A Top-K-Based Cache Model for Deep Web Query (Work-in-Progress)

Yue Kou, Derong Shen, Ge Yu,Tiezheng Nie
School of Information Sci. & Eng
Northeastern University
Shenyang 110004, China
0086-24-83687776

Dong Li
Business Software Division
Neusoft Group Ltd.
Shenyang 110179, China
0086-24-83662207

{kouyue, shenderong, yuge, nietiezheng}@ise.neu.edu.cn

lidong@neusoft.com

ABSTRACT
In this paper we focus on providing a cache model based on Top-K data source instead of expatiatory result records for deep web query. By integrating techniques from IR and Top-K, a data reorganization strategy is presented. Also some measures about cache management and optimization are proposed to improve the performances of cache effectively. Results from the simulation show the proposed cache model significantly improves query performance when compared with various alternate strategies.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Information filtering, Retrieval model, Search process

General Terms
Management, Design, Experimentation, Theory.

Keywords
Cache Model, IR, Top-K, .Data Reorganization, Deep Web.

1. INTRODUCTION
Caching is an important technique to enhance the efficiency of query processing. Unfortunately, most caching mechanisms which directly regard the returned results as cached data [1, 2] are not efficient for deep web [3] because of the cache memories' limitations and data's changeability dynamically [4]. The motivation lies in two aspects: A new cache structure suitable for deep web query should be defined. Some optimization measures should be carried out to improve the performances of the cache. We focus on providing a Top-K-based cache model for deep web query. Through analyzing feasibilities, an effective cache structure is defined. By integrating techniques from IR and Top-K, a data reorganization strategy is presented. Some measures about cache management and optimization are proposed to improve the cache performance.

Infoscale 2007, June 6-8, 2007, Suzhou, China.

2. CACHED DATA REORGANIZATION
For deep web, integrated query results from different data sources are not fit for serving as cached data due to their expensive space cost and dynamic nature. Thus the returned records from deep web should be reorganized into an effective structure. A data source selection strategy will be performed to identify the most matching data sources to the query request. Our goal is to select the most representative data sources to replace the result set as cached data. In order to transform the original result set to data sources, query answers should be reorganized into a suitable cache structure as defined in Definition 1.

Definition 1 [Cache Structure] Cache is essentially a hash table where an entry consists a (key, value) pair. The key is the description of a query with the form {k1, …,kn}. The value is a set of answers that satisfy the key, which is represented by data sources denoted as {<ds1, score1>,...,<dsk, scorek>}.

During data source selection there are two kinds of scoring functions representing record score and data source score respectively. Firstly based on *tf*idf* function in IR, the record score is measured. Secondly, data source score is aggregated by the scores of records provided by the data source. Due to the hugeness of result set, it is difficult to identify data sources' precise scores. Thus intervals are used to evaluate data sources' scores. With the increase of considered records in the process of data source selection, data sources' score intervals with lower and upper bounds are tighten and refined gradually.

Top-K technique is applied to the process of data reorganization to identify the best data sources as early as possible. The efficiency of Top-K data source selection relies on using intermediate answer scores in order to select relevant matches as early as possible. The Top-K algorithm consists of two phases: evaluation and post-process.

For an n-keywords query, the process of evaluation is an n-dimensional Top-K selection by scanning n ranked record lists in descending score orders. Via scanning the lists, the score intervals of current data sources are calculated and the candidates are collected. Not all records in lists must be considered. Once exist a data source whose upper bound is lower than min-k, then the

Acknowledgement: This work was partly supported by the National Science Foundation (60673139, 60473073 and 60573090) and the National Ministry of Education Project of China (GFA060448).

termination condition is satisfied. It means unseen records in lists will not defeat the current answers. Owning to adopting this early termination strategy, the efficiency of evaluation is improved remarkably, especially when record lists are very long.

The other phase prunes and refines these candidates furthermore. Data sources in candidate set should be further selected as the final Top-K based on their precise scores but not on their intervals. In general, the precise scores of all candidates should be calculated and compared together. In this paper, in order to reduce the execution cost, not all candidates but the minimum possible ones from them need to be computed precisely. As a boundary, min-k divides the candidate set into two sections: Temporary top-K (TK) and Temporary Non-top-K (TNK). TK contains the candidates whose scores are not less than min-k. Other data sources in candidate set constitute TNK. The goal of post-process is to identify whether a current data source in TK can possibly be defeated by data sources in TNK.

