基于 Simhash 的中文文本去重技术研究

彭双和,图尔贡·麦提萨比尔,周巧凤
(北京交通大学 计算机与信息技术学院,北京 100044)

摘 要:随着计算机技术的飞速发展,各领域存储系统中的数据存储量迅猛上升,而其中的冗余数据也呈不断增加趋势。以往的研究表明,某些存储系统中的冗余数据已达 60%,其存储管理成本较高。处理冗余数据已成为目前存储系统研究的热点。为此,提出了一种基于 Simhash 的中文文本去重方案。该方案采用数据块作为粒度对重复数据进行去重处理,主要是将中文文本中的"。?!"等特殊字符作为分割点,对数据进行相应的分块处理,并以 Simhash 作为唯一标识,通过海明距离(Hamming Distance)来判断其相似性并以此为依据进行数据去重。对比验证实验结果表明,相比于传统的 hash 去重技术,提出的基于 Simhash 的去重方案具有更高的去重率和准确率,展现了较好的应用价值和应用前景。

关键词:重复数据删除;Simhash;hash;数据分块
中图分类号:TP311 **文献标识码**:A **文章编号**:1673-629X(2017)11-0137-04
doi:10.3969/j.issn.1673-629X.2017.11.030

Research on Deduplication Technique of Chinese Text with Simhash

PENG Shuang-he, Tuergong MAITISABIER, ZHOU Qiao-feng
(School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China)

Abstract: With the rapid development of computer technology, the amount of data storage in various areas of storage systems has been increased rapidly, of which the redundant data also does. Previous studies shown that some storage system has achieved 60% of redundant data, which displays the higher cost of storage management, so processing of that has become a hot spot for storage system research. For this, a method to duplicate redundant data based on Simhash is proposed, which uses the data blocks as the granularity to deduplicate the data, in which the special characters in Chinese documents, such as "。?!", are acted as split points for blocking. Simhash can be the only identifications and the similarity of those is judged by Hamming Distance for data duplication. Experimental results show that compared with the traditional hash deduplication technology, it has higher deduplication rate and accuracy, which displays good application value and application prospect.

Key words: data deduplication; Simhash; hash; data blocking

0 引 言

随着计算机与信息技术的不断发展,信息存储技术广泛应用于各个领域,导致数字化信息量迅猛增加。云存储、云计算技术的出现对数据中心存储能力也提出了更高要求。数据量出现指数级上升趋势,已从 TB 级提高到 EB 级,而云数据流量在 2013-2018 年间以 3.9 倍的速度增长[1]。因此,企业在面临庞大数据量的同时,更面临着数据的备份、恢复、管理以及保存数据成本等一系列问题。研究表明,在应用系统所保存的数据中,冗余数据约占 60% 左右[2],数据量增长的同时冗余数据也不断增多,为了确保数据保存的可靠持久,需要花费更多空间来存储并管理。因此,存储系统中数据冗余问题成为信息存储领域的研究重点。

重复数据删除技术是处理这一类问题的常用技术,也称为智能压缩或单一实例存储,是一种可自动搜索重复数据,将相同的数据只保留一个副本,并使用指向单一副本的指针替换其他重复副本,以达到消除冗余数据、降低存储容量需求的存储技术。该技术[3]的执行步骤为:利用分块方法对输入的大型数据进行分块,使用哈希(hash)算法给每一个块分配唯一的值作为唯一标识,新传入的块通过唯一标识与已存储的块进行比对,如果匹配将冗余数据删除,不匹配则存储新块。分割技术作为重复数据删除技术中的核心内容分为定长分块和变长分块两大类。假设一个文件中包含

收稿日期:2016-11-16 **修回日期**:2017-03-07 **网络出版时间**:2017-07-19
基金项目:中央高校基本科研业务费专项资金资助项目(2015JBM034)
作者简介:彭双和(1974-),女,讲师,研究方向为信息安全;图尔贡·麦提萨比尔(1989-),男(维吾尔),硕士研究生,研究方向为数据去重。
网络出版地址:http://kns.cnki.net/kcms/detail/61.1450.TP.20170719.1111.064.html

重复的数据块,或者在某一个文件中加入或删去部分内容,文件的新版本中会包含大量的重复数据,因此应用分块技术以更小粒度分割文件能提高去重率。常用的定长分块因其边缘敏感,微小变化即会导致“雪崩现象”,并且在一个分割好的文件块中的微小变化也会导致宏观上较大的变化,因此,在较小的粒度上,通过相似性来判断对数据块的最后处理,以达到有效的重复数据删除的目的。

