Dyniqx: a novel meta-search engine for metadata based cross search

Conference or Workshop Item

DYNIQX: A novel meta-search engine for metadata based cross search

Jianhan Zhu, Dawei Song, Marc Eisenstadt, Cristi Barladeanu, Stefan Rüger
Knowledge Media Institute, The Open University, United Kingdom
{j.zhu, d.song, m.eisenstadt, s.rueger} @ open.ac.uk; cristi.barladeanu@gmail.com

Abstract

The effect of metadata in collection fusion has not been sufficiently studied. In response to this, we present a novel meta-search engine called Dyniqx for metadata based cross search. Dyniqx exploits the availability of metadata in academic search services such as PubMed and Google Scholar etc for fusing search results from heterogeneous search engines. In addition, metadata from these search engines are used for generating dynamic query controls such as sliders and tick boxes etc which are used by users to filter search results. Our preliminary user evaluation shows that Dyniqx can help users complete information search tasks more efficiently and successfully than three well known search engines respectively.

1. Introduction and motivation

Large search engines such as Google have achieved tremendous success in recent years, thanks to their effective use of the PageRank algorithm [5], smart indexing, and efficiency in searching terabytes of data [6]. Search engines like Google are now moving into the area of searching professional repositories as evidenced by Google Scholar (http://scholar.google.com) and Google Patent Search (http://www.google.com/patents) etc.

In the light of these large scale powerful search engines, how can traditional professional, academic and library repositories survive and keep their successes within their specific domain? Even given the success of the big search engines, in fact it is still very difficult for them to work effectively with repositories that belong to specific professional or proprietary domains. We think there are two main reasons for this.

First, due to legal/proprietary constraints, sometimes search engines cannot get hold of full content of information and may provide only the link to the place where the information can ultimately be found.

Second, big search engines work on the whole World Wide Web, consisting of many resources of a heterogeneous nature and domain context, and thus it is hard for search engines to perform as well as some domain or context specific search services (for example, in the context of arranging air travel between London and New York, the British Airways website will provide much better search services than Google).

We think that the key for successful domain specific specialized search services is to fully utilize the domain context and metadata which describes the domain context. For example, articles in the PubMed (http://www.ncbi.nlm.nih.gov/pubmed/) databases often have rich metadata information such as title, authors, citations, publication date, and publication journal names etc.

However, a limitation of current domain search services has been identified as the wide existence of *information islands* where the integration is difficult, resulting in a contextual "jump" for users when they are searching different repositories [7]. We think that it is important to give users a unified search interface to get access to multiple information repositories, so that they won't get frustrated in finding where to start with.

We treat the problem of building a meta search engine on top of a number of search engines as a collection fusion problem as defined by Voorhees et al. [3, 4]. The research questions we would like to answer are: How to generate a single ranked search result list based on a number of ranked lists from search engines? How to take into account relevance of each result to the query and the original rankings of the search results in the integrated ranked list? How to integrate metadata in ranking?

After reviewing existing work, we found the necessity for a meta-search system that can seamlessly integrate multiple search engines of different natures. Therefore, we propose a novel dynamic query meta-search system called DYNIQX that integrates multiple evidences, namely, search results' relevance to the query, original rankings, and metadata, in collection fusion, and provides a unified search interface on top of multiple search engines. DYNIQX provides plug-in interfaces for new search engines. DYNIQX can help facilitate our investigation of current cross-search and metadata-based search services, identification of resources suitable for cross-search or metadata-based search, and comparison of single source search, cross-search, and metadata-based search.

In the remainder of this paper, we present our novel dynamic query interface system called Dyniqx in Section 2, and report our user evaluation results in Section 3.

2. DYNIQX

Currently many domain specific search engines have adopted what we call a linear/top-down/hierarchical approach. For example, in the Intute search (http://www.intute.ac.uk), a popular search engine among students for finding high quality educational websites, a searcher may select from a list of subject areas and/or resource types for his/her search, and he/she is then taken

to the result page. We think the rigidity of this linear/top-down/hierarchical approach may limit the user to search within the classification of the resources. Additionally, there are many forms of metadata which have not been fully exploited during the search process.

