The Update Strategies of Global Statistics in Distributed Information Retrieval Systems

Lijing WANG[1,2], Xinbo SONG[1†], Yanlong TAN[1,2], Shuai NIU[1,2], Yao CUI[1,2]

[1]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190*

[2]*Gruduate School of Chinese Academy of Sciences, Beijing, 100149*

Abstract

In distributed information retrieval systems, large-scale datasets are distributed in different sites. If term weights oriented retrieval models are used, these sites must exchange local statistics to update global statistics. As documents are added dynamically, local statistics is constantly changing. Information is exchanged frequently in order to maintain retrieval effectiveness. Great communication cost caused by this will result in low retrieval efficiency. In this paper, three updating strategies are proposed: Static, Semi-static and Dynamic Strategy. They consider both effectiveness and efficiency. In Static Strategy, estimates of global statistics are used. Two methods are stated. One is L-static Strategy which uses local statistics as an estimate of global statistics; the other is H-static Strategy which uses historical statistics. In Semi-static Strategy, updating occurs at a fixed time cycle. In Dynamic Strategy, global statistics are updated partially according to the rate of change of global statistics. Proper distributed environment is simulated with two real-world collections. Experiment results show that: a) Semi-static Strategy achieves retrieval effectiveness comparable to a centralized retrieval system. But communication cost is the highest among three strategies. b) Static Strategy has no communication cost, but it obtains retrieval effectiveness that is the worst among three strategies. Historical statistics is more similar to global statistics than local statistics is. c) Dynamic Strategy is a balancing strategy since its retrieval effectiveness is much better than Static Strategy and its communication cost is lower than Semi-static Strategy. How the varying global statistics impacts retrieval effectiveness is also discussed in this paper.

Keywords: Global Statistics; L-static Strategy; H-static Strategy; Semi-static Strategy; Dynamic Strategy; Communication Cost

1. Introduction

As the online information grows rapidly, the information environment is highly distributed and dynamic. Documents are stored and indexed in distributed sites. One of the great challenges raised by this environment is how to search for useful information in such vast distributed documents. In today's IR systems, most popular information retrieval models, like the vector space model (VSM) [4] or the probabilistic model [9], consider the global term statistics, e.g. the inverse document frequency (*idf*), for relevance weighting. The use of global statistics in retrieval models is beneficial to retrieval effectiveness. However, distributed sites only have local statistics that makes the intermediate results incomparable. They must exchange local statistics to generate global statistics. Since local statistics are constantly changing as new documents are added, distributed sites must frequently exchange local information to maintain retrieval effectiveness. However, frequent exchange between sites may cause network congestion that declines efficiency of IR system. Therefore, new strategy is needed to consider both effectiveness and efficiency.

In this paper we propose three updating strategies: Static, Semi-static and Dynamic Strategy. Our contributions are:

† Corresponding author.
Email addresses: songxb@ict.ac.cn (Xinbo SONG).

1) Valid updating strategies are proposed in this paper. It is proved that Semi-static Strategy and Dynamic Strategy greatly decrease communication cost. And they do not cause serious loss of retrieval effectiveness.

2) A finding that using local statistics as an estimate of global statistics, it gets better search results when documents are divided evenly than divided by timestamp.

3) A finding that all global statistics used in retrieval models should be considered equally. Otherwise, it will have opposite effect on retrieval effectiveness.

This paper is organized as follows. Section 2 reviews background and related work on updating global statistics. Section 3 introduces three updating strategies, followed by the experimental evaluation and discussion in section 4. The conclusion is given in section 5.

2. Background and Related Work

In this section, first two retrieval models (VSM and BM25) which used in our experiment are introduced here. Then a popular measure of retrieval effectiveness NDCG is presented. Next, definition of real-time updating is given. Last, we review some related work on updating global statistics.

2.1. Retrieval Models

Two retrieval models are introduced here: simple VSM model and BM25 model [4]. Both of them use inverse document frequency (*idf*) and are currently among the most popular and effective IR models.

