Using linkage information to approximate the distribution of relevant documents in DIR

Georgios Paltoglou
Michail Salampasis
Maria Satratzemi
Georgios Evangelidis

University of Macedonia, gpalt@yahoo.com
Alexander Technology Educational Institute of Thessaloniki, cs1msa@it.teithe.gr
University of Macedonia, maya@uom.gr
University of Macedonia, gevan@uom.gr

Abstract

In this paper, a method for addressing the source selection problem in a distributed information retrieval environment is presented. The method proposes a solution to the source selection problem by making use of the outlink distribution extracted from a sampling collection in order to locate the most authoritative collections for a particular query. Experiments carried out with the algorithm show that its performance exceeds that of the uniform approach, with much more economical means. Specifically, the link-based method is much more efficient in terms of collection utilization than the uniform strategy, which can be applied under the same conditions. A key feature of the algorithm is that it can be combined and work in parallel with content-based source selection algorithms.

Keywords: Research Track, web IR, distributed IR, source selection, collection fusion

1. Introduction

With the proliferation of the Web, it has become increasingly difficult for users to find relevant information to satisfy their information needs. Often, they are faced with the decision of choosing amongst several information sources in order to find the most appropriate that will provide the necessary information. A solution to this problem is provided by *search engines*, which give users a starting point in finding the information they need.

General purpose search engines, nonetheless, only offer a limited solution. They cannot index the whole of the WWW because of its prohibitive size and rate of growth and thus index only parts of it. In addition to that, a large number of specialized web sites do not allow their content to be analyzed and offer their own search capabilities. Also, a number of sites are not reachable by search engines (i.e.

invisible web) and lastly, there is a vast amount of corporate knowledge accumulated in privately owned and managed networks that remain outside their reach. Thus, a user posing a query to a particular general purpose search engine may be missing on relevant and valuable information.

Distributed Information Retrieval (DIR) (also known as collection fusion) offers a solution to the above problems by offering the user the capability of simultaneously searching remote document collections (i.e. search engines or specialized web sites) through a single interface. The challenge posed by DIR is on how to combine the results from multiple, independent, heterogeneous document collections into a single merged result in such a fashion that the effectiveness of the combination approximates or even surpasses the effectiveness of searching the entire set of documents as a single collection. The process can be perceived as two separate but interleaved sub-processes. *Source selection*, in which a subset of the available information servers is chosen and *results merging*, in which the separate results are combined into a single merged result which is presented to the user.

Collection fusion strategies can be classified using two criteria. The first is based on whether or not there is a need of a training phase before the algorithm can be utilized. In general, algorithms which require training may discriminate better amongst information servers [Voorhess et al (1994)]. On the other hand, they may not be particularly adaptable to volatile and dynamic environments, such as the WWW. The second criterion is based on the amount of information the algorithm receives from the information servers. Algorithms that use only the returned list of documents with or without relevance scores are called *isolated*. Algorithms which rely on the propagation of additional information (i.e. collection size, term or document frequencies) are called *integrated.* Integrated algorithms are able to perform more adequately than isolated, due to the additional information available to them.

The link-based source selection algorithm that is proposed here makes use only of the returned list of documents, without the need for relevance scores. Thus, it is an isolated algorithm with no training phase necessary.

The rest of this paper is divided as follows. Section 2 reports on prior work. Section 3 describes the proposed link-based source selection algorithm. Section 4 describes the setup of the experiments conducted and section 5 reports and discusses those results. Lastly, section 6 concludes the paper, summarizing the findings.

2. Prior Work

A number of source selection strategies have been proposed in recent years. Voorhees, [Voorhees et al (1994)] developed two isolated algorithms that require training. MRDD utilizes training queries to build a model for the distribution of relevant documents on available servers. The Query Clustering algorithm clusters

training queries based on the common documents retrieved while in parallel learns the effectiveness of the queries in each cluster.

The CORI algorithm [Callan et al (1995)] is based on inference networks. It is an integrated algorithm in which every source is regarded as the surrogate of the documents that belong to it. It is one of the most stable algorithms in terms of performance. The vGlOSS [Gravano et al (1999)] algorithm is based on the vector space model. The representation of each server is based on the normalized document vectors as well as its document frequency statistics. The choice of servers is based on the similarity of each posed query with the vector of each server. CVV [Craswell et al (2000)] is based on the cue validity variance of query terms. Terms which discriminate better amongst servers have a higher CVV and thus contribute more to the score of each server. ReDDE [Luo et al (2003)] estimate the distribution of relevant documents across the available databases and ranks the databases accordingly. It utilizes both content similarity measurements as well as database size.