To overcome the rigidity of linear/top-down/hierarchical search, we propose to experiment with the dynamic query approach used to great effect by Shneiderman [1] in other contexts. Dynamic queries help users search and explore large amounts of information by presenting them with an overview of the datasets, and then allow them quickly to filter out unwanted information. "Users fly through information spaces by incrementally adjusting a query (with sliders, buttons, and other filters) while continuously viewing the changing results." A popular example of this approach is that of Kayak.co.uk, a meta-search engine which searches over 100 travel sites to find flights. Kayak uses a dynamic query interface that allow users to change many kinds of filters, such as tick boxes for airlines, and sliding bars for flight departing and arrival times etc., in order to find flights matching these filters. It is our conjecture that a dynamic query interface will dramatically outperform the linear/top-down/hierarchical approach.

In DYNIQX, search results from a number of search engines are fused into a single list by both the relevance of each result to the search query based on our indexing of top results returned from these search engine, and the rankings of the result provided by one or more search engines as below:

$$p_{fuse}(q \mid d) \propto (1-\lambda)\, p(q \mid d) + \lambda / (\log(Rank_{average}(d) + 1)$$

where q is the query, $p_{fuse}(q|d)$ is the fused conditional probability of document d used to rank it in the final list, $p(q|d)$ is the conditional probability of d based on our index, λ is a parameter adjusting the effect of the two components in the final probability, and $Rank_{average}(d)$ is the average ranking of document d given by search engines. In the equation we take the log of the average ranking in order to transform the linear distribution of the rankings of d for integrating with the document conditional probability.

DYNIQX provides a novel way of meta-searching a number of search engines in terms that high quality search results from a number of search engines are integrated, metadata from heterogeneous sources are unified for filtering and searching these high quality search results, high quality results based on a number of queries covering a topic are all integrated in DYNIQX, and features such as metadata-driven controls and term clouds are used for facilitating search.

The architecture of our DYNIQX system is shown in Figure 1. In Figure 1, first, a user sends a query to the DYNIQX system. The query is processed and translated into the appropriate form for each search service, e.g., PubMed. For each query, each search engine, e.g., Intute, PubMed, or Google Scholar, returns a ranked list of search results. Results from all these ranked lists are pooled and indexed by Lucene [8]. Unlike typical search engines where the user can only specify one query at a time, in DYNIQX, the user can specify a number of queries on different aspects of a search topic, e.g., "bird flu", "avian influenza", and "H5N1" etc. in order to find documents relevant to "bird flu". The search results for the number of queries are all pooled and indexed. The user can further refine the search results based on the pooled data. This is illustrated in the DYNIQX search interface shown in Figure 2.

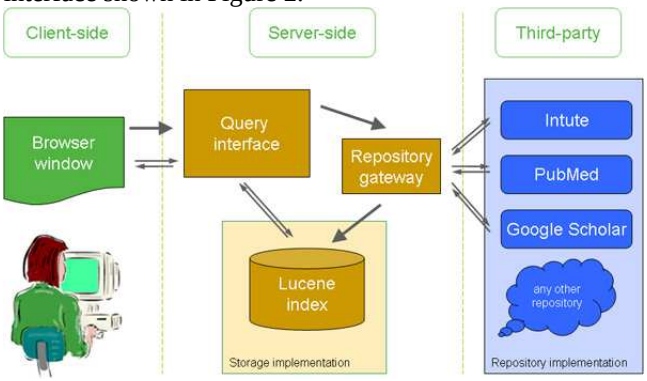

Figure 1: Architecture of DYNIQX

In Figure 2, in Section A the user can add a number of search queries to the pool shown in Section B. The user can reset pool to remove all search results cached in Section A. Statistics of search results from different search engines are shown in a table in Section B. The user can select search engines in Section E. Once search results are retrieved from search engines, the user can view them in Section G. When more new results are obtained from these search services, the user can click a refresh button in Section A to display these new results. Based on the significance of terms measured by document frequency, a term cloud is displayed in Section F. The user can refine the search results by adding some terms from the term cloud to the query. In Section D, the user can further exclude some queries from the pool. Metadata such as article title, author name, journal name, and publication date etc. are used to rank search results in Section C.

3. User evaluation

The aim of our user task-based evaluation is to measure the effectiveness, efficiency, and user satisfaction of DYNIQX. Effectiveness includes at least whether the task was completed successfully. Efficiency includes performance measures such as the elapsed time used for each search and number of viewed pages and mouse clicks etc. We also collect searcher background and satisfaction information.

