

PAAP: Prefetch-Aware Admission Policies for Query Results Cache in Web Search Engines*

Hongyuan Ma, Wei Liu, Bingjie Wei,
Liang Shi, Xiuguo Bao, Lihong Wang
CNCERT/CC
Beijing, China
mahongyuan@foxmail.com

Bin Wang
Institute of Computing Technology of the
Chinese Academy of Sciences
Beijing, China
wangbin@ict.ac.cn

ABSTRACT

Caching query results is an efficient technique for Web search engines. Admission policy can prevent infrequent queries from taking space of more frequent queries in the cache. In this paper we present two novel admission policies tailored for query results cache. These policies are based on query results prefetching information. We also propose a demote operation for the query results cache to improve the cache hit ratio. We then use a trace of over 5 million queries to evaluate our admission policies, as well as traditional policies. Experimental results show that our prefetch-aware admission policies can achieve hit ratios better than state-of-the-art admission policies.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*search process*; H.3.4 [Information Storage and Retrieval]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*

General Terms

Performance, Experimentation

Keywords

Web search engine, admission policy, caching, prefetching

1. INTRODUCTION

Caching is an effective technique to improve the performance of large scale Web search engines. Results cache stores the previous search results which were recently computed to resolve future queries. Web search engines may also prefetch some search engine results pages which are the

*Supported by the Youth Foundation of CNCERT (No.2013QN-18), the National Key Technology R&D Program (No.2012BAH42B01) and the National Natural Science Foundation of China (No.61300206)

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

SIGIR '14 July 06 - 11 2014, Gold Coast, QLD, Australia
Copyright 2014 ACM 978-1-4503-2257-7/14/07 ...\$15.00.

listing of Web pages returned by the Web search engines in response to a query that it predicts to be requested in the near future [1]. In addition to these approaches, query results cache admission policy is employed to prevent infrequent queries from taking space of more frequent queries in the cache.

Web search engine cache admission policy has been studied by a number of researchers. As we know, most of earlier admission policies use either stateless features, such as the length of the query in words, the length of the query in characters, which depend only on the query, or stateful features, such as the past frequency of the query, which are based on historical information. "It is hence an open problem if there exists a feature (or combination of features) that can achieve a performance as good as the one of stateful features." presented by Baeza-Yates et al.[2] is a research challenge in query results cache admission policy.

We take into consideration the fact that prefetching query results can result in infrequent queries in the cache. This paper concentrates on results cache admission policy, and in particular discusses the application of query results prefetching information in the context of Web search engine results cache admission policy. Our goal is to design new admission policies for query results cache that exploit the query results prefetching information to prevent infrequent or even singleton queries from polluting the cache. To do this, we divide results cache into two areas: controlled cache and uncontrolled cache, and evaluation functions based on query results prefetching information were used to decide whether a query is infrequent or not and store in which part of the results cache. Generally speaking, the controlled cache will contain those queries that are likely to be hits. Ideally, with a perfect admission policy, there would be no need for the uncontrolled cache. However, implementing such an admission control policy is very difficult. In addition, we propose a demote operation for the query results cache to improve the cache hit ratio.

2. RESULTS PAGE ACCESS PATTERN

2.1 Analysis of the Search Engine Query Log

We use a query log from a famous Chinese Web search engine which contains around 28 million queries of about 5.8 million people for a period of 2-months (from 09/01/2006 to 10/31/2006) to explore Web search engine query characteristics. The analysis of the search engine query log shows that query reused distance obeys Zipf law which can be expressed as $y = Kx^{-\alpha}$, x represents the rank of query reused

distance and the value of parameter α is 0.49 in our dataset.

2.2 Query Results Page Access Pattern

Let’s recall how Web search engine works. It can be summarized in the following procedure: (1) Web search engine user formulates a query according to his information need, and the query request includes user query interests; (2) Web search engine receives the request, and do index retrieval, similarity calculation, ranking, generating query results page with results URL content snippet; (3) Returns final query results page to the user. It is note that the query result is returned in units of pages. When a user submit an initial query to Web search engine, the first page, namely page number 1, would be returned to the user, and the user would decide whether to submit subsequent page number request based on the quality of current returned query results page.