Most state-of-the-art isolated collection fusion algorithms utilize some sort of sampling collection [Callan et al (2001), Luo et al (2003)] in order to select the most appropriate document collections relevant to a particular query. These are needed in order to extract certain statistical information concerning their contents (i.e. terms, frequencies, size etc) in isolated environments.

In parallel to research in distributed information retrieval (DIR), there has also been much research on the utilization of linkage information amongst hypertext documents in centralized information retrieval (PageRank [Page et al (1998)], HITS etc). Kleinberg in [Kleinberg (1997)] introduced the notion of "authoritative" information sources (i.e. web pages) on specific topics. In his work, Kleinberg utilizes hypertext links in order to discover information sources which are considered the most authoritative on a particular subject.

The work presented in this paper combines the notion of authoritativeness found in Kleinberg with the work found in [Salampasis et al (1999)] that also exploits links amongst hypertext documents in order to solve the source selection problem. In that work, particular emphasis was given on the issue with regard to Hypermedia Digital Libraries (HDL) where bibliographic information (i.e. citations, co citations and bibliographic couplings) was utilized as linkage information. The approach presented here is more generalized, referring to hypertext web pages and linkage information found within, making the strategy applicable in the WWW environment. Another difference between the two approaches is the definition of the sampling collection. In that work, an existing collection was chosen by random to act as a sampling collection, while in our work, the sampling collection is created as descriptor of the contents of the individual collections.

One of the key features of the algorithm is that it differs from the aforementioned approaches in source selection, which evaluate available collections based on their

content. Source selection, in the framework that it is presented here, is solely based on linkage information. Because of that fact, the algorithm can be used in parallel with any of those methodologies provided that there is linkage information to be analyzed. In effect, the algorithm could be utilized to provide additional information in source evaluation and thus improve effectiveness.

3. Link – Based Source Selection

The link-based algorithm that is proposed, attempts to approximate the distribution of relevant documents amongst information servers. The proposed source selection strategy combines two features usually not found in other methods making the link-based strategy practicable and applicable to dynamic and large information seeking environments. First, it solves the source selection problem solely by the use of linkage information. Second, it does not require any training phase.

The link-based fusion strategy is based on the so-called link-hypothesis (i.e. *closely interlinked documents tend to be relevant to the same information needs*) [Salampasis et al (1999)]. It integrates the concept of authorities found in Kleinberg and applies it in a distributed environment. Specifically, the algorithm assumes that for each information need, expressed as a query Q, there are certain authoritative collections, identified by the number of relevant documents they contain. The aim of the algorithm is to locate those collections, making use of the graph topology of the Web.

The version of the algorithm presented here makes use of outbound links to locate those authoritative collections. Specifically, the algorithm operates in two separate steps. First, given an information need expressed by a query Q, an initial search is made using Q to retrieve m documents from a sampling collection C_s. The utilization of a sampling collection as mentioned above is a standard practice in most state-of-the-art distributed information retrieval algorithms and is examined in more depth in later chapters. The m documents retrieved from the sampling collection C_s are processed to extract linkage information. Second, linkage information is passed to a maximisation function which determines the distribution of relevant documents in remote collections. This approximation together with the total number of documents to be retrieved is used by the fusion method to calculate the number of documents that will be requested from each collection.

More analytically, we denote as $R_s = \{d_1, d_2, d_3, \ldots, d_m\}$ the search result of the sampling collection using query Q. We define as *Inbound score* $I(d^c)$ of any document d^c which belongs to collection c, as the number of documents returned from the sampling collection, not belonging to c, that point to it through a hypertext link. In a web environment, that limitation would translate into off-site links. This emphasis was given in order to avoid linkage information for navigational purposes (i.e. web pages pointing to the starting page) and distil linkage information relative only to

content similarity. Therefore, if $d_i \rightarrow d^c$ denotes that document d_i has a link to document d^c, then:

$$I(d^c) = \sum_{d_i \rightarrow d^c} d_i, \text{ where } d_i \notin c \text{ and } d^c \in c \quad \textbf{(1)}$$