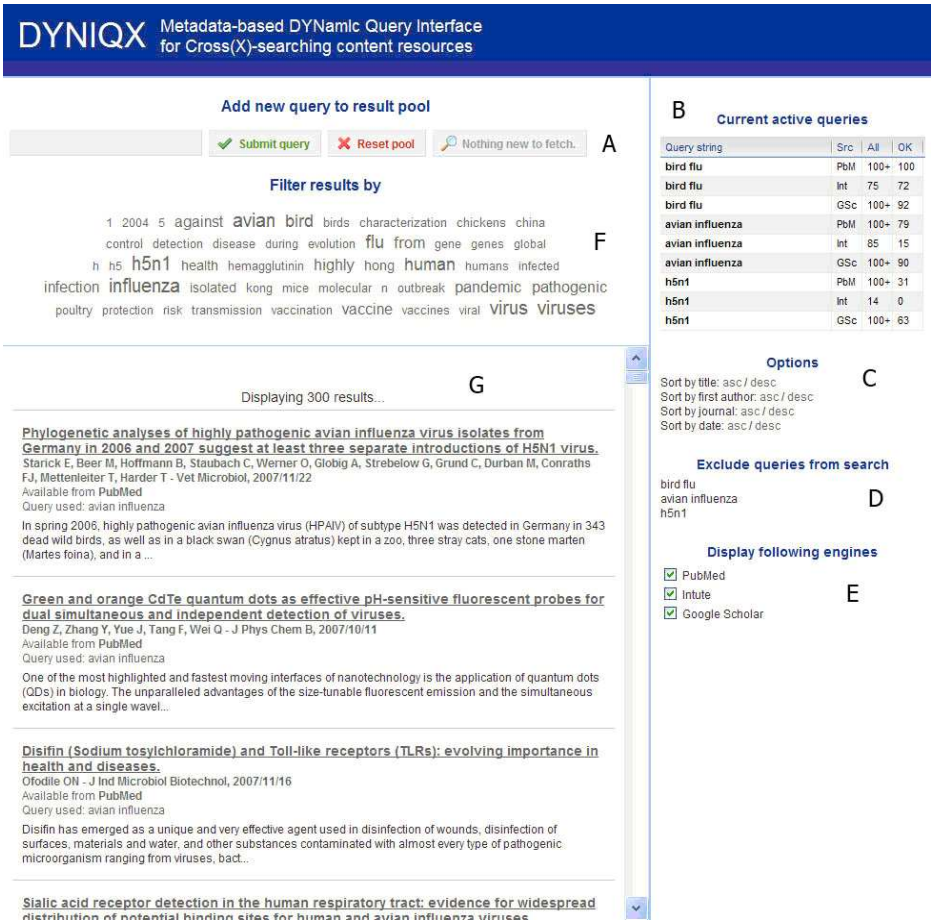

Figure 2: DYNIQX Search Interface

We have carried out a controlled user evaluations of three search engines (Google Scholar, PubMed, Intute), and DYNIQX. In this comparative evaluation, users were given tasks. We used a Latin square design to counterbalance order effects [9]. Based on the comparison, we qualitatively evaluated the usefulness of each search engine.

The four tasks where each consists of a group of related questions are designed as follows which reflect users' real world information needs.

A 'SARS' domain

Q1. Who sequenced the first SARS genome? (if many co-authors, then first two will be sufficient)
Q2. What was the exact publication (journal, date, title of paper)?
Q 3. How long was the genome sequence (typically this means the number of 'bases' or 'base pairs')?

B 'Bird Flu' domain

Q1. When was the first (or second... doesn't matter exactly) officially recorded outbreak of bird flu ('avian flu') in the UK?
Q2. What was the exact publication describing that outbreak [mentioned in 1] (journal, date, title of paper... may not be a scientific paper, but that's OK)?
Q3. What is the name, affiliation (institute) and email address of the lead researcher (don't spend more than a few minutes on this part)?

C 'Foot and Mouth' domain

Q1. When and where was the latest officially recorded outbreak of foot and mouth disease in the UK?
Q2. What was the exact publication describing that outbreak [mentioned in 1] (journal, date, title of paper... may not be a scientific paper, but that's OK)?
Q3. Will foot and mouth disease affect humans? Justify your answer with a journal reference.

D 'Breast Feeding' domain

Q1. What are the pros and cons of breast feeding vs bottle feeding for the baby and the mother (according to a peer-reviewed journal)?
Q2. What is the exact peer-reviewed journal article that has a satisfactory explanation of [1]?
Q3. Is there any connection between breast feeding and breast cancer? Justify your answer with a journal reference.

12 users participated in our evaluation according to the Latin square in Table 1.

