

Queries as Anchors: Selection by Association

Einat Amitay, Adam Darlow, David Konopnicki, Uri Weiss

IBM Research, Haifa Lab, Israel

{einat,darlow,davidko,uriw}@il.ibm.com

ABSTRACT

This paper introduces a new method for linking the world view of the search engine user community with that of the search engine itself. This new method is based on collecting and aggregating associative query trails in the form of query reformulation sessions. Those associative query trails are then used to expand the documents indexed by the search engine. Our method is shown to reduce the time spent searching the index, reduce the need to reformulate queries, and also increase the proportion of queries which fulfill the user's information need. Our work provides a mere glimpse into a new field of study by introducing new types of linking between documents and users' world views. Such links from world views have never previously been considered content that can be indexed and searched over.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Query formulation, Relevance feedback, Retrieval models. H.3.1 [Content Analysis and Indexing]: Indexing methods.

General Terms

Measurement, Documentation, Experimentation, Human Factors

Keywords

Reformulation analysis, Index enhancement, document expansion.

1. INTRODUCTION

To better understand the ideas presented in this paper, we would first like to return to Vannevar Bush's 1945 original manuscript and read his description of the idea of *Selection by Association*.

"The human mind [...] operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. It has other characteristics, of course; trails that are not frequently followed are prone to fade, items are not fully permanent, memory is transitory. Yet the speed of action, the intricacy of trails, the detail of mental pictures, is awe-inspiring beyond all else in nature.

Man cannot hope fully to duplicate this mental process artificially, but he certainly ought to be able to learn from it. In

minor ways he may even improve, for his records have relative permanency. The first idea, however, to be drawn from the analogy concerns selection. Selection by association, rather than by indexing, may yet be mechanized. One cannot hope thus to equal the speed and flexibility with which the mind follows an associative trail, but it should be possible to beat the mind decisively in regard to the permanence and clarity of the items resurrected from storage." [3]

In this paper we chose to take *Selection by Association* in a different direction. We aim to show that the process of associative selection also exists in the paths people choose when querying information, similar to the notion described in [9] and [10]. More importantly, we also aim to show that this process introduces new knowledge into the information system. This knowledge about the world was not previously recorded in the search engine's index and can lead, eventually, to a change in the way the information is being used.

The problem we are trying to solve in this paper is illustrated in Figure 1. Search engines, as will be explained later, capture only parts of the information known to their users. In fact, some users have no overlapping knowledge with the search engine's index and thus these users have no way of finding what they want. Since the knowledge the search engine relies on is based on a fixed collection of pre-authored documents, there is currently no simple method to translate the user's request to something that the engine does know about.

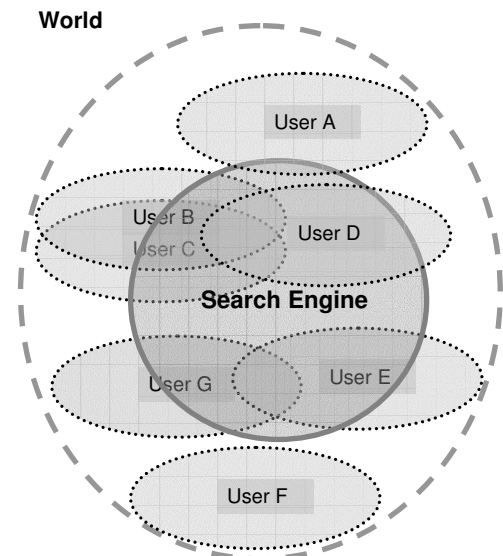


Figure 1 – The world of knowledge as it is perceived by search engines and their users

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. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

HT'05, September 6–9, 2005, Salzburg, Austria.

Copyright 2005 ACM 1-59593-168-6/05/0009...\$5.00.

The work presented here creates dynamic “links” labeled by “anchors” from the world as it is known to the user, to the world as it is indexed by the search engine. These dynamic links and anchors are in turn used to expand the documents within the system. The index is then modified to reflect the knowledge of the search engine’s user community and to adapt its content to the world view of those who search its index. We create those new links and anchors based on associative query trails left by the search engine’s users when they are looking for information.

2. THE POWER OF QUERIES

A search engine serves users by helping them to find content which is relevant to their information needs. This content is taken from a collection of pre-authored documents. What is commonly overlooked is that the searchers are also constantly creating content, the queries with which they search. This content can be helpful in providing searchers with relevant information. Although queries are short and are unlikely to fulfill a searcher’s need as an answer, the information encapsulated in queries can still be used. We have investigated certain groups of queries called *reformulation sessions* [1] and show how they can be used in such a manner.