SVSM (short for simple VSM model) bases on Vector Space Model [4]. It computes the relevance score of a document D for a query Q by the following formula:

$$R(D,Q) = (\sum_{i=1}^{n} (\log tf_i + 1.0) \cdot \log(\frac{N}{df})) \cdot (\frac{1}{\sqrt{length}}) \tag{1}$$

where tf_i is the term frequency; N is the total number of a collection; df_i is the document frequency; *length* is the document length (the number of terms contained in document D).

BM25 computes the relevance score of a document D for a query Q by the following formula:

$$R(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{(k_1 + 1)TF(q_i, D)}{TF(q_i, D) + k_1(1 - b + b \cdot \frac{|L|}{avgL})} \tag{2}$$

where TF is term frequency; L is the document length; $avgL$ is the average document length; k1 and b are free parameters, usually chosen as $k_1 \in [1.2, 2.0]$ and $b = 0.75$; IDF is the *idf* factor, which is given by:

$$IDF(q_i) = \log \frac{N - df + 0.5}{df + 0.5} \tag{3}$$

where N is the total number of documents in the collection, and df is document frequency.

2.2. Evaluation Measure

The NDCG measure has proven to be a popular measure of retrieval effectiveness utilizing graded relevance judgments. Assume *R(j, d)* is the ideal relevance score (in our experiment, *R(j, d)* is document score in centralized retrieval system) of document d for query j. The NDCG for top k results is defined as:

$$NDCG(Q,k) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} Z_{j,k} \sum_{d=1}^{k} \frac{2^{R(j,d)} - 1}{\log(1 + d)} \tag{5}$$

where $Z_{j,k}$ is the normalization factor, which ensures the NDCG for query j is 1; d is document rank in

distributed system.

2.3. Real-time Updating

Distributed IR systems should concern about how to obtain valid global statistics while using global term statistics oriented retrieval models. In most distributed IR systems, distributed sites must exchange local statistics to generate global statistics. In order to maintain retrieval effectiveness, exchange of local information occurs once new documents are added to any of these sites. This frequent exchange of local information is called real-time updating. It may cause network congestion that may greatly declines efficiency of IR system.

2.4. Related Work

There have existed some researches on the updating of global statistics. Harman et al. [5] described a prototype distributed IR system where data was stored centrally but maintained in separate datasets organized by content. Searches could span multiple datasets kept at multiple locations, but any single datasets was never divided. Thus there was no updating of global statistics.

Viles [12] described a method for maintaining global statistics in a distributed IR system. It was redefined as Dedicated-indexer system by Melink, S. and S. Raghavan [7]. The design of Dedicated-indexer topology is showed in Fig.1.

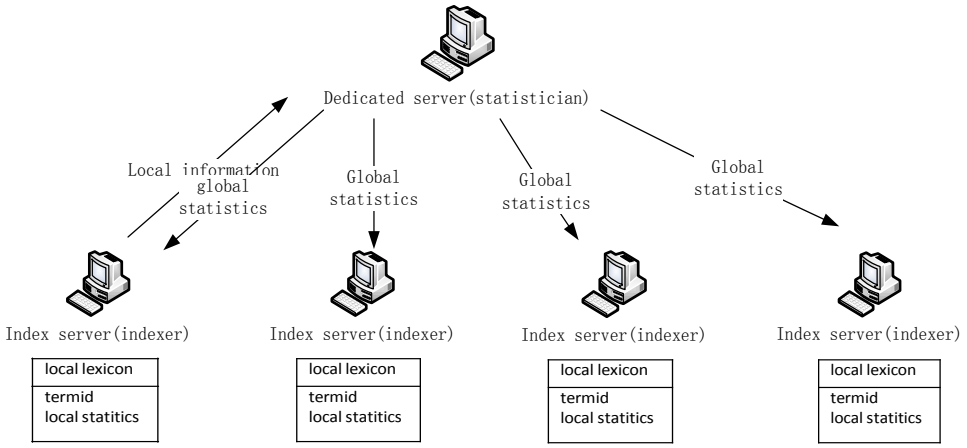


Fig. 1 Dedicated-indexer Topology

Dedicated server known as statistician minimizes the number of conversations among servers, since indexers exchange statistical data with only one statistician. However, real-time updating under this topology also leads to large-scale global statistics exchange between the statistician and indexers.