Table 1. A Latin square for 12 searchers performing four tasks with four search engines

Searcher	Task Order			
	SARS	Bird Flu	Foot&Mouth	Breast Feeding
A1,A2,A3	Intute (I)	PubMed (P)	GS (G)	DYNIQX(D)
B1,B2,B3	PubMed (P)	Intute (I)	DYNIQX(D)	GS (G)
C1,C2,C3	GS (G)	DYNIQX(D)	Intute (I)	PubMed (P)
D1,D2,D3	DYNIQX(D)	GS (G)	PubMed (P)	Intute (I)

Average age of the 12 evaluators is 27. Among them, there are 6 males and 6 females, 6 PhD students, 4 research fellows, and two university staff representing a range of experience using search engines. While 9 of them are experienced search engines users, 10 of them used Google Scholar (GS) only occasionally.

The user followed the following steps in the evaluation:

Step 1: Entry questionnaire

Step 2: System and task familiarization of four search engines under supervision (10 minutes) and practice with a sample task: "find five researchers working on breast cancer treatment"

Step 3: Complete each task with a search engine, and fill out task questionnaire

Step 4: Complete exit questionnaire

All these questionnaires are online via SurveyMonkey (http://www.surveymonkey.com/).

3.1 Evaluation results

We have used a tool called Slogger (http://www.kenschutte.com/slogger/) to automatically log searchers' activities during their entire evaluation process with their consent. The logged data help us to understand more about the searchers' behaviors during evaluation. Based on the logged user data, we can reconstruct each user's search history such as in Table 2.

Table 2. Example of reconstructed search history for user A1 on the "SARS" domain using PubMed, where the number in brackets shows the number of hits for the user's query on its left.

	'SARS' domain
A1	**PubMed:** who sequenced the first SARS genome (1) SARS genome (449) SARS genome sequence (280) The Genome sequence of the SARS-associated coronavirus. (208) "Marra MA"[Author] Got two answers where one is the right one and the other is by Chinese researchers The Genome sequence of the SARS-associated coronavirus sort by publication date p10 Find right answers to Q1-4 in PubMed, fail on Q5

The average time spent by three searchers on each domain using each search engine is summarized in Table 3.

Table 3. Average time spent by three searchers on each domain using each search engine

Average time (mins)	Task Order			
	SARS	Bird Flu	Foot &Mouth	Breast Feeding
A1,A2,A3	10.3(I)	16(P)	23(G)	17(D)
B1,B2,B3	15(P)	11(I)	13(D)	20(G)
C1,C2,C3	10(G)	11(D)	15(I)	12(P)
D1,D2,D3	3.5(D)	16(G)	16(P)	12(I)

In Table 3, we did a *t-test* [10] based on the average time for each system, and Dyniqx is the most efficient system for the users to search for answers with statistical significance. Surprisingly, for three out of four domains, GS is the most inefficient. We think the reason might be that Dyniqx provides efficient ways for users to filter search results, and users spent lots of time reading large amount of search results returned by GS.

The average number of queries issued by three searchers to each search engine on each domain is summarized in Table 4.

Table 4. Average number of queries by three searchers to each search engine on each domain

Average num of page views	Task Order			
	SARS	Bird Flu	Foot &Mouth	Breast Feeding
A1,A2,A3	11.3(I)	4.67(P)	4.33(G)	3.33(D)
B1,B2,B3	3.33(P)	6.67(I)	4.67(D)	4.67(G)
C1,C2,C3	1.33(G)	3(D)	6.67(I)	3.33(P)
D1,D2,D3	4(D)	2.33(G)	4(P)	4.33(I)

In Table 4, for three out of four domains, users issued the least number of queries to GS than the other three search engines with statistical significance judged by *t-test*. This reflects that GS returns more content for each issued query than the other search engines, therefore, users tend to issue less number of queries.

When a user issues a new query or changes the filtering options of a query, e.g., rank the results by publication date etc., the user will get a new page view. We summarize the average number of page views by three searchers using each search engine on each domain in Table 5.

Table 5. Average number of page views by three searchers using each search engine on each domain

Average num of page views	Task Order			
	SARS	Bird Flu	Foot &Mouth	Breast Feeding
A1,A2,A3	21.3(I)	32(P)	22.33(G)	15.33(D)
B1,B2,B3	37(P)	24.67(I)	17.33(D)	25.67(G)
C1,C2,C3	7(G)	12.67(D)	16.67(I)	47.33(P)
D1,D2,D3	9.33(D)	16.67(G)	43(P)	33.33(I)

In Table 5, for all four domains, users viewed the most number of pages using PubMed among the four search engines with statistical significance judged by t-test.