2.1 Query reformulations

When a search engine user issues a query, they are trying to find information to fulfill an information need. Often the user is not satisfied with the results returned by the engine. One option the user has at this point is to rephrase the query. The user may do this a number of times before either they are satisfied or they give up, thus creating an associative query trail called *query reformulation*. This series is referred to as a *reformulation session*. More formally, as illustrated in Figure 2, a reformulation session is defined as a series of at least two queries issued by a user in order to satisfy a single information need.

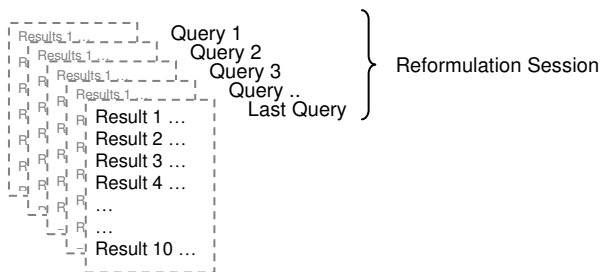


Figure 2 – Reformulation session in response to a single information need

Below are several examples for query reformulation sessions taken from our experimental search engine query log:

Euro other name
"the Euro" other name
Euro name previous

marine vegetation types
marine vegetation
marine plants
"marine vegetation"
sea plant

attention defecit disorder
what is attention defecit disorder
"attention defecit disorder"
what is ADD
ADD attention

Query reformulation sessions are interesting because they are a typical process of *selection by association*. Each query in the session is an association triggered by the information need, the previous queries, and the results they returned. If the search engine was able to make those same associations, the user would not have needed to reformulate the original query.

In [1], we found that, on average, over 30% of queries are part of reformulation sessions. These results are based upon extensive query log analysis of two large enterprise search engines deployed within IBM’s intranet and IBM’s external Web site.

3. QUERIES AS ANCHORS

In this section we show how the associations inherent in query reformulation sessions can be added to the search engine content by treating them as implicit links.

3.1 Reformulations as implicit links

When a user reformulates a query, it is because they did not get the results they need. We assume that the last query in the session does return good results as intended by the original query. This implicitly links each of the queries in the session to those last result documents. It is these implicit links we are concerned with in this paper. The text of the last query itself is less interesting because the search engine already returns the desired documents for that query.

The implicit links derived from query reformulation sessions are products of the associations which the user applied when reformulating the query. For example, the association between the words *vegetation* and *plant* was used in the marine vegetation session above, and is therefore part of the link between the query *marine vegetation types* and the resulting documents of the query *sea plant*. Note that likely the word *vegetation* does not even appear in those documents. This is exactly the information which is missing from the search engine’s index, and which we add using query reformulation driven links.

We later show that even the assumption made earlier about the last query being successful is, in practice, not necessary for improving the findability of information within a search environment enhanced by such links.

3.2 Document expansion

Document expansion is an extreme form of intervention search engine developers choose to take in order to improve search quality. The best known example is the replication of anchor text of a link which points to a certain target page, within the target page [5]. This means that even if the authors of the target page never intended to include the description provided in the anchor text of the link, users of the search engine can still receive it as a result if their query used only words from the anchor text.

Document expansion is a well targeted form of expansion. It allows only very directly related texts to be associated with a specific document.

In this work we chose to explore the idea of further expanding documents with information that is similar in nature to anchor text, but is supplied by the search engine users. That is, we chose to expand documents with queries.

3.3 Enhancing an index with queries

In this work we enhance the search engine's index by attaching query strings derived from query reformulations to documents, as described above. A similar process is commonly performed with anchor text. Specifically, with query reformulations, given a query log that contains all queries and the corresponding result documents for each, the reformulation sessions are extracted as described in [1] and all sessions with a single query are ignored. All query syntax characters, such as plus signs and quotation signs, are removed, and all terms with minus signs are ignored as well. In the *Euro* reformulation example above, the second query was changed from "the Euro" other name to the Euro other name.

For every session with n queries, q_1 to q_n , where the last query, q_n , returned k result documents, we append the text of the first $n-1$ queries, q_1 to q_{n-1} , to each of the k result documents. The process is illustrated in Figure 3. In the same example as above, the texts *Euro other name* and *the Euro other name* were appended to each of the result documents of the query *Euro name previous*.