The *inbound score* of each document is dynamic and specific to particular queries and information needs. A document can have a high score if a large number of documents retrieved from the sampling collection point to it, and a very low score if few or no documents point to it. Considering that each document belongs to a collection, *the score of Relevancy R(c)* of each particular collection c is estimated as:

$$\mathrm{R}(\mathrm{c}) = \sum_{d^c \in c} I(d^c) \quad \textbf{(2)}$$

Intuitively, we can define the *score of Relevancy* of each collection as the number of outlinks extracted from the results of the sampling collection from documents belonging to other collections that point to it. Therefore, assuming that the user requests k documents to be retrieved in total from all collections, then the total number of documents to be requested from each collection $N(c_j)$ is estimated as:

$$N(c_j) = [R(c_j) / \sum_{c_i} R(c_i)] * k \quad \textbf{(3)}$$

Effectively, when $N(c_j)=0$ then collection c_j is not requested to return any documents. As already noted, only outlinks of documents returned from the sampling collection are considered, as this was seen as a more realistic environment. Inlink information (i.e. information about the documents that point to, rather than be pointed by, the results returned from the sampling collection) could also be taken into consideration but it would make the algorithm unrealistic as it would require knowledge which is not readily available at query time.

4. Experimental Methodology

The experiments conducted in this paper are based on the WT10g collection. Specifically, a variation of the WT10g collection is used, called WT-Dense [Cathal (2003)]. The reason for this choice of testbed is the inability of the original WT10g to accurately capture the link distribution of hypertext documents on the Web and, more important for our purposes, the actual average of off-site links. This has often led to unsatisfactory results from algorithms that use linkage information for various tasks in IR [Cathal (2003)]. We believe that the WT-Dense collection alleviates these problems and produces a better representation of a real web environment. More information of WT-Dense is given in Table 1.

Three sets of experiments were conducted. We gradually increased the level of distribution in order to examine the performance of the algorithm in a range of distributed environments. In each set, the collection was divided in 30, 100 and 1000 non-intersecting collections using a content-based clustering algorithm, based on the

homepage of each server. It was imperative for the accurate simulation of a real web environment, that the collections were comprised of complete web sites (i.e. documents belonging to the same site were always in the same collection). It was also considered that the homepage is indicative of the content of the web site. That is clearly not always the case, but the decision for this approach was made in order to show that the algorithm was effective even in environments with loose intra-cluster similarity. In a web environment, these collections could represent content-providing information servers (such as Reuters or NY Times).

Table 1. Comparing WT10g and WT-Dense

Properties	**WT10g**	**WT-Dense**
Number of Documents	1,692,096	120.494
Number of off-site links	171,740	171,740
Average of off-site indegree	0.10	1.43
Number of unique servers	11,680	11,611
Generality	0.15%	0.21%
Number of TREC-9 queries with relevant documents	50	36
Average number of relevant documents per query	52	7

In each set, 4 collection fusion strategies were tested: *uniform, random, optimal* and *link-based* and 1000 documents were requested by each strategy. The uniform approach treats each collection equally and requests the same number of documents from each. If we denote as Docs(c_j) the number of documents that will be requested from collection c_j, then:

$$Docs(c_j) = 1000/(number\ of\ collections), \quad \text{for each } c_j \qquad \textbf{(4)}$$

The random algorithm requests a random number of documents from random collections until a total of 1000 documents are returned. The optimal algorithm is a retrospective algorithm and requires knowledge of the distribution of relevant documents across all collections. If Rel(c_i) is the number of relevant documents in collection c_j, then:

$$Docs(c_j) = [Rel(c_j)/\sum_{c_i} Rel(c_i)] * 1000 \qquad \textbf{(5)}$$

The results from each document collection were merged using a biased C-faced die algorithm [Voorhees et al (1994)]. The final lists produced were compared to the results produced when the WT-Dense was searched as a single collection (*single).*

The retrieval engine at the individual collections in all of the experiments was the same. The effectiveness of the IR engine in absolute terms is not important, since the aim of the experiment is to evaluate the relative effectiveness between different selection strategies. In other words, the only parameter which varied is the source

selection method and any difference in performance must be attributed only to differences in the source selection strategies.

A sampling collection was created by randomly selecting documents from each collection and analyzing their content. This sampling could be achieved by having a crawler do a random walk on each collection, indexing visited pages until some criterion was satisfied. In our experiments, we crawled the same number of pages from each collection. The link-based algorithm initially utilized for linkage extraction the 100 top results from a sampling collection, made of 20% of WT-Dense.