Aalbersberg and Sijstermans [1] used the Vector Space Model [4] as the IR model in the Parallel InfoGuide system. Term weights were applied to the query terms merely. This limited the kinds of term weighting functions that might be used by the system.

In Viles and French [13], it was found that full dissemination was not necessary and that the level of dissemination needed depended on the degree of randomness applied when allocating documents to databases. However, tests with larger collections were needed and further exam for non-random document allocation was also needed.

Witschel [15] showed that a very good retrieval performance can be reached when just the most frequent terms of a collection were known and all terms which were not in that list were treated equally. However, the list could not always be fully estimated from a general-purpose reference corpus. Other great researches have been represented in [6, 7, 8, 14].

3. Updating Strategies

In this section, three strategies are described in details. Static Strategy and Semi-static Strategy can be used in any distributed environment. Dynamic Strategy is suitable for Dedicated-indexer environment.

3.1 Static Strategy

Estimates of global statistics are used in Static Strategy. Thus, no information exchange happens between distributed sites. Two ways proposed to estimate global statistics in Static Strategy.

1) Local statistics as an estimate of global statistics (L-static strategy).

2) Large-scale historical statistics as an estimate of global statistics (H-static strategy).

Static Strategy has no extra information exchange and no extra storage. One challenge of this strategy is to decide whether the historical statistics is appropriate and what kind of collections is appropriate for generating historical statistics.

3.2 Semi-static Strategy

In Semi-static strategy, updating occurs at a fixed time cycle (FTC). Assume that real-time updating updates once a minute. Our Semi-static Strategy updates once a month. Then the communication cost of Semi-static Strategy is negligible compared with real-time updating.

This strategy is based on the theory that only a certain amount of time of information gets together can make a convincing representative of the information change. We also consider the initial time when there is no information at all or the information is too small to generate *idf*. It is pratical to use historical global statistics or to make real-time update at the beginning period of time since *idf* values converge rather quickly [2].

3.3 Dynamic Strategy

As new documents are added, updating of index leads to a change of *idf* of some terms. In practical environment, a slight change on *idf* has little effect on retrieval effectiveness. Thus only updating those *idf*s that have changed too much to maintain the retrieval effectiveness will greatly reduce the size of information exchanged. Our dynamic strategy is a method of partial updating. An extra server is necessary to storage a global lexicon containing global statistics and all local lexicons containing local statistics. Thus Dynamic Strategy is fit for Dedicated-indexer system which we mentioned in 2.3.section very well.

Assumed there are N indexers and one statistician. Each indexer maintains a local lexicon that only stores those terms occurring in the local inverted file. The statistician maintains N local lexicons {Llex1…N} coming from indexers and one global lexicon (Glex) which is the merge result of N local lexicons. Our dynamic strategy is as follows:

Dynamic Strategy Algorithm

1) Indexer who updates its local lexicon must send a local updated list to the statistician. The list contains those terms whose *idf*s have changed.

2) After receiving local updated lists, the statistician merges these lists and updates Glex.

3) The statistician computes the rate of change (R) between local *idf* and global *idf*. The rate of change of term t from Llex_i is defined as:

$$R(t,i) = \left| \frac{(idf - idf_i)}{idf} \right| = \left| \frac{idf_i}{idf} - 1 \right| \tag{4}$$

where $idf = N/df$; idf_i comes from $Llex_i$; idf comes from Glex.

4) If R goes beyond the threshold T, the statistician must send global idf to corresponding indexer and also update corresponding $Llex_i$.

The threshold T is an experimental value. A deep discussion is made in this paper. Next section, three strategies are tested on real datasets. Discussion on experiment results is followed.

4. Experiments

In this section, a measure of evaluating retrieval efficiency is defined in 4.1.section. Then real collections and experimental parameters that are used are displayed in 4.2.section. In 4.3.section, results are presented and discussed in details.

4.1. Efficiency Evaluation

In this paper, NDCG measure is considered to be the most proper measure of evaluating effectiveness because it uses graded relevance judgments. We have introduced NDCG measure in 2.2.section.