为此,提出了一种基于动态分块和相似 hash(Simhash)的文本文档(txt,docx,pdf)有效去重方案,通过 Simhash[4-5] 判断文件及文件块的相似性,以获得更高的重复数据删除率和准确率。

1 现有相关技术分析

随着冗余数据的增多,删除重复数据成为了该领域的研究热点。去重技术按粒度的大小可分为文件级数据去重、块级数据去重和比特级数据去重。重复数据删除技术[6] 目前比较常用的是小粒度的数据去重技术。不同粒度会呈现不同的去重效果,去重效果随粒度变小而变好,但同时也导致了维护复杂度增高、性能降低等问题。如何平衡二者之间的关系,取决于企业的决策,也是企业控制成本的重要手段。

1.1 文件级数据去重

文件级数据去重技术又称 WFD 技术,即以文件为粒度查找重复数据的方法。此方法首先对整个文件进行 hash 计算,然后将该值与已存储的 hash 值进行比对,如果检测到相同的值,则仅将文件用指针替换,不进行实际存储,否则存储新文件。目前 Deep Store、TAPER、Foundation、Dedupvl[7] 等重删系统使用的是文件级去重技术。

1.2 块级数据去重

基于文件的数据去重不能对文件内部进行去重,因此研究者提出了更细粒度的去重技术。

1.2.1 固定大小分块技术(FSP)

基于固定尺寸划分去重的固定尺寸划分算法是按固定大小将文档分块,再计算每个块的 hash 值(常用 MD5,SHA-1)得到一个指纹值作为这个块的唯一标识,该指纹与已存的指纹进行比对,检测到相同的指向索引不存储,否则存储相应数据块。DBLK 中 Tsuchiya 等以 4 KB 作为一个粒度将数据分块并对其进行比对,获得了较高的重删率。目前 Venti、Symantec、iDedup[7] 等重删系统采用的是固定尺寸划分去重技术。该技术可以减少一定的存储空间,节省一定的网络带宽,但是变化敏感度很高,一个字符的变化将对影响重删效率产生极大影响,所以该技术比较适合更新少的数据,如图片、音频等文件。

1.2.2 基于内容的分块重删检测技术(CDC)

CDC[8] 算法是用 Rabin 指纹将文件分割成大小长度不一样的数据块的策略。与固定大小分块不同的是它对编辑及序列不敏感,变化只会影响两个相邻的数据块,但是分块完全取决于设定的期望块的大小,设定会直接影响该方法的去重效果。CDC 目前使用在 REBL、SiLo、ChunkStash 等重删系统以及 Pastiche 备份系统上。

2 基于 SimHash 的数据去重技术

在不同的命名、网页镜像或相似数据等情况下,因其特征或参数的原因往往无法确定相似的数据[9],虽然这些数据看似不同,但事实上存在很大的相似性。像重复数据删除一样,对于相似数据只想要存储数据的一个版本来节省存储空间。可是基于哈希值(hash value)的“数字指纹”特性,一个字符的不同将会导致整个哈希值的不同,因此相似数据的判断不够准确[10]。为此,提出了基于 Simhash 的相似文本数据去重技术。该技术将中文文本文件(TXT,DOCX,PDF)的句子作为最小单位,自然段作为块单位来对其进行数据去重。文中提出的基于 Simhash[11] 的数据去重技术包含四个步骤:数据分块、计算标识、进行比对、数据去重。

2.1 数据分块

目前数据分块中常用的方法是 FSP(Fixed-Size Partitsion)和 CDC[12](Content-Defined Chunk),虽然达到了不错的去重效果,可是对于一个数据块来说少量字符的变化将会导致整个数据块“标识”的不同,从而影响去重效果[13]。

将中文文本中的标点符号作为一个特征,对中文文本进行分块,按句子进行划分,且把一个自然段作为一个数据块,其流程如图 1 所示。

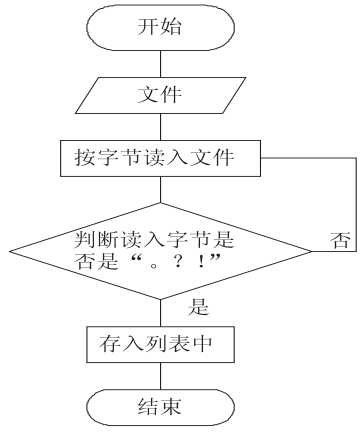

图 1 数据分块流程

具体步骤如下:

(1)读入整个文件;

(2)按字节读入文件到临时列表(list)中,当读入的字符是标点符号,如“。?!”,停止读入;

(3)将临时列表(list)中的数据放入列表(List)中,清空 list;

(4)返回第二步,继续判断。

最后可得若干 List,每个 List 里面的元素是中文的句子,而一个 List 代表一个自然段。

2.2　改进计算策略

如果继续使用 Hash 来对相似数据进行处理,难达到对相似数据判断的准确性。对于相似数据来说,使用 Simhash 算法以达到理想去重效果。Simhash 是一项数字指纹技术,基于一个文档的指纹是它的 hash 特性和相似文件有相似 hash 值的两个属性,可以在 hash 算法的基础上更准确地判断相似数据。

Simhash 像传统 hash 一样,可以作为文件的标识,因为对于完全相同的文件可以算出完全一样的 Simhash 值,而对于类似的文件通过 Simhash 可以得到类似的 Simhash 值,通过 Simhash 值的比对,可以得出两个文件的相似度。相对于传统 hash 值,Simhash 值更适合类似文件的比对。

2.3　计算 Simhash 值

根据 Charikar 提出来的算法思路,将 Simhash 与上述数据分块技术相结合,计算每个数据块,即以自然段的 Simhash 值来计算整个文本文档的唯一标识—Simhash 值。为了达到更高的计算速度,在原本的 Simhash 算法上做修改,首先不再选择特征词并计算其权重,而视每一个中文句子为一个特征量,然后不再根据特征词的出现次数设定不同的权重,而视每个特征的权重为 1。具体计算过程如图 2 所示。

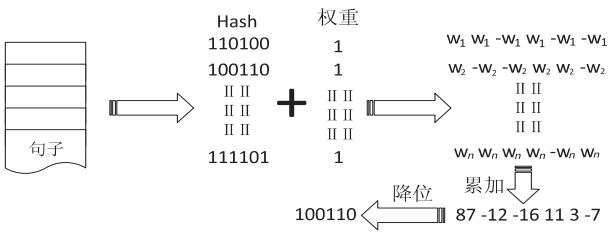

图 2　Simhash 计算流程

结合图 2 与文中方法,其计算步骤如下:

(1)确定指纹大小;

(2)创建一个 m 维向量,并初始化为 0。m 位的二进制数 S 初始化为 0;

(3)对 List 里每个元素(句子)使用 MD5 或者 SHA-1,产生一个 m 位的签名 G。对 $i=1$ 到 m:如果 G 的第 i 位为 1,则 V 的第 i 个元素加上该权重(默认每个句子的权重为 1),否则,V 的第 i 个元素减去该元素的权重;

(4)进行纵向累加,累加每一个元素加完权重以后的值。如果 V 的第 i 个元素大于 0,则 S 的第 i 位为 1,否则为 0;

(5)输出 S 作为 List 的签名;

(6)将计算出来的 S 作为这个数据块(List—自然段)的唯一标识,通过比较两个文件 S 的海明距离(Hamming Distance)得出数据块的相似度。

2.4　数据去重

应用上述方法对数据进行分块,再用 Simhash 计算每个数据块的唯一标识 S 后进行数据去重处理。

数据进行分块后应用 Simhash 算法进行标识,之后对得到的 S 与已存储的 S 做比对(做异或运算)得到相应的 Hamming Distance(两个二进制向量中不相同位的个数),比对二者相似程度,并通过预先设定的相似度阈值来判断这个文件的相似性;当得到的海明距离等于 0,则认为其完全相同;当小于预先设定的阈值,则认为其相似度较大,需对其进行进一步的分块,重新执行数据去重操作;若大于阈值,则认为其不相同,存储相应数据,当作一个新的数据块来处理。

