In the Proceedings of SDAIR '96, Las Vegas, NV, April 1996.

Improving Full-Text Precision on Short Queries using Simple Constraints

Marti A. Hearst
Xerox PARC
3333 Coyote Hill Rd
Palo Alto, CA 94304
(415) 812-4742
hearst@parc.xerox.com

Abstract

We show that two simple constraints, when applied to short user queries (on the order of 5–10 words) can yield precision scores comparable to or better than those achieved using long queries (50–85 words) at low document cutoff levels. These constraints are meant to detect documents that have subtopic passages that includes the most important components of the query. The constraints are: (i) a simple Boolean constraint which requires the user to specify the query as a list of topics; this list is converted into a conjunct of disjuncts by the system, and (ii) a subtopic-sized proximity constraint imposed over the Boolean constraint. The vector space model is used to rank the documents that satisfy both constraints. Experiments run over 45 TREC queries show significant, almost consistent improvements over rankings that use no constraints. These results have important ramifications for interactive systems intended for casual users, such as those searching on the World Wide Web.

1 Introduction

Recent efforts in evaluation of full-text retrieval systems, especially as seen in the TREC competition [9], work with very long, detailed query specifications. It has been observed in TREC that longer query descriptions result in higher overall results than short ones [17, 30], and an overall drop in the scores of the top-ranked systems was observed when given shorter, more "realistic" queries in TREC-4 [10].

However, casual users in most cases do not formulate long queries; online queries usually consist of 10 words or fewer [6, 21]. This problem can be remedied to some extent with relevance feedback [27] and automatic term expansion [17], both of which have been researched extensively. Nevertheless, a system responding to *ad hoc* queries should make the best use possible of limited user queries, and methods are needed for improving retrieval results given only very short queries. This is especially relevant for search systems that run on the World Wide Web.

Additionally, we conjecture that an important goal in interactive information access systems is to try to ensure high precision in the top-ranked documents so that users do not have to look far down the list of retrieved documents in order to find ones that are relevant. In other words, it is desirable for some information seeking tasks to provide very high precision among the first k documents, where k is small (on the order of 5 - 30 documents), even at the expense of high recall. This goal is in contrast with tasks such as TREC, in which precision and recall are considered equally important, and systems are evaluated at cutoff depths up to 1000 documents [9].

To summarize, two important issues for interactive information access systems for casual users are

(i) casual users tend to issue short queries in simple format, but current evaluation has focused on producing systems that can depend on long queries to achieve strong results, and

(ii) casual users prefer to have a few relevant documents ranked uppermost, but current evaluation is influenced by recall at very large document cutoff levels.

In our earlier discussions of main topic and subtopic structure in full-text documents [12, 14], we have stressed the importance of recognizing the relationship between the terms of the query and the topic structure of the documents in the collection. Especially in a full-text document, there may be a subtopic discussion that is relevant to a query even if the main topic of the document does not directly address its concerns. Thus, we are interested in developing ranking strategies that capture the relationship between the query and the subtopic structure of the retrieved documents. Here we suggest doint this by computing how the overlap and distribution of query terms in the retrieved documents.

The way we capture the effect of term overlap is to structure the user query into a set of components, each component representing one topic of the query, and then retrieve documents that have at least one subtopic that contains a discussion of each of the query components. We implement this strategy by making use of two very simple constraints, or filters, and using them in conjunction with a vector space ranking.

Thus, in this paper we present a method that takes subtopic structure into account to address both issues listed above. This method results in greatly improved precision at small document cutoff levels given only short, simple queries. This is done by applying two constraints; the first constraint requires some simple specification on the part of the user, and the second is completely automated. The constraints filter out a subset of the collection, leaving the remaining documents to be ranked using a standard bag-of-words method, such as the vector space model. The filtering procedure is intended to cause relevant documents to appear higher up in the ranking than they would if no filtering were applied. Experiments in this paper show significant improvement in precision at fixed document cutoff levels for most of the queries in the test set when compared against a standard vector space ranking.

Several studies have shown that adding structure to queries, e.g., with the extended Boolean model [26, 28] and with a network-based model [29], can yield stronger results than either Boolean search or the vector space model. These and related methods focus on ways to relax the stringent requirements of Boolean search [20] in order to achieve a ranking. The work presented here differs in at least two important ways. First, we are interested in taking into account issues related to term overlap and distribution in full-text documents. The extended Boolean models in general do not address these concerns. Second, our experiments are run against a very large collection (> 3 gigabyte) consisting of full-text documents, as opposed to the very small test collections consisting mainly of titles and abstracts used in most earlier evaluations of this kind. Another potential difference is the emphasis in this work on simplicity of specification of the required structure, and use of standard, well-known, search and ranking methods.