Based on Table 3 and 5, we can calculate the average time spent by three searchers using each search engine on each page view as summarized in Table 6.

Table 6. Average time spent by three searchers using each search engine on each page view for each domain

Average time per page view	Task Order			
	SARS	Bird Flu	Foot &Mouth	Breast Feeding

A1,A2,A3	0.4843(I)	0.5(P)	1.03(G)	1.1089(D)
B1,B2,B3	0.4054(P)	0.4459(I)	0.7501(D)	0.7791(G)
C1,C2,C3	1.4286(G)	0.8682(D)	0.8998(I)	0.2535(P)
D1,D2,D3	0.3751(D)	0.9598(G)	0.3721(P)	0.3600(I)

Users spent most amount of time per page view using Goolge scholar among all four search engines with statistical significance judged by *t-test*. This matches our observation that each page view returned by GS tends to have more contents than any of the other three search engines, therefore, the users had more to read using GS. However, users spent least amount of time per page view using PubMed among all four engines with statistical significance judged by t-test. This is due to the reason that users are generally having difficulty finding answers using PubMed, therefore, they tend to change the queries or filtering options more often and read less per page view. Our observation is that sufficient amount of time spent for each page view is an important indicator of the quality of search results, i.e., short amount of time spent reading search results indicates that the users are getting frustrated and tend to change the queries or filtering options more often.

Each user rated each search engine on each domain by choosing from very ineffective (-2), ineffective (-1), neutral (0), effective (1), or very effective (2). We average the ratings given to each engine on each domain by three searchers and summarize the results in Table 7.

Table 7. Average rating given by three searchers for each search engine on each domain

Average rating	Task Order			
	SARS	Bird Flu	Foot&Mouth	Breast Feeding
A1,A2,A3	-1.33(I)	0.67(P)	1.33(G)	1.67(D)
B1,B2,B3	-1.67(P)	-1.33(I)	1.33(D)	1.33(G)
C1,C2,C3	0.33(G)	1(D)	-1.33(I)	0.67(P)
D1,D2,D3	1.33(D)	0.33(G)	0.33(P)	-1.33(I)

Dyniqx is the best rated search engine by users, and GS is the second best rated search engines with statistical significance respectively.

We rate the quality of the answers given by each searcher to questions in each domain by choosing from very poor (-2), poor (-1), neutral (0), good (1), or very good (2). We average the quality ratings for three searchers' answers using each search engine on each domain and summarize the results in Table 8.

Table 8. Average rating given by three searchers for each search engine on each domain

Average answer quality rating	Task Order			
	SARS	Bird Flu	Foot &Mouth	Breast Feeding
A1,A2,A3	0.33(I)	0.67(P)	1.33(G)	1.33(D)
B1,B2,B3	-1.33(P)	-1.33(I)	1.33(D)	1(G)
C1,C2,C3	0.33(G)	0.67(D)	-1.33(I)	1(P)
D1,D2,D3	0.67(D)	-0.67(G)	0.67(P)	0.67(I)

Users gave the highest quality answers to questions using Dyniqx among all four engines with statistical significance. We think the reason is that Dyniqx successfully fuses search results from the three engines and the dynamic query interface is effective for filtering and searching.

Overall, based on the quality of the answers found, user ratings for each search engine, and time spent for finding answers, we judge Dyniqx as the most effective, and GS as the second best. Users can use Dyniqx to find better answers more efficiently than the other three search engines. The users also gave Dyniqx the best ratings overall. We think the best performance of Dyniqx is due to its effective use of metadata for filtering, term cloud, pooling of high quality results based on a number of queries, and collection fusion of a number of search engines. GS's good performance is due to its large coverage of information, ranking mechanism, and use of citation information.

3.2 Discussions

Searchers with different background tend to have different behavior in searching for information.

Some searchers seem to be more familiar with search and they are able to issue more complex search queries, such as using complex syntax in query formulation in PubMed.

English speakers can be more able to find answers than non-English speakers, and experienced users can more easily adjust to new search engines and find answers more effectively.

Many people tend to use Google more often than any other search engines and Google has an effect on them when they start using other search engines. For example, some searchers are used to natural language (NL) type of queries while using Goolge. However, other search engines such as Intute and PubMed cannot handle NL type of queries very well.

Due to its domain specific nature, Intute does not have as large a dataset as GS and PubMed. Therefore, users need to choose search query keywords carefully in searching, which create additional difficulty for novice users.

