**北京航空航天大学计算机学院**

**硕士学位论文开题报告**

**论文题目**：面向实时检索的搜索引擎缓存技术研究

**专 业**：计算机技术

**研究方向**：信息检索

**研 究 生**：欧韬

**学 号**：ZY1306107

**指导教师**：李建欣 王丽宏 马宏远

**北京航空航天大学计算机学院**

2014年11月17日

目录

1 课题来源 1

2 论文选题的背景与意义 1

3 国内外相关研究工作分析 2

3.1 面向有限容量缓存和固定索引的缓存技术研究 2

3.2 面向无限容量缓存和持续更新索引的缓存技术研究 3

4 研究内容与初步方案 4

4.1 论文研究目标和研究内容 4

4.2 初步方案 5

(一) 系统架构 5

(二) 文档纲要生成模块 5

(三) 缓存内容失效模块 6

(四) 查询预取更新模块 7

5 关键技术或难点 8

6 论文研究计划 8

6.1 研究计划 8

6.2 预期成果 8

7 主要参考文献 9

**面向实时检索的搜索引擎缓存技术研究**

# 课题来源

本课题来源于国家科技支撑计划和国家自然科学基金(61402123),Web搜索引擎的多层次缓存数据布局方法及实时检索缓存内容更新策略研究。

# 论文选题的背景与意义

据中国互联网信息中心（CNNIC）研究报告[22]显示，截止到2014年7月，中国网民达到6.32亿，其中Web搜索引擎用户规模为5.07亿，（以下简称搜索引擎）使用率为80.3%，搜索引擎已经是人们获取信息的主要工具。一方面，互联网网民增多，搜索引擎工作负载变大；另一方面，互联网网页也逐年增多，其规模已经超过1500亿，搜索引擎需要尽可能多地索引这些网页以提供准确和丰富的查询结果。在如此海量信息中，高效获取所需查询对学术界和工业界提出挑战。

缓存技术通过存放将来可能被请求的数据来提高数据访问速度。被存放的数据可能是计算结果或者数据的副本。如果请求的数据在缓存中被查找到，就直接获取该结果或者副本，否则，请求需要再次计算或者从其位置获得该数据。如果更多的请求能够被缓存所满足，则意味着系统的整体效率会被提高。缓存被广泛应用于计算机系统及应用软件的各个层次，譬如CPU缓存、磁盘缓存、Web缓存。

网络搜索引擎使用庞大的倒排索引在紧张的时延约束下处理用户查询，需要包含数千台计算机的大量设施和不停的投资来维护这些设施。优化网络搜索引擎效率对节省设备费用至关重要。即使微小的效率提高也可能为搜索引擎节约巨额的费用。最近的研究[6-15]表明，缓存技术（Caching）是优化搜索系统效率的关键技术。搜索引擎缓存给搜索系统带来令人满意的好处：它减少查询用户所察觉的时延，提升了用户体验。它减少后面查询处理器的负荷，提升了搜索系统的效率，进而可以减少搜索集群的数目，从而节省硬件资源，和降低能源消耗。

随着Web2.0时代的到来，网民对信息的实时性有了更高的需求，如方便用户查找筛选新闻、微博等时效性高、更新速度快的文档。实时检索(real time search)对搜索引擎缓存技术提出了新的挑战。搜索引擎为了更好地满足实时检索需求，需要频繁地更新倒排索引，会引发搜索引擎缓存的查询结果陈旧，使有效的缓存命中率下降。

传统的缓存替换策略面向相对固定的索引未考虑到倒排索引更新引发的搜索引擎缓存页面陈旧。因此，传统的缓存更新策略不能及时更新缓存内容使其与倒排索引一致，无法更好地适应实时检索系统。本文研究实时检索场景下的搜索引擎缓存技术。

# 国内外相关研究工作分析

调研了SIGIR、WWW、CIKM等信息检索领域顶级会议的论文，对搜索引擎缓存技术的研究分为如下两个阶段。第一阶段，2010年前，研究人员考虑有限容量的物理缓存而假设搜索引擎倒排索引固定不变，在此条件下研究搜索引擎缓存技术;第二阶段，从2011年开始，随着twitter, facebook等实时信息传递系统的流行，以雅虎研究院为代表的研究人员开始关注实时检索下的缓存技术。具体情况如下：