3. CACHE MANAGEMENT AND OPTIMIZATON

The scores calculated during data source selection can reflect uses' preferences from some extents. The higher scores, the more precision of matching will be. So we can use these scores to measure the matching precision. By integrating access frequency and matching precision, some extensions based on LRU [5] strategy are performed. Cached data which is the least recently used is evicted superiorly. When there are some cached data with the same access frequency, the scores of them are compared further and the one with least score is evicted superiorly.

Intuitively, if two queries owning similar answers, the queries themselves are also similar essentially. Thus we can calculate the similarities of these answers to estimate the relations of queries further. As for similar queries, they should be clustered together to eliminate the redundancy of cache. And also by clustering the query requests aiming at uses' preferences will benefit personalize web search. If the similarity is higher than the threshold defined in advance, the new query and the existing cached query will be considered as similar queries in semantics and will be merged into a single cached record. This can relieve the cache memories' limitations to some extend.

4. PERFORMANCE EVALUATION

We use a data set including 10^4 records as result set generated by WISE-Integrator [6] which is an automatic search interface extraction and integration tool.

The execution time of three different strategies during data reorganization are compared: without early termination strategy performed (NET), with early termination strategy performed (ET) and with pruning based on ET performed (ET&P). Figure 1 shows the execution times of them for different scales of data set.

To validate the stability of our cache model, we randomly invalidate some records in result set with different invalidation rates. By comparing the Top-K set stored in cache with the current standard answers, the query precision can be calculated. As can be seen in Figure 2, the query precision of our cache model is better even when the invalidation rate reaches 40%.

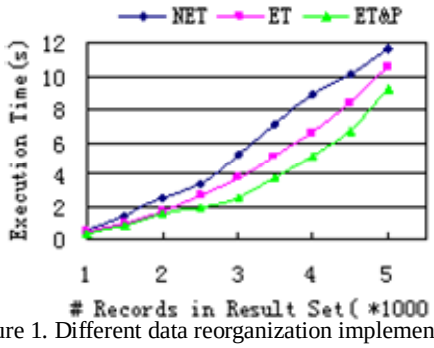

Figure 1. Different data reorganization implementations

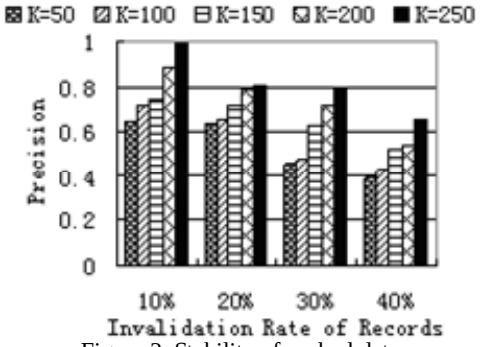

Figure 2. Stability of cached data

5. CONLUSION

In this paper we develop a research into cache modeling for deep web query. Experiments show that the proposed cache model is very useful and promising. Our future work is to make a further research on the efficiency of cache replacement and the intelligence of caching in semantics.

6. REFERENCES

[1] Florescu, D., Yagoub, K., Valduriez, P. WEAVE: a Data-Intensive Web Site Management System. *Proceedings of EDBT*, 2000

[2] Anton, J., Jacobs, L., Liu, X. Web Caching for Database Applications with Oracle Web Cache. *Proceedings of SIGMOD*, 2002, 594-599.

[3] Fetterly, D., Manasse, M., Najork, M. A Large-scale Study of the Evolution of Web Pages. *Proceedings of World Wide Web*, 2004, 669-678.

[4] Yagoub, K., Florescu, D., Issarny, V. Caching Strategies for Data-Intensive Web Sites. *Proceedings of VLDB*, 2000, 188-199.

[5] Megiddo, N., Modha, D. Outperforming LRU with an Adaptive Replacement Cache. IEEE Computer Society, 37(4) (2004), 58-65.

[6] He, H., Meng,W.Y., Yu, C. WISE-Integrator: A System for Extracting and Integrating Complex Web Search Interfaces of the Deep Web. *Proceedings of VLDB*, 2005, 1314-1317