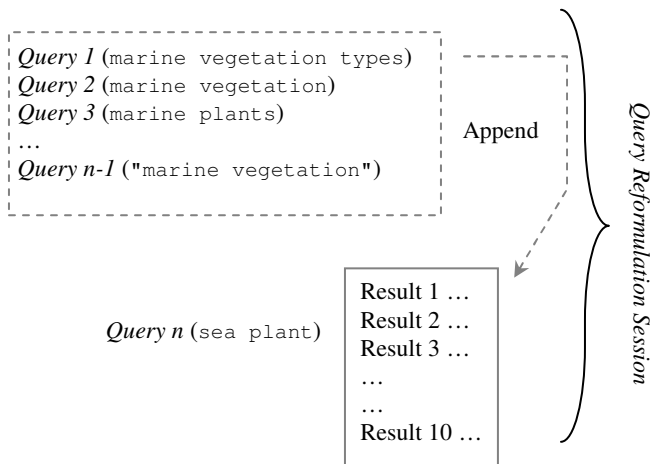


Figure 3 - Adding the text of queries 1 to $n-1$ in the reformulation session to the text of documents returned for the last query n in the session

4. RELATED WORK

Improving the users' queries by adding information to them is a very well known solution used throughout the information retrieval research world. Most systems use the index to expand or refine the user query [22]. There are some other solutions proposed by the research community to facilitate the findability of information within the search system. Since this paper proposes to solve the findability of information problem by attaching full query reformulation sessions to documents that were found last in the session, we would like to mention several related studies that make use of queries for suggesting new options to the user; that

base new suggestions on users' query history (be it in a session or not); that expand documents with queries (be it in a session or not); or that translate queries into anchors as we do in this paper.

The direct outcome of query reformulation sessions has not been studied by many since the notion of reformulation was not well defined. There are studies that do utilize a guided form of query reformulation to enhance the querying process [11][21]. In these early studies the user is invited to choose reformulations from a set of predefined related concepts. These studies were preceded by the pioneering work of Reisner in 1966 [17] who conducted an experiment by conventional snail-mail to build a thesaurus of information needs and concepts based on users' suggested reformulations.

The idea that query reformulations are useful for enhancing the search process was also studied in the 1980s by a very prolific group at Bell Labs [7][8][18]. This group studied the utility of reformulation trails for automating an interactive search process of guided query reformulation. In the hypertext SuperBook [18], for example, users were invited to add annotations and possible synonyms to terms in the documents in order to improve the findability of those documents.

The use of single queries and query reformulation trails for automatic query expansion was more recently given a fresher outlook [2][6][12]. Those recent studies learn from the analysis of previous queries about what might be of interest to a user currently looking for similar information. They are cautious not to alter the index itself but do automatically expand or alter the queries submitted to reflect the knowledge accumulated from previous query sessions.

Document expansion, as in attaching queries to documents, is proposed by Scholer and Williams in [19]. However, the authors only reinforce documents by appending the query that was used to retrieve that very same document and hence strengthening information that already exists in the index. This approach is reported to improve on tasks that resemble home page finding on the Web. However, Scholer and Williams do not solve the problem of missing descriptive content or alternative phrasings that would help users find relevant documents.

In our work presented in this paper, we attach entire query sessions to the last set of results that were retrieved with the last query in a reformulation session. We chose not to append the last query to the retrieved documents in order to make sure we distinguish ourselves from the study of Scholer and Williams and to better understand the benefits of our approach.

The last study we would like to mention, by Kraft and Zein, also relates anchors with queries [13], but takes it in the opposite direction. Anchors are used by Kraft and Zein to enhance the automatic query expansion process. The authors report that query expansions generated from anchors were perceived to be superior to those generated by simply taking a list of previous queries (not divided into query reformulation sessions but taken as single occurrences).

5. EXPERIMENT DESIGN

In order to validate our claims that query reformulation sessions appended to documents can help searchers find better results with fewer reformulations, we conducted an experiment in the form of

a search contest (named the “Search Wiz Quiz”). This contest creates a setting where many users share the same information need. We specifically chose questions whose answers exist in the collection in some form but are difficult to find. Below we outline the settings of this experiment.

5.1 Experiment stages

The experiment we conducted included the following stages:

1. A first round of the search contest was run. A group of users was invited by email to participate and answer a set of questions. Users were able to quit without completing the set.
2. We applied our algorithm to enhance the index using the query reformulation sessions recorded in the first round.
3. A second round of the search contest was run with different participants. Participants in this round searched for the same contest questions using the enhanced index.
4. Finally, we analyzed the difference in results between the two groups of participants to measure the effects of our techniques.

5.2 The environment

The contest was implemented with a specially designed web site. Upon login in to the contest site, each participant was asked to provide their email for future identification (and a prize draw). The rules of the contest were then displayed. Participants were asked to answer a random set of five questions (out of 11 possible questions) whose answers were found in some documents indexed in the dedicated collection. Participants were asked to answer the questions without using other sources, although we were unable to enforce this. Then, the screen was divided into two frames. On the left side was a question with an input field for entering an answer. On the right side was the search engine interface.