In our experiments, communication cost is evaluated by information size transferred during one updating process. It is defined as:

$$C = \frac{B\sum_{i=1}^{I} S_i}{TC} \tag{6}$$

where I is the number of indexers; S_i is the number of IDFs needed to be update; TC is the average time cycle of inverted list updating in indexers; B represents the size of one single posting (*termid, N, df*).

4.2. Experimental Setup

SVSM model and BM25 model are both tested in this paper. Using two different retrieval functions also ensures that results are not artifacts of a particular weighting scheme. Our experiments are carried out with one real-world query collection: SogouQ, and two real-world news collections: SogouCS and SogouCA. Collection description displays in Table 1.

Table 1 Collection Description

Collection	Description	#Documents	Form of Documents	Topic
SogouCS	Sohu news data (Jan. – Jun. 2008)	2053448	URL and full text information	From 18 channels of Olympics, sports, IT, domestic, international and so on.
SogouCA	the whole network news data (May – Jun. 2008)	1411646	URL and full text information	From 18 channels of Olympics, sports, IT, domestic, international and so on.
SogouQ	User query log (June, 2008)	51537390	access time/user ID/full text query	

In order to simulate the real distributed environment, SogouCS is divided into 100 parts by Round-robin order (R order) and by plain sequence of timestamp order (P order). Our L-static strategy is tested on both orders. The merging collection of SogouCS and SogouCA is treated as the historical collection of SogouCS. Global statistics of historical collection is then counted. It will be used for H-static strategy.

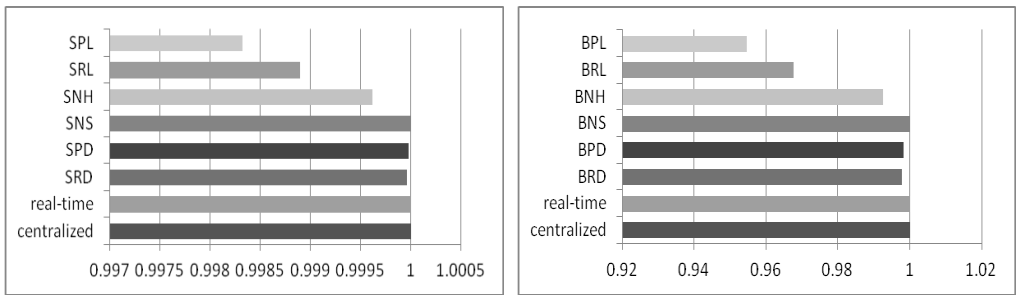
SogouCS is divided into six sub collections according to timestamp of news time. Semi-static strategy is tested on these sub collections. In our experiment, FTC is set to one month.

Dynamic strategy is implemented on simulated environment (both R order and P order). Retrieval effectiveness is tested with T varies from 0.0 to 1.0. Results of each condition are discussed in the following section.

Parameters used in our experiments are displayed in Table 2.

4.3. Experimental Results

Figure 2 presents NDCG under different strategies. Figure in left (using SVSM) indicates that Semi-static Strategy (FTC = one month) achieves retrieval effectiveness (NDCG = 0.999999) comparable to a centralized retrieval system. Dynamic Strategy obtains an excellent retrieval result next to Semi-static Strategy's. Using Static Strategy, the results are unsatisfactory. H-static Strategy obtains NDCG that is better than L-static Strategy. Using L-static strategy, it gets better search result when documents are divided evenly than divided by timestamp. The same conclusion is made when using BM25 (right figure). This indicates that retrieval models do not affect our strategies.



a) SVSM b) BM25

Fig. 2 NDCG for Different Strategies Using SVSM and BM25 (Annotation: S (simple VSM), B (BM25); P (P order), R (R order), N (no division); L (L-static), H (H-static), S (Semi-static), D (Dynamic). e.g. SPL is short for SVSM + P order + L-static)

NDCG of strategies using SVSM and BM25 are compared in Fig.3. It shows that using SVSM gets better effectiveness than BM25 in our experiments. That's might because there are two global statistics (idf, avgL) in BM25 formula, but only idf is taken into consideration. In practical environment, all global statistics should be concerned equally.