5. Experimental Results

5.1 Source Selection Strategies Compared

The results of the first set of experiments are reported in Table 2. Recall and precision percentages indicate differences against the single collection. *Collection Utilization* refers to the percentage of the collections that were selected to contribute documents.

One of the early conclusions that can be drawn is the fact that the link-based algorithm performs much better than the random in all settings, thus validating the hypothesis that linkage information does not select sources in a random way and indeed provides viable and valuable information in source selection. It should be stressed that the comparison with the random approach is made only to validate that the proposed method has a rationale and is based on a valid underlying principle.

One point that must be addressed is that the precision scores differ in a significant manner between the single and the distributed approaches. It is believed that these results are influenced by the results merging algorithm, a belief that was confirmed later in our experiments. The focus on this paper is primarily on source selection, which is most notably shown in recall, which better demonstrate the ability of the algorithm to adequately distinguish between important and trivial collections.

In the first set, both uniform and link-based algorithms perform similarly to the single in terms of recall. The fact that the link-based performs so closely to the single collection means that most of the relevant sources were discovered successfully. The link-based algorithm actually outperforms the recall of the uniform, with 16% less server utilization, which means that the algorithm utilized on average 25 of the 30 available collections. The optimal algorithm, although performing better, does not produce a statistically significant difference. In terms of precision, the link-based algorithm is marginally worse than the uniform.

The results follow the same pattern in the second set with the 100 collections. The link-based algorithm steadily performs better than the uniform in terms of recall, with 28% less server utilization. Both algorithms perform better than the single collection. The implications of this result are discussed below. Precision of link-based is again

almost the same as that of the uniform, while both perform significantly lower than the single collection.

Table 2. Experimental Results

Setting	Source Selection Method	Average Recall		Average Precision		Collection Utilization
30 Collections	Link-Based	0.6831[a]	-6%	0.0394[a]	-73%	84%[a]
	Optimal	0.8148[a]	+12%	0.1706[b]	+15%	16%[b]
	Random	0.1371[b]	-81%	0.0227[a]	-85%	23%[b]
	Single	0.7279[a]		0.1478[b]		NA
	Uniform	0.6543[a]	-10%	0.0554[a]	-63%	100%[c]
100 Collections	Link-Based	0.7845[a, c]	+8%	0.0420[a]	-72%	72%[a]
	Optimal	0.9382[a]	+29%	0.2780[b]	+89%	5%[b]
	Random	0.0786[b]	-89%	0.0057[a]	-96%	7%[b]
	Single	0.7279[c]		0.1478[c]		NA
	Uniform	0.7442[a, c]	+2%	0.0489[a]	-67%	100%[c]
1000 Collections	Link-Based	0.2573[a]	-65%	0.0341[a]	-77%	15%[a]
	Optimal	0.8412[b]	+16%	0.2983[b]	+102%	0.50%[b]
	Random	0.0786[c]	-89%	0.0056[a]	-96%	0.70%[b]
	Single	0.7279[b]		0.1478[c]		NA
	Uniform	0.3977[a]	-45%	0.0494[a]	-67%	100%[c]

Measurements in the same column and setting, with superscripts of different letter indicate statistically significant difference at 0.05 level

In the last set, the performance of the link-based algorithm decreases, but remains at the same levels as the uniform, with 85% less server utilization. Precision is almost identical as in the second set, showing that even in very distributed environments, its performance remains steady.

Another significant result is that the (retrospective) optimal fusion strategy steadily performs better than any other strategy, including the single collection.In some cases (100 collections) the link-based method also performed better than the single collection. These results verify the feasibility of realizing distributed information systems without sacrificing effectiveness.

5.2 Variations in the Sampling Collection

In the results shown as far, the link-based algorithm utilized a sampling collection made of 20% of WT-Dense for linkage extraction. It would be interesting to explore how the algorithm is affected by the size of the sampling collection.

In Table 3, the algorithm is tested utilizing two more sampling collections for each setting, made of 10% and 40% of WT-Dense. The first 100 results from each sampling collection are used for linkage extraction in each case. For reasons of comparison, we also include the results obtained from the utilization of the 20%.