At every stage, participants could take one of the following actions:

1. Enter a query in the search field and press the “Search” button to execute a search against our dedicated collection and obtain a set of results. Only the top ten results were displayed with short document summaries. No “next” button was provided, and participants could not navigate beyond those top ten results.
2. Click on one of the results to examine the content of the document.
3. Enter an answer to the current question, press the “Submit answer” button and go to the next question (or to the end of the quiz).
4. Click on the “I don’t know” button and go to the next question (or to the end of the quiz) without providing any answer to the current question.

5.3 The log

All users’ interactions were logged. While being logged in, every participant was associated with a unique ID and was identified using an HTTP cookie. Every action was timestamped and logged together with the user’s ID. The actions logged were:

1. questions asked
2. queries issued
3. results that were returned by the index for each query
4. results that were clicked by the user
5. answers provided to the questions presented
6. “I don’t know” button being pressed

5.4 The collection

The collection used was TREC-8 [20] which consists of 528,155 documents from various news sources. The TREC-8 collection consists of non-linked texts and hence has no anchor text available for indexing. This collection has been widely studied and has 249 “topics” which consist of queries and corresponding sets of marked relevant documents. This was useful for creating difficult search questions. The collection was searched using the Juru search engine [4].

5.5 The questions

Each participant was presented with a set of five questions, randomly chosen and ordered, out of eleven possible questions. Those questions were “difficult” in the sense that search queries that are similar to the question itself did not return results containing the answer. This is mostly due to ambiguity or use of synonyms or acronyms, none of which were handled specially by the search engine. This forced users to reformulate their queries. One way in which we created difficult questions was by deriving them from TREC queries for which we know the Juru search engine returns no relevant results in the top ten results.

Table 1 lists all the questions used in the experiment. The table also shows the number of answers received for those questions in each round.

Overall, 283 people participated in the experiment, of which 178 people participated in the first round of the experiment and 105 people participated in the second round. None of the participants overlapped and all participants were IBM employees.

Table 1 – Questions selected for the experiment and the number of users who received each question in the first and second rounds

| Question | Derived from TREC | Num. of users in 1 st / 2 nd rounds |
|--|-------------------|---|
| When was the war in Islas Malvinas? | | 43/41 |
| What is the other acronym closely related to ADD? | | 54/35 |
| Who opposed the Euro (list at least three)? | ✓ | 58/33 |
| Who filled the PM position in Britain in 1990? | | 63/40 |
| What does the FLAG acronym mean? | ✓ | 53/36 |
| Name 3 types of marine vegetation | ✓ | 54/34 |
| What was the other name used for the Euro? | | 59/38 |
| In Russia, sometime in March 1992, there was a leak in a nuclear plant, what was its name? | | 51/35 |
| What type of engine was invented in 1816? | ✓ | 53/36 |
| What is ADD? | ✓ | 63/35 |
| Name three organic soil enhancers? | ✓ | 54/42 |
| Total | | 606/405 |

5.6 Creating the new index

After the first round, the reformulation sessions were extracted from the log. We then added the query reformulation sessions text to the index, as described in section 3.3. The queries were added as normal text without using any weighting or threshold methods.

In creating the index we considered several options. One option was to only include reformulation sessions which ended with a correct answer. Another option was to include reformulation sessions that led to both correct and incorrect answers. We chose a third option, including everything that was identified as a session, whether correct, incorrect or ending with pressing the “I don’t know” button. We chose this option in order to demonstrate that even when there is no certainty as to whether a user received satisfactory results, the agreement among the “correct” sessions may be strong enough to “silence” the noise introduced by the “incorrect” sessions.

We actually tried all three options and found that the effect was very similar in all three cases, and that the same documents came up as being the most enhanced with queries regardless of the option used. The reason for this being that the successful sessions agreed among themselves to a great degree and the unsuccessful sessions did not. This resembles the consensus among authors of handwritten annotations in similar sections in hard copies of the same book studied by Marshall [14]. We consider this a strong indication of the robustness of the algorithm in the case of

sessions which did not end in a successful query, relaxing the assumption we made earlier that the last query in a reformulation session should be successful.

For example, 117 query strings were added to document FT923-9701, several of which appear below:

<what type of engine was invented in 1816>
<what engine was invented in 1816>
<in 1816, what engine was invented>
<who invented an engine in 1816>
<1816 engine invent>
<stirling>
<stirling engine>
<engine, 1816>
<engine, 1816 patent>
<invention 1816>
<engine invented in 1816>
<19th century invention engine>
<invent before 1820 engine>

Overall, we expanded nearly 1600 documents within our collection with queries coming from reformulation sessions. The most “enhanced” documents were expanded with more than 100 query strings.