Query	1157071632984	01/Sep/2006:08:47:13	李宇春	1
Query	1157072044546	01/Sep/2006:08:51:31	李宇春	1
Query	1157072044546	01/Sep/2006:08:51:44	李宇春	2p
Query	978280076820	01/Sep/2006:09:45:46	李宇春	1
Query	1151472574296	01/Sep/2006:14:57:56	李宇春	1
Query	1151472574296	01/Sep/2006:14:58:18	李宇春	2p
Query	1151472574296	01/Sep/2006:14:58:39	李宇春	3p
Query	1151472574296	01/Sep/2006:15:03:04	李宇春	5p
Query	1157099281740	01/Sep/2006:16:15:34	李宇春	1
Query	1153147858234	01/Sep/2006:16:15:15	李宇春	1
Query	1153147858234	01/Sep/2006:16:15:33	李宇春	2p
Query	1153147858234	01/Sep/2006:16:15:42	李宇春	3p
Query	1153147858234	01/Sep/2006:16:16:03	李宇春	4p
Query	1153147858234	01/Sep/2006:16:16:06	李宇春	5p

Figure 1: Some query requests about "Li Yuchun" (in Chinese) (From a famous Chinese Web search engine)

Here, we divide queries submitted by users into the following two categories: (1) Initial Query: The first query request on a topic in a session; (2) Follow-up Query: All query request following the initial query in a session. Figure 1 shows several query sessions about the topic "Li Yuchun" a famous female singer in China from a famous Web search engine query log on September 1, 2006 (Note that these are only a part of queries about the topic "Li Yuchun"). Column 5 is the query results page number and follow-up query is marked "p" in the tail. Taking the queries about the topic "Li Yuchun" submitted by the user "1151472524296" for example, after an initial query, the user submits three follow-up queries with query results page number 2, 3 and 5. We find that this query results page access pattern is widespread in Web search engines and we define this pattern as follow-up pattern. Therefore, prefetching in advance subsequent query results pages can reduce the latency of future query requests.

2.3 Impact of Access Pattern on Query Results Cache

Follow-up pattern means that Web search engines have good spatial locality, and prefetching is useful to improve system performance. We define query as $Query(Topic, Page_{no})$. When the request with page number $Page_{no}$ is received by Web search engine, it would prefetch query results page

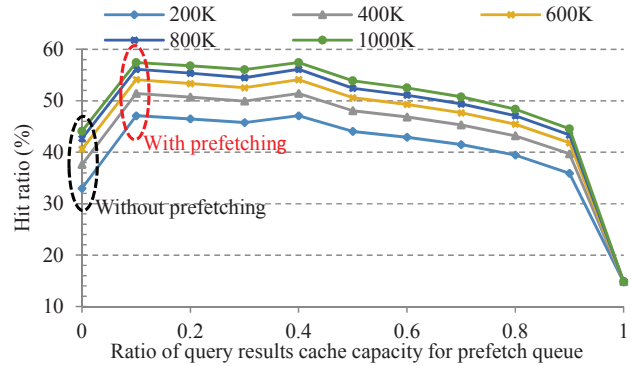


Figure 2: Hit ratio of various prefetch queue capacities

ranged from $Page_{no} + 1$ to $Page_{no} + M$ to query results cache to reduce the response time of follow-up queries. Figure 2 shows the experimental results when prefetching parameter M is 2 (Note that the results with other parameters are similar). In the figure, the horizontal axis represents the ratio of total query results cache capacity for prefetching queue and the vertical axis represents the hit ratio of the query results cache. It shows that the hit ratio of the query results cache with prefetching is better than without prefetching (the ratio of total cache for prefetching queue is 0), and prefetching query results is very necessary for the performance of the query results cache.