In the first set, it is obvious that the size of the sampling collection does not play a vital role in the recall of the algorithm. While it does improve as the sampling grows in size, the improvement is not statistically important. Precision on the other hand seems to be much more susceptible to size, showing an important improvement. Server utilization also remains practically the same.

Measurements do not differ in the 100 collections set. An increase in recall is observed with the 20% sampling collection but does not persist in the bigger collection. What is worth noticing is that in both 30 and 100 collection settings, the 20% collection outperforms the 40%. Precision slightly improves as the sampling collection grows in size, but not considerably.

Table 3. Results with various Sampling Collections

Setting	**Sampling Collection**	**Recall**	**Average Precision**	**P@100**	**Collection Utilization**
30 Collections	10%	0.6133	0.0304	0.0149	82,4%
	20%	0.6831	0.0395	0.0163	84,8%
	40%	0.6690	0.0495	0.0154	83,5%
100 Collections	10%	0.6720	0.0418	0.0151	65,6%
	20%	0.7845	0.0420	0.0169	74,0%
	40%	0.7454	0.0448	0.0177	81,4%
1000 Collections	10%	0.2630	0.0089	0.0071	30.7%[a]
	20%	0.2574	0.0341	0.0120	15,4%[b]
	40%	0.3281	0.0280	0.0097	36,9%[a]

In the last set, when the distribution of collections is particularly high, there is a noticeable difference in recall when the biggest sampling collection is used. Precision none the less is better for the medium sampling collection. Overall, the size of the sampling collection isn't a critical factor for the algorithm, thus allowing the algorithm to function effectively with more economical means.

6. Conclusions and Future Work

In conclusion, the performance of the link-based algorithm is promising. Its recall performance approximates or even exceeds that of the uniform while at the same time it prevails on collection utilization. This is a very positive result considering the emphasis that should be given to efficiency when searching in large distributed environments such as the WWW or P2P networks.

One of the main features of the algorithm is that it can be used in parallel with other state-of-the-art source selection algorithms (e.g. CORI, ReDDE etc). The idea of using both linkage and content information is standard practice in centralized IR, but has not been utilized in DIR. The link-based algorithm is a step in that direction.

Finally, in an effort to get a more insightful view of the link-based strategy, it has further been tested in various other settings such as tuning the number of results from the sampling collection, using cut-off on collection utilization and applying weights on the sampling results. Unfortunately, the results from these experiments cannot be reported here due to size limitations of this conference paper.

Acknowledgements

This work was supported by PENED, measure 8.3, action 8.3.1 under project number 404, named "Collection fusion algorithms".

References

Callan, J. P., Zhihong, L. U., Croft, W. B. (1995) *Searching distributed collections with inference networks*, SIGIR '95, ACM Press, pp. 21-28.

Callan, J., Connell, M. (2001) *Query-based sampling of text databases*. ACM Trans. Inf. Syst. 19, (2), pp. 97-130.

Cathal Gurrin, A. F. S (2003), *Replicating Web Structure in Small-Scale Test Collections.* Centre for Digital Video Processing.

Craswell, N. (2000) *Methods for distributed retrieval*, PhD Thesis, The Australian Nation University.

Craswell, N., Bailey, P., Hawking, D. (2000) *Server selection on the World Wide Web*, DL '00, ACM Press, pp 37-46.

Gravano L., Garcia-Molina, H., Tomasic, A. (1999) *GlOSS*: *Text-source discovery over the Internet.* ACM Trans. Database Syst. 24, pp. 229-264.

Jinxi, X. U., Croft, W. B. (1999) *Cluster-based language models for distributed retrieval*, SIGIR '99, ACM Press, pp. 254-261.

Kleinberg, J. (1997) *Authoritative sources in a hyperlinked environment*. Technical Report RJ 10076, IBM 1997.

Luo, Si, Callan, J. (2003) *Relevant document distribution estimation method for resource selection*, SIGIR '03, ACM Press, pp. 298-305.

Salampasis, M., John Tait. (1999) *A link-based collection fusion strategy*. Inf. Process. Manage. 35, pp. 691-711.

Page, L., Brin, S., Motwani, R., Winograd, T. (1998) *The PageRank Citation Ranking: Bringing Order to the Web*, Stanford Digital Library Tech. Project.

Voorhees, E. M., Gupta, N. K., Johnson-Laird, B. (1994) *The Collection Fusion Problem*, TREC-3, Harman, pp. 500-725.