基于 Simhash 算法的重复数据删除的具体流程如图 3 所示。

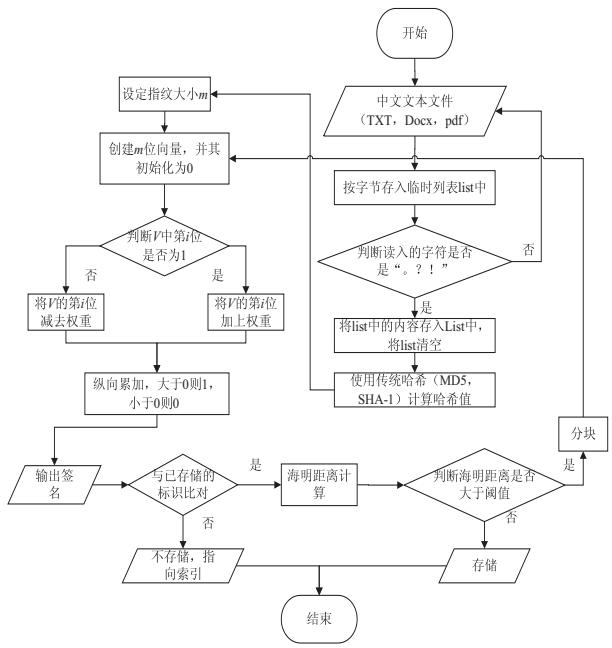

图 3　基于 Simhash 的重复数据删除的具体流程

通过大量实验发现,对于一段中文文档,当其相似度为 50% 时,它们的海明距离一般会在 50 ~ 70 之间,所以设定 70 作为阈值。当海明距离小于 70 时,表明这段数据相似度较高;如果高于 70,默认为不相同数据进行处理(所用到的文本权重默认为 1,权重的设定会影响海明距离的大小)。

3　实验结果分析

以中文文本为实验对象,通过基于 Simhash 的数

据去重方案,在可行性、去重率及准确率方面与传统去重技术进行比对。第一组实验通过 Simhash 与传统 hash 作对比证明其可行性;第二组实验通过新方案与现有的去重技术在去重率上进行对比;第三组实验进行准确率对比。

首先,为了证明 hash 与 Simhash 在数据去重上的差别,修改了目标文档的字符串,分别计算其 hash 值和 Simhash 值,比较使用两种标识是否达到字符串相似率判断的目的,结果如图 4 所示。

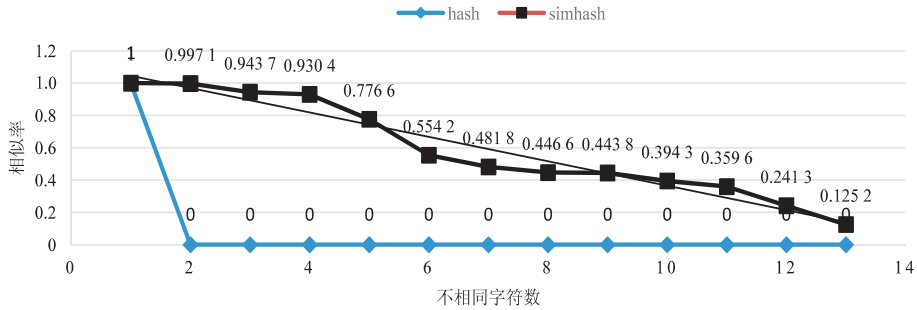

图 4 修改简单字符串后的 hash 值与 Simhash 值的对比

由图 4 可知,当字符串不发生任何变化时,通过 hash 值和 Simhash 值都能判断字符串是否完全相似,可是当字符串发生一个字节的变化,hash 值判断的相似率就会变成 0%,而使用 Simhash 判断的字符串,不会因为字符变化在相似率上发生过大的变化,随着不同字符的增多,相似率会下降且呈直线趋势,表明通过 Simhash 判断文件相似率是一种更有效的策略。

在实验 2 中,对不同大小的目标文件随机插入一些干扰项,进而应用固定大小分块(FS)去重算法、CDC 去重算法以及基于 Simhash 的数据去重算法比较其优劣,结果如图 5 所示。

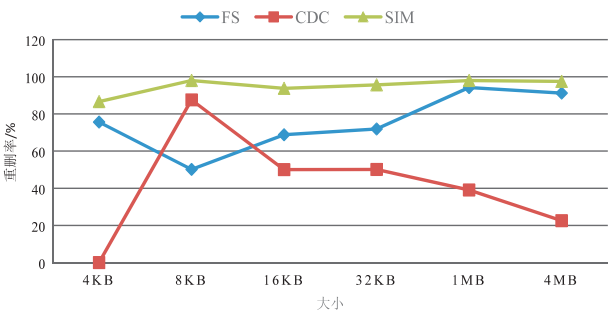

图 5 随机插入干扰项的不同算法去重率对比

由图 5 可知,不同算法的去重率[14](去重率 = $\frac{\text{BytesIn}-\text{BytesOut}}{\text{BytesIn}}\times 100\%$)随文件大小的改变而改变。FS 技术对于小文件效果比较明显,但对于较大文件其去重率表现不佳;CDC 技术随着文件的增大去重率显著增长;基于 Simhash 的数据去重方案对大文件或小文件都表现出了较稳定的去重率,较其他两种技术优势明显。