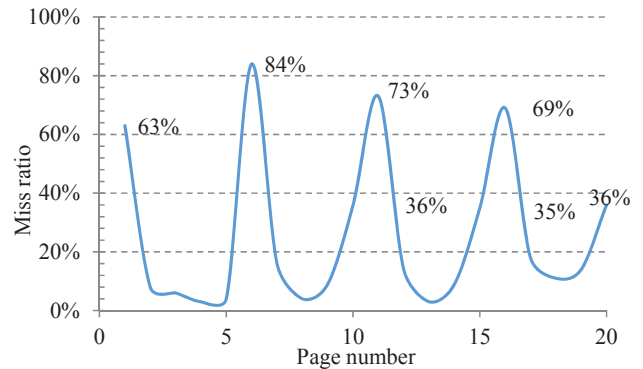


Figure 3: Miss ratio of various page numbers

From the above analysis, we can see that prefetching is an important technology to improve the Web search engine system performance. Next, we explore the characteristics of the queries submitted to the core of the Web search engine. Figure 3 shows the statistics result of cache miss page number when cache capacity is 200K query results pages and M is 4. The cache miss ratio of initial queries is 63%, because initial query usually can’t fetched in advanced, and it has high cache miss ratio. The follow-up queries page number ranged from 2 to 5 have low cache miss ratio because of prefetching. Due to prefetching parameter M is 4, query results page number 6 has high cache miss ratio. The access behavior of the subsequent query results pages follows this rule. As a result of prefetching technology using by query results cache, it makes the cache miss ratio of page numbers much different.

3. PREFETCH-AWARE ADMISSION POLICIES

3.1 Cache Admission Policy Framework

Query results cache evaluates the value of a query through cache admission policy to decide whether the corresponding query results page is infrequent or not. The approach of evaluating the value of a query is the core issue, and we define it as evaluation function. Figure 4 shows the cache admission policy framework. When a query request is received by the cache, it decides the query results page entry into the controlled cache or the un-controlled cache based on the evaluation function. When a query has a high score as evaluated by the function, the query results page is stored in the controlled cache, otherwise stored in the un-controlled cache. Controlled cache and un-controlled cache can use the same cache replacement policy, and also can use different replacement policies. Generally speaking, controlled cache only admits those queries that the evaluation function classifies as future cache hits, and uncontrolled cache admits the rest of all queries. Ideally, with a perfect admission policy, there would be no need for the uncontrolled cache.

Next, we give a formal definition of evaluation function. Given a query sequence Q_s , the definition of query evaluation function is $f : Q_s \rightarrow \{0, 1\}$, if $f(q) = 1$ then the query results page is stored in the controlled cache; if $f(q) = 0$ then the query results page is stored in the un-controlled cache.

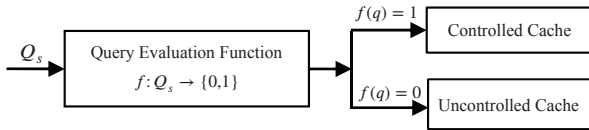


Figure 4: Query results cache admission policy framework

3.2 Evaluation Function

In the query results cache admission policy framework, choosing an effective evaluation function is the key issue. Baeza-Yates et al.[1] summarized the features those were proposed in previous studies, including the frequency of the query, the length of the query in words, the length of the query in characters and so on. We find that the prefetching technology makes the cache miss ratio of query results page numbers much different. Therefore, we choose prefetching information as the feature of the evaluation function.