The next section describes the two kinds of constraint. The first is a simple Boolean construct which requires the user to specify the query as a list of topics. This list is converted into a conjunct of disjuncts by the system. The second constraint is a simple proximity constraint imposed over the components of the query. The successful application of these constraints also has important ramifications for the TileBars graphical user interface [13] which is also described. Section 3 presents the ranking algorithm in summary form. Section 4 describes experiments using this algorithm on queries from the TREC collection, showing that these constraints almost consistently improve precision at low document cutoff levels when given a short user query. Finally, Section 5 discusses these results and their relationship to other work.

2 The Constraints

2.1 A Simple Boolean Filter

The first constraint, or filter, is needed in order to ensure that all the necessary components of the query are represented in the retrieved documents. To bring this about, we propose the use of a Boolean query that consists of a conjunct of disjuncts (otherwise known as conjunctive normal form), in which each disjunct component represents one topic in the query. This can be accomplished in the user interface in a very simple way: each topic (disjunct) is listed on one line, and a set of lines together act as a conjunction of these disjuncts. Our preliminary user studies [11] suggest that users are able to adapt to this format easily, although we have not yet formally shown this to be the case.

Once this idea is adopted, we can determine its utility empirically. In the experiments reported here this constraint improved or kept constant the performance for 27 out of 29 queries (in Experiment 1) and 22 out of 25 queries (in Experiment 2). The results must depend in part on how the query was formulated, but it should be noted that in Experiment 1, the query construction was done by the author in an estimated average of 90 seconds per query.

2.2 An Example

Below we show an example of a conversion of a short TREC topic description to a short query in conjunctive normal form. The original topic description (number 237) is:

> *Identify alternative sources of energy for automobiles. Include additives to gasoline that either decrease pollution or reduce oil consumption.*

This topic description can be broken down into its subcomponents: *pollution reduction* and *alternative energy* and *automobiles*. As a practical consideration, since there are many ways to express the concept of alternative energy automobiles, this in turn can be broken down still further so that synonyms can be listed, e.g., *alternative energy, electric, battery, solar* and *automobile, car, vehicle*, in order to give the system a better chance of finding a match (addition of automatically or partially-automated term expansion may be helpful at this point). This process might result in a query of the form:

(AND
  {OR *solar alternative energy*}
  {OR *pollution clean*}
  {OR *car automobile vehicle*})

This formulation is meant to capture the crucial aspects of the topic description, namely, alternative energy options, pollution, and cars. Breaking down the query explicitly indicates which components of the query are required in a relevant document. It also invites the user to expand each topic with additional terms, but without requiring the explicit formulation of a Boolean query.

There are, of course, other ways to express a topic besides listing synonyms. If the contents of the collection have been assigned subject or category labels, these can in some cases substitute for a list of synonyms, or can be listed in addition to the terms of interest. Research on bibliographic databases has suggested that neither subject labels or free text alone perform as well as the combination of the two when searching that kind of collection [22, 15, 19]. Furthermore, there are cases in which the user wants to require the presence of a very specific concept that may well not be represented by those pre-assigned categories. Whether each query topic is expressed by pre-defined categories or by synonym groups the point to be emphasized here is that the query should be specified as a conjunction of all of its important topics.

2.3 A Simple Proximity Filter

The second constraint is on collocation of occurrence of the components of the Boolean query. As described in the Introduction, This idea is motivated by our earlier discussions of main topic and subtopic structure in documents [12, 14]. We have stressed the importance of recognizing the relationship between the terms of the query and the topic structure of the documents in the collection. We have introduced an algorithm called *TextTiling* [12] that automatically segments long documents into multi-paragraph subtopical units. The use of this algorithm is not required for the ideas discussed in this paper, however; paragraphs or fixed-length blocks of text can be substituted for TextTiles.

The proximity constraint suggested here is one that requires a highly-ranked document to contain at least one subtopical segment with a representative of each of the topics in the query. Subtopical segments tend to be large, on the order of 100 to 300 words, so this is a much larger proximity window than has been explored in most experiments on proximity constraints

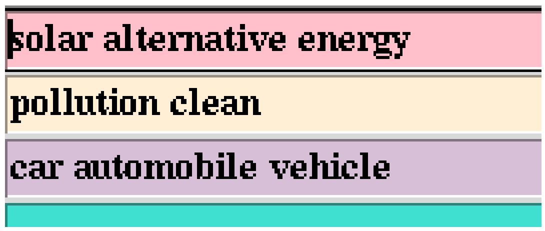

Figure 1: The entries into which users enter termsets, which are then treated as a conjunct of disjuncts.