准确率(准确率 = 系统检测出的正确重复数据/系统检测出的重复数据)是数据去重上较为重要的因素,准确率高则表示此去重方法在重复数据删除上更为有效。因此,在实验 3 中,应用上述三种技术,处理不同大小的文档,其准确率比对结果如图 6 所示。

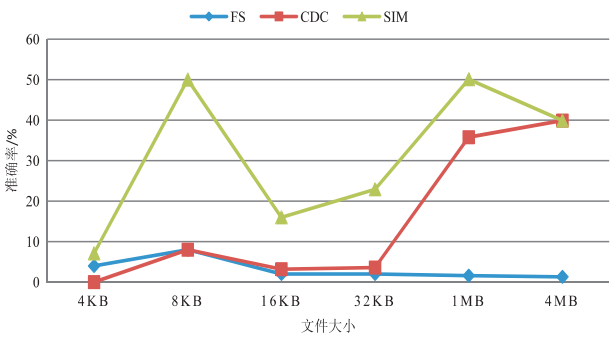

图 6 不同算法的准确率对比

由图 6 可知,FS 技术的准确率随着文件的增大呈下降趋势,且准确率一直偏低;CDC 和 Simhash 去重技术随着文件大小的变化准确率都表现出一定的波动,但 Simhash 算法始终保持相对较高的准确率,证明了其优越性。

4 结束语

针对存储系统中的冗余数据处理问题,提出了基于 Simhash 的重复数据删除技术,利用中文文本的特殊符号与 Simhash 结合达到了更高的去重率。实验结果表明,相较于其他去重技术,文中的技术方案在去重率、准确率等方面均呈现出了一定的优越性。

参考文献:

[1] 敖 莉,舒继武,李明强. 重复数据删除技术[J]. 软件学报,2010,21(5):916-929.

[2] Clements A T, Ahmad I, Vilayannur M, et al. Decentralized deduplication in SAN cluster file systems[C]//USENIX annual technical conference. [s. l.]:USENIX,2009:101-114.

[3] 付印金,肖 依,刘 芳. 重复数据删除关键技术研究进展[J]. 计算机研究与发展,2012,49(1):12-20.

[4] Charikar M S. Similarity estimation techniques from rounding algorithms[C]//Proceedings of the thirty-fourth annual ACM

(下转第 145 页)

2 000,已成功完成评测任务几十万次,这些数字还将随着系统的持续运行不断增长。

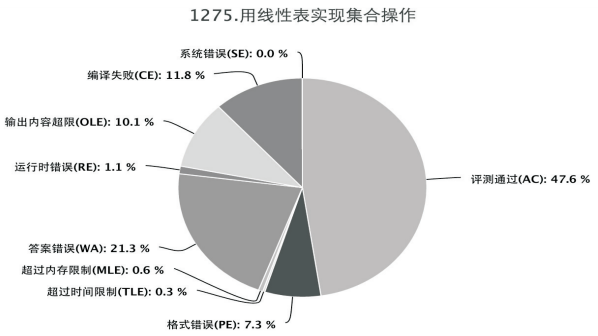

图 4　实际应用部分截图

8　结束语

以 ACM/ICPC 在线评测系统为蓝本,设计了程序设计实验教学在线评测辅助系统。系统对学生提交的程序源代码进行实时评测,具有人工评判所无法比拟的优点。

系统有别于传统意义上面向比赛的 OJ 系统,拥有更适合课程教学的教学中心和作业管理子系统,对程序设计类课程如 C/C++/Java/数据结构等的实验教学起到了很好的辅助作用,有效地提升了教学质量。文中设计的基于 setUID、LXC、全虚拟化、Linux 操作系统权限机制的多级沙箱模型,不但可以隔离非受信的源代码,而且降低了维护成本,兼顾了系统运行效率。为了适应大规模并发评测的需求,设计了基于 RabbitMQ-Celery 的生产者-消费者并发机制,在实际评测时系统可自动调整评测机的数量以适应不同规模的教学需求。系统已成为我院程序设计课程辅助教学的有利工具,对提高教学质量、提升学生学习兴趣很有帮助。

参考文献:

[1] 何钦铭,颜　晖,苏小红,等. "程序设计基础"课程教学实施方案[J]. 中国大学教学,2010(5):62-65.