According to the behavior of prefetching in the query results cache, we propose two types of features:

Non-Memory Prefetching Feature (PAAP-NMPF): It needn't maintain the information of past queries, and the decision of the evaluation function is based on current state. This is a stateless feature. The evaluation function with non-memory feature is as follows:

$$f(q) = \begin{cases} 0 & \text{if } q \text{ is hit in the prefetching queue,} \\ 1 & \text{else.} \end{cases}$$

Memory Prefetching Feature (PAAP-MPF): It needs maintain the information of past queries, and the decision of the evaluation function is based on historical information. Memory feature cost much more memory space than non-memory feature. This is a stateful feature. We define the

total number of the past queries for results page number $Page_i$ as $S(Page_i)$ and the total number of the past queries for $Page_i$ through prefetching as $P(Page_i)$. Therefore, the ratio of $Page_i$ through prefetching is:

$$Ratio_{Page_i} = P(Page_i)/S(Page_i)$$

The evaluation function with memory feature is as follows:

$$f(q) = \begin{cases} 0 & \text{if the } Ratio_{Page_i} \text{ of } q \text{ is greater than 0.5,} \\ 1 & \text{else.} \end{cases}$$

3.3 Demote Operation

Next, we will discuss the impact of demote operation on queries results cache. **Demote operation:** it is used to transfer evicted query results pages from the controlled cache to the uncontrolled cache rather than out of the query results cache directly. The motivations of demote operation is as follows: making those queries that the evaluation function classifies as future cache hits stay in the cache longer. Table 1 shows the impact of demote operation on the query results cache with PAAP-NMPF admission policy. The hit ratio of the query results cache with demote operation is higher than without demote operation.

Table 1: Impact of demote operation on query results cache

Capacity	Non-Demote(%)	Demote(%)	Improve(%)
200K	51.96	52.22	0.26
400K	55.99	56.28	0.29
600K	58.46	58.75	0.29
800K	60.02	60.26	0.24
1000K	61.10	61.32	0.22

4. EXPERIMENTAL RESULTS

4.1 Dataset

The dataset is from a famous Chinese Web search engine from 09/01/2006 to 10/31/2006. We chose a period of query logs that contained 5 million queries. The first 1 million queries constitute the training set to warm up the cache. The remaining about 4 million queries are reserved as the test set.

4.2 Evaluation Method

Query results cache is a key technology to improve Web search engine system performance. It usually uses cache hit ratio as its measure. Due to the capacity limitation of the query results cache, capacity factor need to be considered. When the cache is full, it needs a policy to replace some of the query results in the cache. There are many replacement policies, such as LRU, LIRS, 2Q, MQ and so on. LRU was adopted in our experiments, and the impact of various replacement policies is our future work.

In order to evaluate the effectiveness of our admission policies, we used the state-of-arts policies as the baseline, including the past frequency of the query (PastF), the length of the query in words (LenW) and the length of the query in characters (LenC). We used least recent used (LRU) as query results cache replacement policy and pre-SDC[3, 4] as query results prefetching policy.

4.3 Results and Analysis

We conducted our experiments on the query results cache with the prefetch-aware admission policies: (1) PAAP-NMPF and (2) PAAP-MPF. Cache capacity is given as the number of query results pages that are cached. The capacity of the cache is from 200K query results pages to 1000K query results pages, and the number of prefetching query results pages (parameter M in pre-SDC) is 2 in our experiments. It is noteworthy that the experimental results with other M values are similar, therefore, we would not discuss the impact of the parameter M . We allocate a γ fraction of the query results cache for the controlled cache and the rest for the un-controlled cache. The experiments were conducted with $\gamma = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$ and 1.0 .

First of all, we give the experimental results of the baseline such as admission policy with PastF, LenW and LenC in the paper. Table 2~4 show the results when the capacity of the cache is 800K query results pages. We found similar results to the previous studies [2]: the hit ratio of the admission policy with PastF is higher than the hit ratio of the admission policy with LenW and LenC. Table 2 shows the results of the admission policy with PastF, and optimal cache hit ratio is 59.35% when the threshold of the query frequency is 3. Because most of the queries in our Web search engine logs are in Chinese, we used a Chinese word segmentation tool – ICTCLAS released by Institute of Computing Technology, Chinese Academy of Sciences to preprocess the query logs. Table 3 shows the results of the admission policy with LenW, and optimal cache hit ratio is 56.56% when the threshold of the query length in words is 5. Table 4 shows the results of the admission policy with LenC, and optimal cache hit ratio is 56.97% when the threshold of the query length in characters is 25.