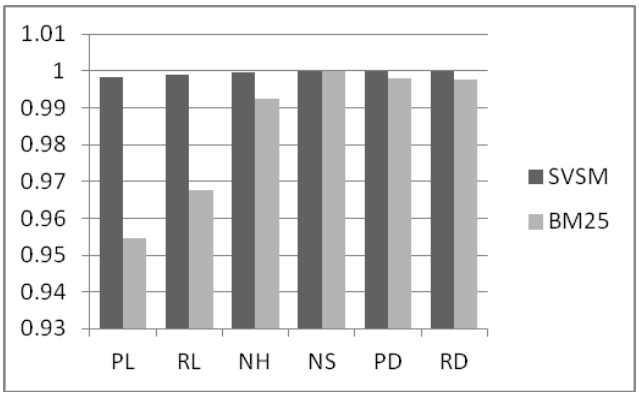


Fig. 3 Difference in Retrieval Effectiveness when using SVSM and BM25

Table 3 presents communication cost of each strategy using formula (6). From Table 3 we know that Static Strategy has no communication cost. The cost of Dynamic Strategy reduces approximately 100% compared with real-time updating. Such a little cost is negligible. And Semi-static Strategy causes the

highest communication cost among three strategies. However, a 99.72% decrease in communication cost is gratifying.

Table 2 Selected Parameters for Experiment

name	value
Q (number of queries)	100
k (top k results)	100
k1 (parameter in BM25)	2
b (parameter in BM25)	0.75
FTC (fixed time cycle)	1month
T (threshold of variation of idf)	0.05
I (number of indexers)	100
B (size of one posting)	16 Bytes
TC (real-time updating cycle)	2 hour

Table 3 Communication Cost of Different Strategies

updating strategy	communication cost(kb/s) / reduce%
centralized	0
real-time	145.64
SPL / BPL	0 / 100
SRL / BRL	0 / 100
SNH / BNH	0 / 100
SNS / BNS	0.41 / 99.72
SPD / BPD	0.02 / 99.99
SRD / BRD	0.02 / 99.99

Figure 4 gives the relationship between communication cost and retrieval effectiveness over varying T. SVSM is used and documents are distributed in R order. Similar results are received when using BM25 or P order. Since values of NDCG are too close to each other, we make a transformation for these values. New NDCG = -log (1.000001-NDCG). The vertical axis represents communication cost (KB), and the horizontal axis represents new NDCG values. It is easy to conclude that the smaller T is the greater communication cost is. Effectiveness of search results increases as T decreases. When T <= 0.1, an obvious increase in effectiveness is occurred. But when T <= 0.05, the growth rate in communication cost is very fast. Thus, the range of T is identified as [0.05, 0.1] in our distributed environment. Range of T will drift on different collections. Researchers should set threshold T base on practical demand.

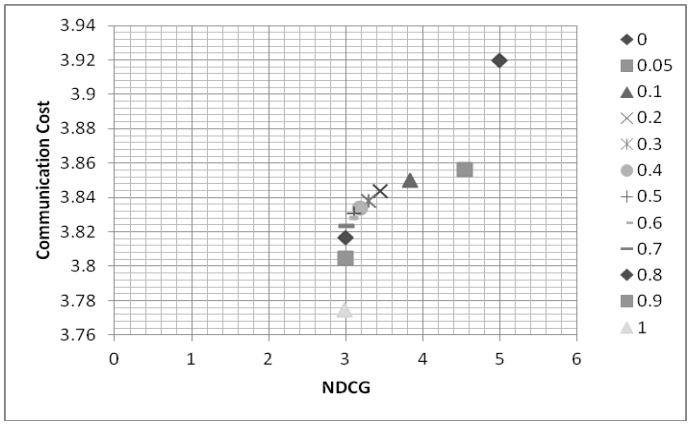


Fig. 4 Retrieval Effectiveness and Communication Cost with Varying Levels of T

5. Conclusion and Future Work

In this paper, we propose three updating strategies: Static, Semi-static and Dynamic Strategy. Efficiency of IR system receives the same attention as effectiveness in our strategies. Experiment results show that:

1) Static Strategy has no communication cost, but it obtains retrieval effectiveness that is the worst among three strategies. Thus if a system has very strict efficiency requirements, but not too strict demands on accuracy, then Static strategy is a good choice.