## 面向有限容量缓存和固定索引的缓存技术研究

2000年，Markatos[12]比较了LRU缓存替换策略及其若干变种和静态缓存策略的命中率，得出静态缓存策略仅适用于缓存容量较小的情况。在大容量缓存上的性能很差，操作系统里使用的基于替换策略如lRU或LFU的动态缓存能够达到相对更好的性能。可以认为静态缓存利用了用户查询分布的二八定律，将经常被用户发起的那20%查询结果存入静态缓存。而LRU策略利用了用户查询的时间局部性。2006年，Fagni等人[13]设计了一个双层缓存(Static-Dynamic Caching (SDC))，包括一个存储频繁查询的静态缓存和一个采用替换策略的动态缓存。SDC同时考虑到了查询的时间局部性和查询历史的统计特性。在Altavista日志上的实验表明，SDC缓存达到了超过30%的缓存命中率。

2002年，Xie等人[29]讨论了缓存位置的影响，分别讨论了客户端缓存、代理服务器缓存和搜索引擎服务器缓存技术。2003年，Lempel和Moran[11]计算衡量查询在将来被发起概率的得分，这个技术被称为概率驱动缓存(Probability Driven Caching (PDC)).2007年,Baeza-Yates等人[6]为搜索引擎结果缓存设计了许可策略(Admission Policies)，来阻止不频繁的查询通过替换策略获得频繁的查询在缓存中的空间，并在对用户查询日志的实验中得到了较好的缓存命中率。

查询结果并非唯一能可以被搜索引擎缓存的数据。2001年，Saraiva等人[32]提出一种两级缓存方案，结合了缓存查询结果和缓存被频繁访问的倒排记录表。2006年，Long 和Suel[31]扩展了这一思想，同时缓存一对总是在查询中一起出现的词项倒排记录表的交集。2008年，Baeza-Yates等人[14]看出，缓存存储倒排记录表可以比只存储查询结果和词项获得更高的缓存命中率。应当注意的是，在包含搜索引擎、倒排记录表缓存和查询结果缓存的大规模分布式系统中，它们可能不需要在同一台机器上竞争内存资源。2008年，SKobeltsyn等人[33]的工作描述了一个ResIn体系，缓存查询结果和修剪过的索引。2008年，Zhang等人[28]通过倒排表压缩结合缓存技术，取得了较好的缓存命中率。

以上的搜索引擎缓存技术研究对缓存记录活跃度、缓存位置、缓存数据类型等方面进行了研究，但没有考虑到搜索引擎更新对缓存内容有效性的影响。

## 面向无限容量缓存和持续更新索引的缓存技术研究

2010年，Cambazoglu等人[3]提出一个新颖的算法，用生存时间值来设置缓存实体(cache entries)的过期时间并选择性地通过向搜索引擎后台发送更新查询来更新缓存结果，这个算法的一个实现目前被yahoo所采用。2010年，Blanco等人[2]提出一种缓存内容失效预测(Cache Invalidation Predictor CIP)方案。搜索引擎后台为增、删或修改的文档产生文档概要，并发送给缓存。缓存根据文档概要判断文档的修改是否引起缓存中某查询结果陈旧。2011年,Alici等人[8]提出一种较为简单易于搜索引擎实现的缓存内容失效方案中，在文档被增、删和修改时仅需要相应调整倒排记录表和文档的版本时间戳。但在同等假正率的情况下未能达到采用CIP的缓存所达到的内容新鲜度。2012年,Jonassen等人[1]提出线下和线上两种预取策略来更新缓存中可能陈旧的查询结果。2013年，Sazoglu等人[5]考虑三种设置TTL的方式，基于时间的TTL、基于频率的TTL和基于用户点击的TTL，并考虑这三种方式的析取和合取。通过实验表明基于时间的TTL和基于频率的TTL以析取的方式结合能达到最好的性能。