5.7 Checking the index integrity

A concern which was raised during this research regards the noise that document expansion can cause. More specifically, even if enhancing the index has a positive effect for the queries which were added, most other queries might be negatively affected. This could have a detrimental effect on the overall quality of the search. In order to test this we ran the 249 TREC topic title queries (which correspond in length and quality to “regular” search engine queries) before and after enhancing the index and compared the results. The TREC topics are a standard benchmark for search quality.

6. RESULTS

We compared the two experiment rounds using two kinds of measures: those which measure how successful the users were in finding answers, and those which measure how long it took them to provide an answer. We will show that enhancing the index with queries resulted not only in shorter sessions but also in more correct answers provided by participants.

Although the experiment was not extremely large as far as the number of users (283) or the range of questions asked (11 overall), the results are statistically significant and decisively positive. On average, users who were given the enhanced index (second round) found more correct answers with fewer reformulated queries than those given the original index (first round).

6.1 Session length

We compared the average session length for each question, both in number of queries and in seconds. These measures map directly to the gain that can be achieved by adding queries to an enterprise search:

- Fewer queries means a lower load on the search engine system and the network
- Shorter duration means employees spending less time finding the information they are searching for

Table 2 – Average session length in queries per question before and after index change

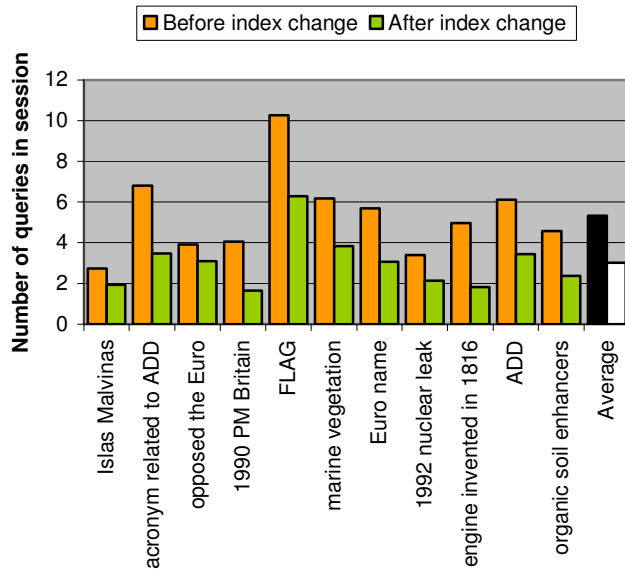
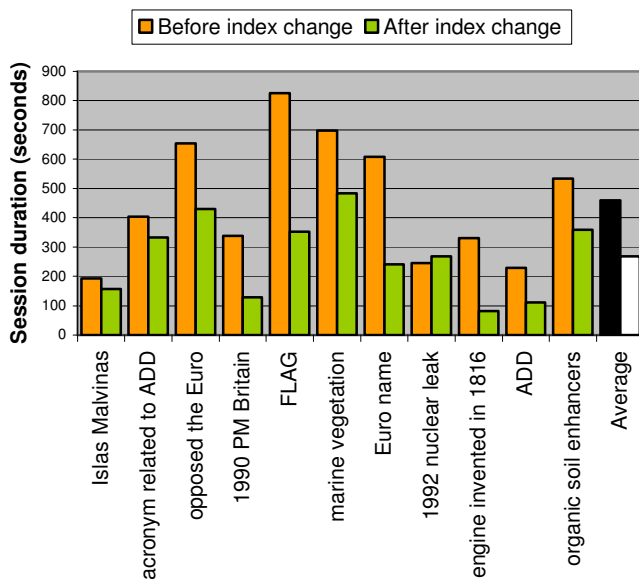


Table 3 - Session duration in seconds, before and after introducing index change



As shown in Table 2, on average, the number of queries in each session was reduced by about 44% from 5.33 queries per

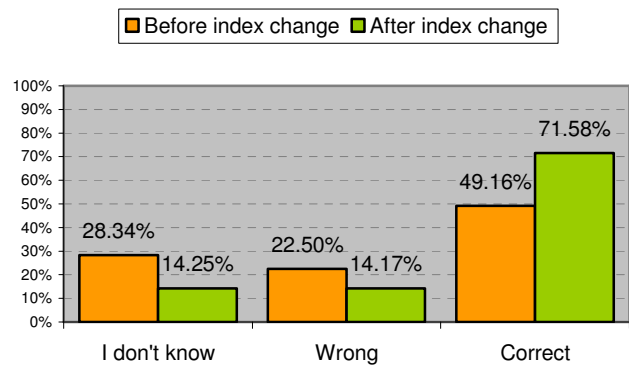
reformulation session in the first round to 3.01 queries per reformulation session in the second round.

Table 3 shows how the average session duration in seconds was reduced by almost 42% from 459.61 seconds per reformulation session in the first round to 267.71 seconds per reformulation session in the second round. This is a significant reduction which shows quite clearly the effectiveness of our method for document expansion with query reformulation sessions.