[2] 葛文庚,蔺　莉. 程序设计基础课程教学模式研究与设计[J]. 电子设计工程,2012,20(4):44-46.

[3] 孟学多,俞雪永,颜　晖. 基于多核的在线判题系统的设计与研究[J]. 计算机时代,2011(7):7-9.

[4] 韩建平,刘春英,胡维华. "课内外贯穿,竞赛教学融合"的程序设计教学模式[J]. 实验室研究与探索,2014,33(6):169-171.

[5] 刘　楠,孙国道,田贤忠. ACM 在线评判系统设计与实现[J]. 计算机时代,2012(2):34-35.

[6] 谢　迪,李文新,郭　炜. "百练":一个程序设计技能训练与水平测试平台[J]. 合肥工业大学学报:社会科学版,2008,22(4):172-176.

[7] 张浩斌. 基于开放式云平台的开源在线评测系统设计与实现[J]. 计算机科学,2012,39(11A):339-343.

[8] 曾棕根. 源程序在线评测系统技术改进[J]. 计算机工程与应用,2011,47(4):68-71.

[9] 车明洙,纪洪波. 一种基于 ACM 程序设计竞赛在线评测系统解决方案[J]. 微型机与应用,2010(4):11-13.

[10] 蒋　辉,汪大菊. 在线评测系统的设计与实现[J]. 计算机与现代化,2012(2):111-115.

[11] 庄奇东,王键闻,张　楠,等. Online Judge 系统的优化[J]. 计算机系统应用,2011,20(8):115-121.

[12] 黄洪波,宋鸿陟,彭红星,等. 大规模程序评判系统的设计与实现[J]. 计算机工程与设计,2016,37(3):825-831.

[13] 韩君泽,钟　美,刘东升. 程序设计在线评测辅助教学系统的设计与实现[J]. 内蒙古师范大学学报:自然科学汉文版,2010,39(5):473-476.

[14] 陈丹伟,唐　平,周书桃. 基于沙盒技术的恶意程序检测模型[J]. 计算机科学,2012,39(6A):12-14.

[15] 吴佳杰. 基于 LXC 的 Android 系统虚拟化关键技术设计与实现[D]. 杭州:浙江大学,2014.

(上接第 140 页)

symposium on theory of computing. [s. l.]:ACM,2002:380-388.

[5] Manku G S,Jain A,Sarma A D. Detecting near-duplicates for web crawling[C]//Proceedings of the 16th international conference on world wide web. [s. l.]:ACM,2007:141-150.

[6] Denehy T E,Hsu W W. Duplicate management for reference data[R]. [s. l.]:[s. n.],2003.

[7] 谢　平. 存储系统重复数据删除技术研究综述[J]. 计算机科学,2014,41(1):22-30.

[8] Bobbarjung D R,Jagannathan S,Dubnicki C. Improving duplicate elimination in storage systems[J]. ACM Transactions on Storage,2006,2(4):424-448.

[9] 郭　颖,陈峰宏,周明辉. 大规模代码克隆的检测方法[J]. 计算机科学与探索,2014,8(4):417-426.

[10] Kulkarni P,Douglis F,Lavoie J,et al. Redundancy elimination within large collections of files[C]//Proceedings of USENIX technical conference. Berkeley,CA,USA:USENIX Association,2004.

[11] 王　格,吴　钊,李　向. 基于全文检索的文本相似度算法应用研究[J]. 计算机与数字工程,2016,44(4):567-571.

[12] Policroniades C,Pratt I. Alternatives for detecting redundancy in storage systems data[C]//Proceedings of USENIX technical conference. Berkeley,CA,USA:USENIX Association,2004.

[13] Zamora J,Mendoza M,Allende H. Hashing-based clustering in high dimensional data[J]. Expert Systems with Applications,2016,62:202-211.

[14] 尹美娟,陈庶民,刘晓楠,等. 基于邮件正文的邮箱用户别名抽取[J]. 计算机科学,2011,38(12):182-186.