面向无限容量缓存和持续更新索引的缓存技术研究认为:廉价的存储设备容量大到足够存储所有过去用户查询的结果。在缓存容量无穷大的假设下，除了特定查询的第一次发起(compulsory missed)，其它的查询都能由缓存提供答案。一个无穷大的缓存面临的关键问题是缓存结果陈旧(stale)的问题。借由一些启发式的算法，缓存结果的新鲜度得到了提升，但仍有很大的提升空间。

# 研究内容与初步方案

## 论文研究目标和研究内容

本论文的研究目标是通过研究实时检索的搜索引擎缓存技术，设计一种应用于搜索引擎的缓存机制，提升搜索引擎缓存内容的新鲜度，进而提升搜索引擎缓存的有效命中率。在此基础上实现一套面向实时检索的缓存系统。发表一篇学术论文，完成硕士毕业论文。

研究内容包括搜索引擎查询结果缓存(query result cache)的替换算法、缓存预取算法、缓存许可策略、和缓存内容失效技术等。

## 初步方案

### 系统架构

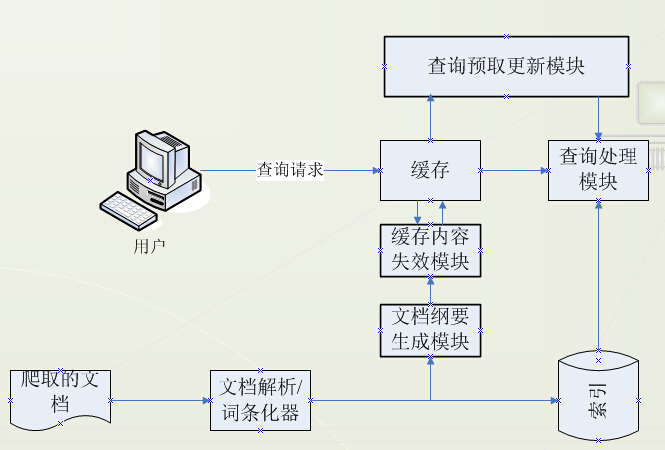


Figure 1　系统架构

如图1所示，系统的关键模块为1.文档纲要生成模块。2.缓存内容失效模块。3.查询预取更新模块。文档解析、索引器和查询处理模块拟调用lucene搜索类库实现，而在实验中爬取的文档拟采用搜狗搜索引擎的历史文档集来模拟。用户的查询请求从搜狗的用户查询日志中得到。实验后拟通过计算缓存命中率，并把缓存命中率与baseline:LRU缓存替换策略对比验证缓存技术的有效性。

### 文档纲要生成模块

事实上，缓存中的查询结果就是对给定查询文档得分排名前K的文档(链接)。通过估计被更改的文档对特定查询的得分，能大致预测此文档的改变是否影响查询的结果。文档纲要生成模块尝试发送相应文档的压缩表示，能由文档纲要大致估计对任意查询的文档得分。它的主要输出是由文档中TF-IDF权重高的词项组成的集合，文档可能因为这些词项而获得较高的查询文档得分。为了控制文档纲要的长度，文档生成模块只发送一定比率n(0<=n<=1)的查询到集合中。较短的文档纲要能降低通信量，但会增加缓存内容失效预测的错误率。文档微小的修改通常不会影响文档的得分排名。因此，需要比较文档修改前后的差异，仅对差异大于阀值di的文档生成纲要。拟采用Jaccard距离来衡量文档修改前后的差异。增加di能使该模块生成更少的文档纲要，进而减少通信量，但会降低缓存内容陈旧预测的准确率。

### 缓存内容失效模块

大多数情况下，一个文档的更新只可能影响到它匹配的查询。缓存内容失效模块接收到文档纲要之后，让所有缓存里的查询与文档纲要在合取模式下进行匹配。当文档纲要匹配上一个查询时，我们计算文档纲要对这个查询的文档得分。如果文档得分超过了缓存结果中排名最后的文档的得分。我们就认为文档的这次更新将会改变这条查询的结果，所以使这条查询的结果失效。