Table 2: Cache hit ratio with PastF admission policy

PastF	1	2	3	4	5
HitRatio(%)	59.24	59.32	59.35	59.22	59.17
PastF	6	7	8	9	10
HitRatio(%)	59.22	59.19	59.17	59.25	59.23

Table 3: Cache hit ratio with LenW admission policy

LenW	1	2	3	4	5
HitRatio(%)	56.35	56.33	56.52	56.53	56.56
LenW	6	7	8	9	10
HitRatio(%)	56.41	56.16	56.25	56.31	56.33

Table 4: Cache hit ratio with LenC admission policy

LenC	5	10	15	20	25
HitRatio(%)	55.98	56.01	56.44	56.89	56.97
LenC	30	35	40	45	50
HitRatio(%)	56.78	56.69	56.72	56.39	56.45

Figure 5 is the comparison of the query results cache with the PAAP-NMPF, PAAP-MPF admission policies and traditional policies. As can be seen from the experimental results, the PAAP-NMPF and PAAP-MPF admission policies are much better than those traditional policies. Compared with the LenC and LenW admission policies, the PAAP-NMPF admission policy can get 6.38%~11.99% performance increase. Compared with the PastF admission policy, the MPF admission policy can get 7.51%~9.93% performance increase. Compared with the PAAP-NMPF admission policy, the PAAP-MPF admission policy can get 4.44%~5.95%

performance increase. As can be seen from the above analysis, the hit ratio of the PAAP-MPF admission policy is higher than the PAAP-NMPF admission policy, but the PAAP-MPF admission policy uses more storage space to record the information of query results pages. According to a previous study [5], the queries whose request page number less than 10 account for 93.59%, so the storage overhead is as follows: $(number\ of\ queries) * (10\ pages) * (storage\ space\ per\ query, usually\ less\ than\ 16\ Bytes)$.

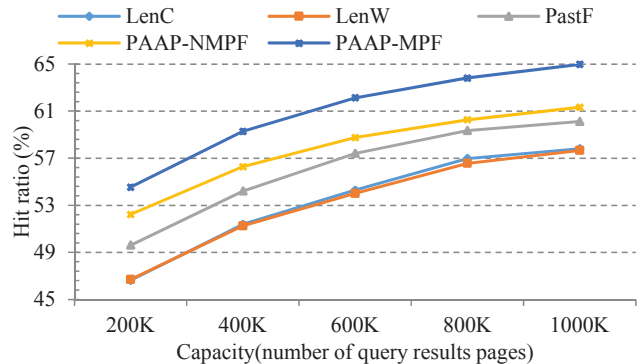


Figure 5: Comparison of the hit ratios for all requests

5. CONCLUSIONS

In this study, we introduce prefetch-aware admission policies for query results cache. Experimental results show that the new admission policies can effectively improve the performance of Web search engines. The future work involves the impact of various query results cache replacement policies and query results prefetching approaches on these admission policies.

6. REFERENCES

- [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] R. Baeza-Yates, F. Junqueira, V. Plachouras and H.F. Witschel. Admission Policies for Caches of Search Engine Results. In *Proceedings of the 14th String Processing and Information Retrieval Conference*. SPIRE'07. Springer.
- [3] 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 (Jan. 2006), 51-78.
- [4] H.Y. Ma and B. Wang. User-aware caching and prefetching query results in web search engines. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*. SIGIR'12. ACM.
- [5] Y.N. Li, S. Zhang, B. Wang and J.T. Li. Characteristics of Chinese web searching: A large-scale analysis of Chinese query logs. *Journal of Computational Information Systems*, 4(3):1127-1136, 2008.