6.2 Quality of answers

In addition to the length of sessions, we also compared how many users answered each question correctly. Although improving session length is the direct objective of the algorithm, it is not surprising that these measures were affected as well, as seen in Table 4. One explanation is that different users, through their query reformulations, tend to agree mostly on correct answers, leading to the documents containing those answers being strengthened more than any others. Therefore, any similar query is likely to return some of those documents. Another explanation is that users are less inclined to give up when they are required to perform less query reformulations.

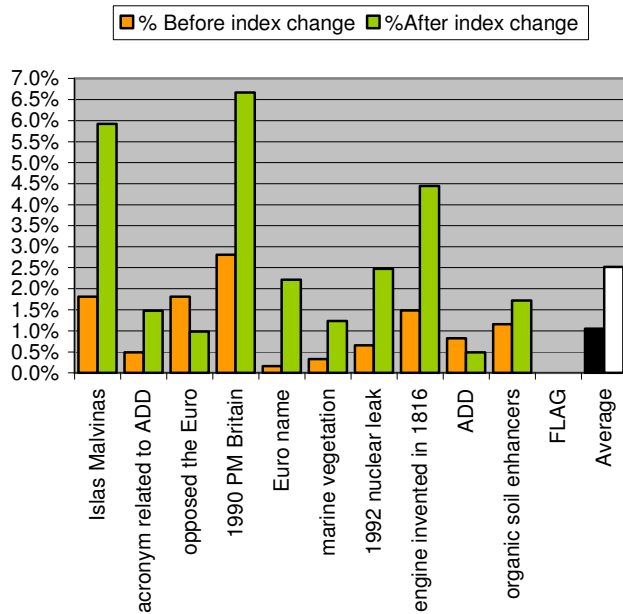
Table 4 – Percentage of correct, incorrect and “I don’t know” answers given per experiment before and after index change



To strengthen our conclusions we also counted the number of users who found a correct answer with the first query they submitted. This means that no reformulations were necessary, which is the desired scenario in a search engine. Table 5 provides the percentage of people who were able to provide such a correct answer for each question out of the 11 within the two experiment rounds.

From the two groups of participants taking part in our experiment, only 9 participants (5.06%) were able to provide a full set of five correct answers in the first round (before the index changed), while 21 participants (20%) were able to provide such a full set of correct answers in the second round (after the index changed).

Table 5 - Percent of participants able to provide a correct answer after a single query



6.3 TREC results

Since TREC is a standard test for search engine quality, we used it to check how the overall quality of the index was affected. The net effect was a slight improvement in all measures as seen in Table 6. The measures presented in this table are correctness of the result ranked first (Prec@1), correctness of the top five results (Prec@5), correctness of the top ten results (Prec@10). Also provided are the mean average precision (Average Prec), and precision at R, where R is the number of relevant documents in the collection for the query (Average R Prec). Also in the table is the Mean Reciprocal Rank measure (MRR) which provides an indication about the rank of the first correct result within the top ten results. Number of Prec@10=0 and Number of Prec@10=10 are simply counts of how many sets of no results or full sets of ten correct results were found within the 249 topics tested.

What is interesting to analyze is the breakdown of these results since they exemplify both the strengths and weaknesses of our approach. Nine queries were significantly improved. These include all the six queries from which our experiment questions were derived. Three additional queries which shared some information with the derived queries, like the query “*inventions, scientific discoveries*” were slightly improved because of the conceptual overlap.

Six other queries slightly worsened. Two queries were significantly worsened by the additions of query reformulation session texts. One of the worsened queries was “*Falkland petroleum exploration*”. The negative effect was caused by the question “*When was the war in Islas Malvinas?*” which is relevant to Falkland but not to petroleum exploration. This demonstrates the democratic nature of our algorithm. Since all of our users were interested in wars in Falkland, and none were interested in petroleum exploration, documents concerning the

petroleum exploration received no boost. We further address this issue in the discussion section. The other worsened query was “*Valdez wildlife marine life*” which was affected in a similar manner by the question “*Name 3 types of marine vegetation*”.