缓存内容失效模块还设置一个时间阀值ttl,并使所以在缓存中的年龄超过ttl的查询结果失效。这样给缓存里查询结果的陈旧度设置了一个上界。

此外，显然，如果一个查询结果包含被删除的文档那它一定是陈旧的。所以缓存内容失效模块将所有包含被删除文档的查询结果失效。

### 查询预取更新模块

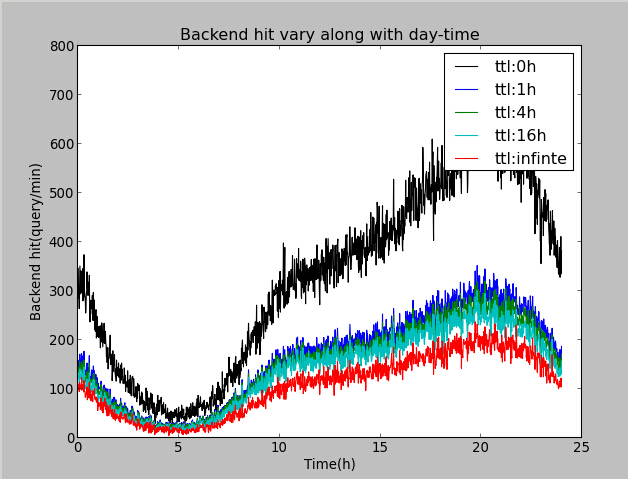


Figure 2 搜索引擎缓存后台负载周期

图2是我们从美国aol搜索引擎查询日志统计出的在不同缓存查询结果生存时间下，一天之内到达后台的请求数。由图三我们可以看出搜索引擎有许多时候处于闲置或低负荷状态。我们可以利用这些闲置的时间周期来预取更新那些在不久的将来会被用户发起的,在缓存中却即将陈旧的查询的结果。

查询预取更新模块拟采取线上和线下更新策略更新缓存。

线下更新策略利用历史查询记录来决定更新的查询。某些查询会在离上次被发起的固定一段时间dt后被再次发起。如，某些美剧一周更新一次，这使得许多观众会间隔一周的时间去搜索相应美剧的名字。将缓存里在与当前时刻相距dt的时候被发起查询放入等待更新的队列里，并按一定的度量计算其更新权重。根据当前搜索引擎后台负载更新队列更新权重较高的查询。

线上更新策略以历史查询记录为训练集尝试多种机器学习方法来预测缓存里查询再次被发起的时间间隔,选择预测最准确的方法对发起时间间隔进行预测。进而求得缓存里每个查询q再次被发起的时刻t。如果当前q的结果已陈旧，且当前时刻t0满足t-t0>ttl且将查询放入等待更新的队列里。

# 关键技术或难点

缓存内容陈旧预测过程中，即使获取了改动文档的概要，仍然无法准确地计算出缓存中的查询结果是否陈旧。文档微小的修改通常不会影响文档的得分排名因而不改变查询的结果。因此，需要比较文档修改前后的差异，仅对差异大于阀值di的文档生成纲要。增加di能使该模块生成更少的文档纲要，进而减少通信量，但会降低缓存内容陈旧预测的准确率。而减小di能检测出更多陈旧的查询，却会使假正率提高，即将一些仍然新鲜的查询预测为陈旧。

查询预取更新模块中无法精确地预测缓存中查询再次发起时间。需要选取一些合适的查询属性和合适的数据挖掘算法尽可能提高预测查询再次发起时间的精确度。

# 论文研究计划

## 研究计划

表2列出了论文的研究计划，论文研究分为六个阶段，每个阶段都有相应的目标和任务。

表1 研究计划表

|  |  |
| --- | --- |
| 2014.11 — 2014.12 | 学习相关资料和技术 |
| 2015.01 — 2015.03 | 实现分布式搜索引擎查询结果缓存系统 |
| 2015.03 — 2015.05 | 在缓存系统上实现LRU等各种缓存替换算法 |
| 2015.05 — 2015.09 | 实现他人论文中的缓存更新算法，用作baseline |
| 2015.09 — 2015.11 | 提出并实现面向实时检索的缓存策略与baseline比较，撰写小论文 |
| 2015.11 — 2015.12 | 撰写毕业论文，答辩 |