Users' familiarity of a particular domain can affect their search on the domain. For example, questions in the SARS domain tend to be more difficult for searchers with little medical knowledge. Therefore, the quality of answers shown in Table 8 for this domain is relatively lower than that for the other domains.

We also found that it is easy for searchers to find information relevant to a domain, but it can be very difficult for them to confirm whether the information is the most relevant to a question. For example, many users spent lots of time trying to find out whether a paper is the first report on SARS genome sequencing since there are a number of papers published around that time.

Searchers' habits also have effect in evaluation. Some people are more cautious in deciding the right

answers than the others. For example, for the SARS domain, searcher A found a paper returned by GS as the first result. He judged the paper as the first report on SARS genome sequencing based on the paper's high ranking and citation counts. However, searcher B spent lots of time investigating whether this paper is the first paper on the subject by comparing its publication date with many other papers. Clearly that searcher B ended up spending a lot more time than searcher A. In our evaluation, some searchers may even have found the right answer without knowing it.

On the other hand, this shows that search engines are typically much better at finding relevant information than providing proof of the authenticity of the information. Therefore, the process of finding proof can be time consuming.

Since PubMed is for people with medical background looking for academic publications, searchers have some difficulty in using it. [2] shows that PubMed has better coverage on scientific papers than GS, since GS tends to favor older publications which have attracted more citations. However, GS has features such as citations, abstract, keyword highlighting, and PageRank based ranking algorithms to outweigh the benefits of PubMed in our evaluation.

Most users tried to use metadata as soon as they are available, e.g., extracted author information in GS. Our observation is consistent with Kazai and Trotman's findings [11].

3.3 Comments

GS, PubMed, and Intute are built on different datasets. GS has the widest coverage of resources among the three by tapping into a large number of publication information sources on the web. PubMed searches proprietary medical publications. Intute is based on a database of 120365 manually constructed records of high quality descriptions of web resources written by subject specialists from a network of UK universities and partners. Therefore, the three search engines are affected by the scope of the information they search. In order to counterbalance the effect of the scope of information, we have designed the tasks by making sure that all three search engines have a good coverage of all four tasks.

4. Conclusions and future work

In this paper, we propose a novel metadata based search engine called DYNIQX which fuses information from data collections of heterogeneous nature. Metadata from multiple sources are integrated for generating dynamic controls in the forms of sliders and tick boxes etc for the users to further filter and rank search results. Since the effect of metadata in IR has not been sufficiently studied previously, our work provides insights into how to integrate metadata with mostly content based information retrieval systems. Our user evaluation shows that DYNIQX can help users to complete real world information search tasks more effectively and efficiently with statistical significance than three well known search engines, namely, Google scholar, Intute, and PubMed, respectively. In the future, we will integrate other search engines in DYNIQX evaluate our approach on standard TREC datasets, and study the effect of different ranking algorithms in collection fusion.

5. Acknowledgements

The work is funded in part by the JISC (Joint Information Systems Committee) DYNIQX project.

6. References

[1] Shneiderman, B. (1994) Dynamic Queries for Visual Information Seeking. IEEE Software 11(6): 70-77.

[2] Giustini, D., and Barsky, E. (2005) A Look at Google Scholar, PubMed, and Scirus: Comparisons and Recommendations. The Journal of the Canadian Health Libraries Association 26(3).

[3] Voorhees, E.M. et al. (1994) The Collection Fusion Problem. In Proc of Text REtrieval Conference (TREC).

[4] Voorhees, E.M. et al. (1995) Learning Collection Fusion Strategies. In Proc. of SIGIR: 172-179.

[5] Brin, S., and Page, L. (1998) The Anatomy of a Large-Scale Hypertextual Web Search Engine. Computer Networks 30(1-7): 107-117

[6] Ghemawat, S. et al. (2003) The Google File System. In Proc. of ACM Symp. on Operating Systems Principles.

[7] Awre, C. et al. (2005) The CREE Project: investigating user requirements for searching within institutional environments. D-Lib Magazine, October 2005, 11(10).

[8] Hatcher, E., and Gospodnetic, O. (2004) Lucene in Action. Manning Publications Co, ISBN: 1932394281.

[9] MacKenzie, S. (2002) Research Note: Within-subjects vs. Between-subjects Designs: Which to Use? Toronto, Ontario, Canada, http://www.yorku.ca/mack/RN-Counterbalancing.html

[10] Press, W.H. et al. (1992) Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press

[11] Kazai, G., and Trotman, A. (2007) Users' perspectives on the Usefulness of Structure for XML Information Retrieval. In Proc. of ICTIR.