Table 6 – TREC standard measures on both indices: Basic index without query reformulations and “grown” index including those queries

| | Before index change | After index change |
|-----------------------------|---------------------|--------------------|
| Prec@1 | 0.414 | 0.426 |
| Prec@5 | 0.36 | 0.365 |
| Prec@10 | 0.333 | 0.335 |
| Average Prec | 0.231 | 0.233 |
| Average R Prec | 0.254 | 0.255 |
| MRR | 0.527 | 0.54 |
| Number of Prec@10=0 | 46 of 249 | 42 of 249 |
| Number of Prec@10=10 | 6 of 249 | 7 of 249 |

7. DISCUSSION

Query reformulations have a distinctly democratic nature. The approach of adding them directly to documents has advantages and disadvantages. The main advantage is that unlike query expansion methods, many types of noise have little effect, as demonstrated by adding both the wrong and the “I don’t know” answers to the correct answers in our experiments. Only documents that are linked to by many queries are enhanced significantly. The main disadvantage is that in communities in which there is less agreement about the correct answer, if a certain topic/answer is very popular, the queries pointing to it may make locating semantically similar but less popular topics, difficult.

7.1 Implications of enhancing documents with query reformulation sessions

Large scale deployment of this procedure must be done carefully, similar to the way anchors are used in current document expansion methods. If all reformulation sessions are added as normal text to documents, the original text would become negligible as more and more queries are attached to each document. There are several straightforward ways to limit the effects of the queries which we discuss below.

7.1.1 Upper bound for expanding documents

An *upper bound* limits the combined weight of all queries attached to a specific document. This is used to prevent any single document from receiving too large a boost from the queries. It could also be used to ensure that queries do not become more significant than the original text. From our experience with expanding documents with anchor text such an upper bound threshold stands in the lower thousands for a large collection. An upper bound could be enforced by limiting the number of queries added to a document, reducing the weight of query text during query evaluation or a combination of both.

7.1.2 Timeliness of queries

Queries are temporal in nature. They often represent current interest and the most recent information need. As Vannevar Bush stated in his 1945 article, “...trails that are not frequently followed are prone to fade, items are not fully permanent, memory is transitory” [3].

In the case of associations, which change often with time, impermanence is as much a strength as a weakness. This suggests that a similar strategy should be used when attaching queries to documents. Each query should be detached from the expanded document and removed after a certain period of time. If the association which the anchor represented is still valid, it is likely to be replaced with other similar queries that users add repetitively and continuously.

7.1.3 User community bias

Adding queries to documents is a democratic process. In this sense the search engine user community dictates what results are more important given a query or a query association trail. This process affects ranking directly and may sometimes result in introducing a very focused interpretation of a query term that may not be acceptable in a different search community. Nevertheless, if there are several interpretations of the query term with significant portions of the user community looking for them this may result, as is the case with anchors being added to documents today, in a rich assortment of results pertaining to all the popular interpretations.

In the extreme case, queries as anchors strengthen documents on which there is a single consensus. If 99% of the queries for “Bush” were then reformulated as “President Bush” it would become significantly more difficult to find documents pertaining to Vannevar Bush. For this reason, as stated earlier, it would make sense to limit the scope of the queries added to those documents by introducing an upper bound threshold.

7.1.4 Sharing query reformulations among users

Sometimes it makes sense to share information between groups of people working together, sharing common interests, or simply friends with common background knowledge. In such sub-communities with similar interests, each user may want to share their associative query trails and be affected more significantly by other users from their own community.

In this way, an occasional query reformulation by a user from the hypertext conference community from “Bush” to “Vannevar Bush” would strengthen the hypertext-related interpretation of the term “Bush” within the documents shared by this sub-community.

This idea of sharing associative query trails resonates in recent studies about sharing information found in documents on the Web or sharing annotations about shared materials within working groups of people. We contend that query reformulations, in a similar manner, can be shared/traded by users to improve the searchability of certain environments or certain social/work-related groups [15][16]. In such an environment, new people coming to an established community will receive “permission” to use the meta-information stored within the documents. This meta-information will be the query reformulations collected by all group members.

7.2 The ideal environment

The ideal environment for applying our approach is probably within an enterprise search engine or a search engine that is dedicated to a certain focused collection where anchors/links and other meta-information are not as prevalent. Also, such closed environments are less prone to link spamming and commercially-driven optimization of search results, which may be an issue with large and public Web search engines.

A focused user communities may also enjoy such an environment, as mentioned above, where members of the community share interests, common goals, and associative thinking. In the enterprise, for example, such sharing of information may result in substantially saving costs by reducing the duration of searching within the organizational information collection.

A good example for such a community may be the call-center or the helpdesk of an enterprise, where the questions are usually very focused and there is a price tag to every minute spent on solving customer problems. Such an environment also does not have any traditional hypertext structure and hence the democratic voting process inherent in anchor-driven document expansion is absent from the collection.

8. FUTURE WORK

Since the approach presented in this paper is new it requires further study. Many aspects of our techniques require longitudinal observations for monitoring proper threshold settings, monitoring usage, monitoring temporal effects introduced by user communities, and last but not least – monitoring the usefulness of such a system in a real user community with common goals and interests.