## 预期成果

论文研究的预期成果应该包括：开题过程中的开题报告，文献综述；中期检查的中期报告；最终的硕士学位论文，以及研究过程中的对于创新点的论文，系统实现的相关文档和代码。

# 主要参考文献

1. S.Jonassen, B.B. Cambazoglu, and F. Silvestri. Prefetching query results and its impact on search engines. *In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. SIGIR’12.ACM.*
2. [Roi Blanco](http://dblp.uni-trier.de/pers/hd/b/Blanco:Roi), [Edward Bortnikov](http://dblp.uni-trier.de/pers/hd/b/Bortnikov:Edward), [Flavio Junqueira](http://dblp.uni-trier.de/pers/hd/j/Junqueira:Flavio), [Ronny Lempel](http://dblp.uni-trier.de/pers/hd/l/Lempel:Ronny), [Luca Telloli](http://dblp.uni-trier.de/pers/hd/t/Telloli:Luca), [Hugo Zaragoza](http://dblp.uni-trier.de/pers/hd/z/Zaragoza:Hugo). Caching search engine results over incremental indices. [SIGIR 2010](http://dblp.uni-trier.de/db/conf/sigir/sigir2010.html#BlancoBJLTZ10): 82-89
3. [Berkant Barla Cambazoglu](http://dblp.uni-trier.de/pers/hd/c/Cambazoglu:Berkant_Barla), [Flavio Paiva Junqueira](http://dblp.uni-trier.de/pers/hd/j/Junqueira:Flavio_Paiva), [Vassilis Plachouras](http://dblp.uni-trier.de/pers/hd/p/Plachouras:Vassilis), [Scott A. Banachowski](http://dblp.uni-trier.de/pers/hd/b/Banachowski:Scott_A=), [Baoqiu Cui](http://dblp.uni-trier.de/pers/hd/c/Cui:Baoqiu), [Swee Lim](http://dblp.uni-trier.de/pers/hd/l/Lim:Swee),[Bill Bridge](http://dblp.uni-trier.de/pers/hd/b/Bridge:Bill). A refreshing perspective of search engine caching. [WWW 2010](http://dblp.uni-trier.de/db/conf/www/www2010.html#CambazogluJPBCLB10): 181-190
4. [Xiao Bai](http://dblp.uni-trier.de/pers/hd/b/Bai_0002:Xiao), [Flavio Paiva Junqueira](http://dblp.uni-trier.de/pers/hd/j/Junqueira:Flavio_Paiva). Online result cache invalidation for real-time web search. [SIGIR 2012](http://dblp.uni-trier.de/db/conf/sigir/sigir2012.html#BaiJ12): 641-650
5. [Fethi Burak Sazoglu](http://dblp.uni-trier.de/pers/hd/s/Sazoglu:Fethi_Burak), [Berkant Barla Cambazoglu](http://dblp.uni-trier.de/pers/hd/c/Cambazoglu:Berkant_Barla), [Rifat Ozcan](http://dblp.uni-trier.de/pers/hd/o/Ozcan:Rifat), [Ismail Sengör Altingövde](http://dblp.uni-trier.de/pers/hd/a/Alting=ouml=vde:Ismail_Seng=ouml=r), [Özgür Ulusoy](http://dblp.uni-trier.de/pers/hd/u/Ulusoy:=Ouml=zg=uuml=r). Strategies for setting time-to-live values in result caches. [CIKM 2013](http://dblp.uni-trier.de/db/conf/cikm/cikm2013.html#SazogluCOAU13): 1881-1884
6. [Ricardo A. Baeza-Yates](http://dblp.uni-trier.de/pers/hd/b/Baeza=Yates:Ricardo_A=), [Flavio Junqueira](http://dblp.uni-trier.de/pers/hd/j/Junqueira:Flavio), [Vassilis Plachouras](http://dblp.uni-trier.de/pers/hd/p/Plachouras:Vassilis), [Hans Friedrich Witschel](http://dblp.uni-trier.de/pers/hd/w/Witschel:Hans_Friedrich). Admission Policies for Caches of Search Engine Results. [SPIRE 2007](http://dblp.uni-trier.de/db/conf/spire/spire2007.html#Baeza-YatesJPW07): 74-85
