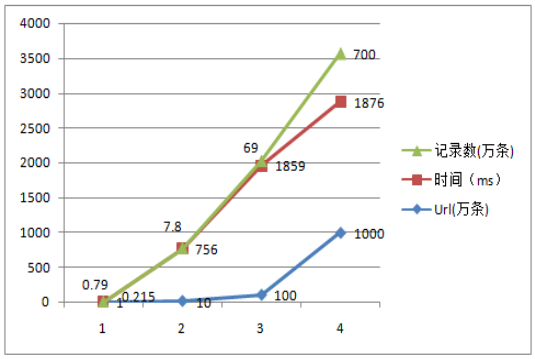


图 4　URL 对应的网页文档进行定长分块计算指纹

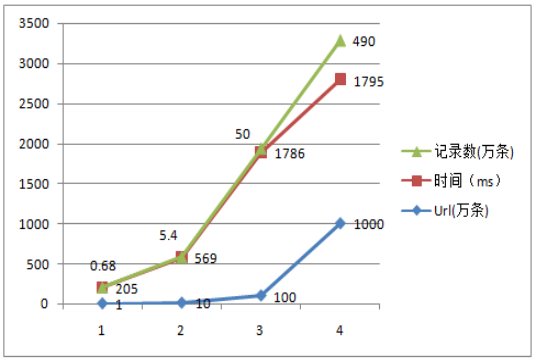
这种算法既考虑到搜索引擎所搜索 URL 地址的去重, 又考虑到虽然是不同地址但页面内容可能相同或相似的去重情况. 在首次搜索时虽然会花费比文献[2]更多的时间, 但精确度提升了不少, 且有效的 URL 地址数也明显减少, 大大的节省存储空间. 在后期的访问中访问一条有用的 URL 地址只需要一位, 根据索引寻址, 简化了检索过程, 提高检索的效率和精确度.

3 实验与分析

文献[2,3,6]中分别用 Rabin 指纹方法、Trie 树方法、Hflp 算法的 Hash 方法对 URL 地址去重进行了研究和测试, 但并没有进一步对所查找到的网页内容进行去重判断, 故本算法的测试结果无法同单一的 URL 地址去重进行对比. 为测试该算法的性能, 在 Windows 7 系统中用 Java 语言、JDK1.7 对该结果进行模拟实现搜索过程. 模拟网页页面数据分别是 1 万、10 万、100 万、1000 万条记录数. 通过不用 Rabin 指纹算法进行查找和使用 Rabin 指纹方法后去重进行查找. 其结果如下图 5 所示.



a)一般的去重方法查询结果



b)使用Rabin指纹算法的去重查询结果

图 5 测试结果对比图

对于同样的数据, 在一般的搜索去重方法中搜索相同内容的网页, 网页数量少时, 所消耗的时间区别不大, 查找到的网页数差别也不大, 但当网页数大大增多时, 本算法的优势就体现出来, 时间上虽只是略占优势, 但查找到的网页数也明显下降了不少. 且在本算法中还考虑经常使用的 URL 地址和相识相同网页访问率高, 显得该网页比较重要的特点, 让这些网页优先显示出来, 这样大大的提高了搜索的精度.

4 结语

在大数据时代的今天数据去重将会涉及到方方面面, 在搜索引擎中如何去除重复的 URL 地址和相同的网页, 提高访问效率和节省存储空间是势在必行的. 本文设计的基于 Rabin 指纹去重算法在搜索引擎中的应用, 就是从这两个方面考虑地址的去重和网页内容的去重. 对促进提高搜索引擎的搜索效率和精确度有一定的现实意义.

参考文献

1 搜索引擎网页去重算法分析.http://lusongsong.com/info/post/346.html. 2013.2.

2 梁正友,张林才.基于 Rabin 指纹方法的 URL 去重算法.计算机应用,2008,(12):193–194.

3 Broder AZ. Some applications of Rabin's fingerprinting method. Sequences II: Methods in Communications, Security, and Computer Science. New York. Springer-Verlag. 1993. 143–152.

4 Manber U. Finding similar files in a large file system. Proc. of the Winter 1994 USENIX Technical Conference. San Fransisco, CA, USA. 1994. 1–10.

5 Broder AZ. On the resemblance and containment of documents. Proc. of the International Conference on Compression and Complexity of Sequences (SEQUENCES). Positano, Salerno, Italy. 1997. 21–29.

6 叶允明,于水,马范援,宋晖,张岭.分布式 Web Crawler 的研究: 结构、算法和策略.电子学报,2002(Z1):2008–2011.

7 俞枫,王引娜.基于 DRPKP 算法的文本去重研究与应用.微型电脑应用,2014,(1):58–60.

8 魏建生.高性能重复数据检测与删除技术研究[学位论文].武汉:华中科技大学,2013.

9 Forman G, Eshghi K, Chiocchetti S. Finding similar files in large document epositories. Proc. of the 11th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD). Chicago, Illinois, USA. 2005. 394–400.

10 Broder AZ, Mitzenmacher M. Network applications of bloom filters: A survey. Internet Mathematics, 2004, (4): 485–509.

11 成功,李小正,赵全军.一种网络爬虫系统中 URL 去重方法的研究.中国新技术新产品,2014,(12):23.

12 李伟.Web 记录自动抽取与去重方法的研究与实现.西安电子科技大学学报,2014,(2):25–35.

13 孙有军,张大兴.海量图片文件存储去重技术研究.计算机应用与软件,2014,(4):56–57.