Below we provide a glimpse into two issues that were raised by our experiments. We hope that these examples for the problems that arise by introducing the world views of the search engine user to the index of the search engine itself will whet the readers’ appetite to further explore the path of document expansion with queries.

8.1 How to use syntax reformulations

In this work, we chose to drop all terms that are marked by minus signs. This is valuable query disambiguation information provided by our users which we discard only because we do not yet know how to use it.

The problem of expanding documents with negative information is both logical and technical. We considered some possible solutions to this problem, however none is ideal. When solving this problem one should remember that as opposed to the positive terms added to the last result set, people use negation as a filter and thus they probably refer to all seen results except the last set.

Other query syntax elements such as phrases and plus terms might also be worth future investigation.

8.2 Combining document expansion with query expansion

In this paper we have been very conservative with our approach, testing only the document expansion hypothesis. However, combining document expansion with query expansion methods

may help improve syntax reformulation issues like the one mentioned above. Dynamic techniques which combine document expansion with offering users a selection of query expansions may provide an interesting venue for future research.

9. REFERENCES

- [1] Amitay E., Darlow A., Weiss U. (2005). Conversearching with Engines. Submitted.
- [2] Billerbeck B., Scholer F., Williams H.E., Zobel J. (2003). Query expansion using associated queries. in Proceedings of ACM CIKM 2003, pp. 2-9.
- [3] Bush V. (1945). As We May Think. *The Atlantic Monthly*, 176(1):101-108.
- [4] Carmel D., Amitay E., Herscovici M., Maarek Y., Petruschka Y., Soffer A. (2001). Juru at TREC 10 - Experiments with Index Pruning. in Proceedings of NIST TREC 10, Nov 2001.
- [5] Chakrabarti S., Dom B., Raghavan P., Rajagopalan S., Gibson D., Kleinberg J. (1998). Automatic resource compilation by analyzing hyperlink structure and associated text. in Proceedings of the 7th WWW conference, Computer Networks and ISDN Systems, 30(1-7):65-74.
- [6] Fitzpatrick L., Dent M. (1997). Automatic feedback using past queries: social searching? in Proceedings of ACM SIGIR '97, p.306-313.
- [7] Furnas G.W. (1985). Experience with an adaptive indexing scheme. in Proceedings of ACM CHI '85, pp. 131-135.
- [8] Furnas G.W., Gomez L.M., Landauer T.K., Dumais S.T. (1982). Statistical semantics: How can a computer use what people name things to guess what things people mean when they name things? in Proceedings of the 1982 conference on Human factors in computing systems, pp. 251-253.
- [9] Golovchinsky G. (1997a). What the Query Told the Link: The Integration of Hypertext and Information Retrieval. in Proceedings of ACM Hypertext '97, pp. 67-74.
- [10] Golovchinsky G. (1997b). Queries? Links? Is There a Difference? in Proceedings of ACM CHI '97, pp. 407-414.
- [11] Henninger S. (1994). Using Iterative Refinement to Find Reusable Software. *IEEE Software*, 11(5):48-59.
- [12] Huang C.K., Chien L.F., Oyang Y.J. (2003). Relevant term suggestion in interactive web search based on contextual information in query session logs. *JASIST* 54(7):638-649.
- [13] Kraft R., Zien J.Y. (2004). Mining anchor text for query refinement. in Proceedings of WWW 2004, pp. 666-674.
- [14] Marshall C. (1998). Toward an ecology of hypertext annotation. in Proceedings of ACM Hypertext '98, pp. 40-49.
- [15] Marshall C.C., Bly S. (2004). Sharing Encountered Information: Digital Libraries Get a Social Life. in Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL04), pp.218-227.
- [16] Marshall C.C., Brush A.J. (2004). Exploring the Relationship between Personal and Public Annotations. in Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL04), pp.349-357.
- [17] Reiser P. (1966). Evaluation of a "Growing" Thesaurus. IBM Research Paper RC-1662, IBM Watson Research Center, 19 p..
- [18] Remde J.R., Gomez L.M., Landauer T.K. (1987). SuperBook: an automatic tool for information exploration — hypertext? in Proceeding of ACM Hypertext '87, pp. 175-188.
- [19] Scholer F., Williams H.E. (2002). Query association for effective retrieval. in Proceedings of ACM CIKM 2002, pp. 324-331.
- [20] Voorhees, E. M. (2003). Overview of the trec 2003 robust retrieval track. Proceedings of the Twelvth Text Retrieval Conference (TREC-12). National Institute of Standards and Technology (NIST).
- [21] Williams M.D. (1984). What Makes RABBIT Run? *International Journal of Man-Machine Studies*, 21(4):333-352.
- [22] Xu J., Croft W.B. (1996). Query expansion using local and global document analysis. *ACM SIGIR* 1996, pp. 4-11.