7. [Hongyuan Ma](http://dblp.uni-trier.de/pers/hd/m/Ma:Hongyuan), [Wei Liu](http://dblp.uni-trier.de/pers/hd/l/Liu:Wei), [Bingjie Wei](http://dblp.uni-trier.de/pers/hd/w/Wei:Bingjie), [Liang Shi](http://dblp.uni-trier.de/pers/hd/s/Shi:Liang), [Xiuguo Bao](http://dblp.uni-trier.de/pers/hd/b/Bao:Xiuguo), [Lihong Wang](http://dblp.uni-trier.de/pers/hd/w/Wang:Lihong), [Bin Wang](http://dblp.uni-trier.de/pers/hd/w/Wang:Bin). PAAP: prefetch-aware admission policies for query results cache in web search engines. [SIGIR 2014](http://dblp.uni-trier.de/db/conf/sigir/sigir2014.html#MaLWSBWW14): 983-986
8. [Sadiye Alici](http://dblp.uni-trier.de/pers/hd/a/Alici:Sadiye), [Ismail Sengör Altingövde](http://dblp.uni-trier.de/pers/hd/a/Alting=ouml=vde:Ismail_Seng=ouml=r), [Rifat Ozcan](http://dblp.uni-trier.de/pers/hd/o/Ozcan:Rifat), [Berkant Barla Cambazoglu](http://dblp.uni-trier.de/pers/hd/c/Cambazoglu:Berkant_Barla), [Özgür Ulusoy](http://dblp.uni-trier.de/pers/hd/u/Ulusoy:=Ouml=zg=uuml=r). Timestamp-based result cache invalidation for web search engines. [SIGIR 2011](http://dblp.uni-trier.de/db/conf/sigir/sigir2011.html#AliciAOCU11): 973-982
9. [Xiao Bai](http://dblp.uni-trier.de/pers/hd/b/Bai_0002:Xiao), [Flavio Paiva Junqueira](http://dblp.uni-trier.de/pers/hd/j/Junqueira:Flavio_Paiva), [Adam Silberstein](http://dblp.uni-trier.de/pers/hd/s/Silberstein:Adam). Cache refreshing for online social news feeds. [CIKM 2013](http://dblp.uni-trier.de/db/conf/cikm/cikm2013.html#BaiJS13): 787-792
10. Mauricio Marin, Veronica Gil-Costa, Carlos Gomez-Pantoja. New caching techniques for web search engines. In: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, 2010
11. R. Lempel, and S. Moran. Predictive caching and prefetching of query results in search engines. In WWW, pp 19-28, 2003
12. E. Markatos. On caching search engine query results. Computer Communications, 24(7), 2000.
13. T. Fagni, R. Perego, F. Silvestri, and S. Orlando. Boosting the performance of Web search engines. Caching and prefetching query results by exploiting historical usage data. ACM TOIS, 24(1):51-78, 2006.
14. R. Baeza-Yates, A. Gionis, F. Junqueira, V. Murdock, V. Plachouras, and F.Silvestri. Design trade-offs for search engine caching. ACMTWEB, 2(4):1-28,2008.
15. D. Puppin, F. Silvestri, R. Perego, and R. Baeza-Yates. Load-balancing and caching for collection selection architectures. In INFOSCALE, 2007.
16. D. Puppin, F. Silvestri, R. Perego, and R. Baeza-Yates. Tuning the Capacity of Search Engines: Load-driven Routing and Incremental Caching to Reduce and Balance the Load. To appear in TOIS, 2009.
17. B.B. Cambazoglu and R. Baeza-Yates. Scalability challenges in web search engines. In M. Melucci, R. Baeza-Yates, and W. B. Croft, editors, Advanced Topics in Information Retrieval, volume 33 of The Information Retrieval Series, pages 27-50. Springer Berlin Heidelberg,2011.
18. L. A. Barroso, J.Dean, and U. Holzle. Web search for a planet: the Google cluster architecture. IEEE Micro,23(2):22-28,2003