2) Semi-static Strategy achieves retrieval effectiveness comparable to a centralized retrieval system. But communication cost is the highest among three strategies. It needs neither extra computational service nor extra storage space.

3) Dynamic Strategy is a balancing strategy since its retrieval effectiveness is much better than Static Strategy and its communication cost is lower than Semi-static Strategy. It is perfect for those systems which have high demand on both efficiency and effectiveness. However, a high-performance server is necessary to do computation and to store lexicons.

We see several directions for future work. All global statistics used in retrieval models should be considered equally. What kind of collections is appropriate for historical collection is also worth further studying. When using Semi-static Strategy, FTC should be tested on diversified collections to see whether there are specific values for different types of IR systems. A more dynamic, distributed environment would also be necessary to verify whether our strategies are practical.

Acknowledgement

This work is partially supported by the National High Technology Research and Development Program of China (863 Program) under grant No. 2011AA010703.

References

[1] I. J. Aalbersberg and F. Sijstermans. High-quality and High-performance Full-text Document Retrieval: The Parallel InfoGuide System. In *Proc. 1st Intl. Conf. Parallel and Distributed Information Systems*, Miami, FL, IEEE Computer Society Press: 142-150, 1991.

[2] T. Brants and F. Chen. A System for New Event Detection. In *Proceedings of SIGIR'03*, pages:330-337, 2003.

[3] A. R. Chowdhury. On the Design of Reliable Efficient Information Sysstems. *Ph.D. thesis, Illinois Institute of Technology*, 2001.

[4] G. Chowdhury. Introduction to Modern Information Retrieval, Third Edition. *Facet Publishing*, 2010.

[5] D. Harman, W. McCoy, et al. Prototyping a Distributed Informaion Retrieval System Using Statistical Ranking. *Information Processing and Management*, 27(5):449-460, 1991.

[6] A. Z. Kronfol. FASD: A Fault-tolerant, Adaptive, Scalable, Distributed Search Engine. *Master's thesis. Princeton University*, 2002.

[7] S. Melink and S. Raghavan, et al. Building a Distributed Full-text Index for the Web. *ACM Trans. Inf. Syst*. 19(3): 217-241, 2001.

[8] Z. Mazur. On a Model of Distributed Information Retrieval Systems Based on Thesauri. *Information Processing and Management*, 20(4):499-505, 1984.

[9] S. E. Robertson, S. Walker, et al. Okapi at TREC-4. In *Proc. TREC-4*, 1995.

[10] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage*, 24(5): 513-523, 1988.

[11] C. Tang, Z. Xu, M. Mahalingam. pSearch: Information Retrieval in Structured Overlays. *ACM SIGCOMM Computer Communication Review*, 33 (1): 89–94, 2003.

[12] C. L. Viles. Maintaining State in a Distributed Information Retrieval System. In *Proc. 32nd ACM Southeast Conf*, pages 157-161, Tuscaloosa, AL, March 1994.

[13] C. L. Viles and J. C. French. Dissemination of Collection Wide Information in a Distributed Information Retrieval System. *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. Seattle, Washington, United States, ACM: 12-20, 1995.

[14] C. L. Viles and J. C. French. On the Update of Term Weights in Dynamic Information Retrieval Systems. *Proceedings of the fourth international conference on Information and knowledge management*, Baltimore, Maryland, United States, ACM: 167-174, 1995.

[15] H. F. Witschel. Global Term Weights in Distributed Environments. *Inf. Process. Manage*, 44(3):1049-1061, 2008.

[16] T. W. Yan and H. Garcia-Molina. Index Structures for Selective Dissemination of Information under the Boolean Model. *ACM Trans. Database Syst*. 19(2): 332-364, 1994.

[17] Y. Yang, T. Pierce and J. Carbonell. A Study of Retrospective and On-line Event Detection. In *Proceedings of SIGIR'98*, pages:28–36, 1998.