19. 马宏远,王斌.一种基于预取感知接纳策略的查询结果缓存方法[J]计算机研究与发展，2012(S1)
20. 马宏远,王斌.一种基于查询特性的查询结果缓存与预取方法[J]中文信息学报,2011(5):37-43
21. 马宏远,王斌.基于日志分析的搜索引擎查询结果缓存研究[J]计算机研究与发展,2012
22. China internet Network Information Center (CNNIC). The 34th Statistical Report on Internet Development in China. [2014-07-21] http: //www.cnnic.net. cn/hlwfzyj/hlwxzbg/hlwtjbg/201407/P020140721507223212132. pdf
23. Cockburn A, Jones S. Which way now? Analysing and easing inadequacies in WWW navigation. International Journal of Human-Computer Studies, 1996, 45:105-129
24. Tauscher L, Greenberg S. How people revisit web pages: Empirical findings and implications for the design of history system. International Journal of Human-Computer Studies, 1997,47:97-137
25. Silverstein C, Henzinger M, Marais H,et al. Analysis of a very large web search engine query log. SIGIR 1998:6-12
26. Baeze-Yates R. Web Usage Mining in Search Engines. Reading, MA: Addison Wesley, 1999:307-322
27. Baeze-Yates R. Gionis A, Junquerira F, et al. The impact of caching on search engines. SIGIR 2007:183-190
28. Zhang J, Long X, Suel T, et al. Performance of compressed inverted list caching in search engines. WWW 2008:387-396
29. Xie Y, O’Hallaron D R. Locality in search engine queryies and its implications for caching. IEEE 2002:1238-1247
30. Li Y, Zhang S, Wang B, et at. Characteristics of Chinese Web searching: A large-scale analysis of Chinese query logs. Journal of Computational Information Systems, 2008, 4(3):1127-1136
31. [Xiaohui Long](http://dblp.uni-trier.de/pers/hd/l/Long:Xiaohui), [Torsten Suel](http://dblp.uni-trier.de/pers/hd/s/Suel:Torsten). Three-Level Caching for Efficient Query Processing in Large Web Search Engines. [World Wide Web 9(4)](http://dblp.uni-trier.de/db/journals/www/www9.html#LongS06): 369-395 (2006)
32. [Patricia Correia Saraiva](http://dblp.uni-trier.de/pers/hd/s/Saraiva:Patricia_Correia), [Edleno Silva de Moura](http://dblp.uni-trier.de/pers/hd/m/Moura:Edleno_Silva_de), [Rodrigo C. Fonseca](http://dblp.uni-trier.de/pers/hd/f/Fonseca:Rodrigo_C=), [Wagner Meira Jr.](http://dblp.uni-trier.de/pers/hd/m/Meira_Jr=:Wagner), [Berthier A. Ribeiro-Neto](http://dblp.uni-trier.de/pers/hd/r/Ribeiro=Neto:Berthier_A=), [Nivio Ziviani](http://dblp.uni-trier.de/pers/hd/z/Ziviani:Nivio). Rank-Preserving Two-Level Caching for Scalable Search Engines. [SIGIR 2001](http://dblp.uni-trier.de/db/conf/sigir/sigir2001.html#SaraivaMFMRZ01): 51-58
33. [Gleb Skobeltsyn](http://dblp.uni-trier.de/pers/hd/s/Skobeltsyn:Gleb), [Flavio Junqueira](http://dblp.uni-trier.de/pers/hd/j/Junqueira:Flavio), [Vassilis Plachouras](http://dblp.uni-trier.de/pers/hd/p/Plachouras:Vassilis), [Ricardo A. Baeza-Yates](http://dblp.uni-trier.de/pers/hd/b/Baeza=Yates:Ricardo_A=). ResIn: a combination of results caching and index pruning for high-performance web search engines. [SIGIR 2008](http://dblp.uni-trier.de/db/conf/sigir/sigir2008.html#SkobeltsynJPB08): 131-138