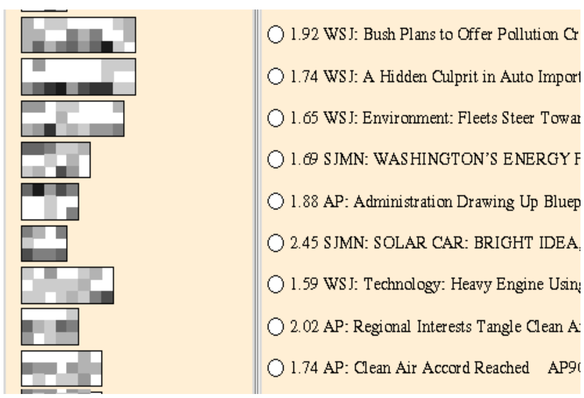

Figure 2: The TileBar Display on a query consisting of the three termsets shown in Figure 1.

(e.g., [18, 8]) that focus on proximity ranges on the order of 10 words.

Thus, the first constraint requires that documents contain at least one representative of each topic in the query. The second constraint requires that these representatives all co-occur within at least one subtopic segment of each document; in other words, that their discussion must overlap, rather than be distributed throughout the document.

It is important to realize that this strategy is different than the "best segment" or "best passage" strategies that have been explored recently in the literature ([14, 31, 25, 24, 4]), in that documents are not assigned a rank based on how well the best segment matches the query; rather, the algorithm simply eliminates from initial consideration documents that do not have any promising subtopic units at all.

2.4 Use in Interactive Information Access

This kind of query specification is used in the *TileBars* graphical display [13] to aid users in the interpretation of retrieval results. TileBars simultaneously and compactly indicate:

(i) the relative length of the document,

(ii) the frequency of the query terms in the document, and

(iii) the distribution of the terms with respect to the document and to each other.

In the interface, each disjunct is known as a *termset*, and users specify their queries as a list of termsets. Figure 1 shows how a list of termsets appears, using the example discussed in the previous subsection. Figure 2 shows some results of running this query on the TREC/TIPSTER collection. The graphical representation works as follows. Each rectangle represents a document. Each row of the rectangle represents the corresponding termset in the query display, i.e, the top row corresponds to *solar, alternative* or *energy*, the second row to *pollution* or *clean*, and the third row to *car, automobile* or *vehicle*. The rectangles are also subdivided into columns, where each column represents a text segment, as described above. Thus, the leftmost column indicates the first segment, or paragraph, of the document, the column to the right of this indicates the second segment of the document, and so on.

Each square represents the number of hits for the corresponding termset in the corresponding document segment. The darkness of the square indicates the number of times the query occurs in that segment of text; the darker the square the greater the number of hits (white indicates 0, black indicates 8 or more hits, the frequencies of all the terms within a term set are added together). Thus the user can quickly see if some subset of the terms overlap in the same segment of the document, and can see at what position of the document this overlap occurs.

For example, in Figure 2, which shows TileBars for nine documents that have been retrieved in response to the query of Figure 1, the first document discusses all three topics at some length. The second document focuses more on other topics related to autos, although there is a discussion of pollution issues towards the end. The fifth document has one subtopic discussion of alternative energy and pollution and automobiles, apparently in the context of a larger discussion of solar or other forms of energy. By contrast, the last two documents seem to discuss auto pollution issues with only a passing reference to the related notion of alternative energy.

Other systems have posited the use of a line-oriented query specification that corresponds to conjunctive normal form. For example, the Grateful Med system for medical abstracts [16] has the user enter the query as a sequence of lines of words. The EUROMATH system [23] requires the user to specify queries as a conjunct of disjuncts in a database-like form, with the AND and OR keywords marked explicitly. The AI/STARS system [1] has a clever graphical interface which converts a two-dimensional query specification into a Boolean formula that can take on a much more complicated in form than that used here. To our knowledge, however, no researchers have performed experiments that explictly compare users' performance on full-text systems with such a formulation to a vector-space or related ranking method's performance.[1]

3 The Algorithm

Our goal is to optimize the ranking for the first k documents, where k is small enough that users need not look far to find relevant documents. However, in some circumstances, for example, to aid in the evaluation of this algorithm, it is necessary to retrieve a fixed number n of documents, even if there are fewer than n documents that pass through the filters. One way to satisfy this requirement is to use multiple ranking strategies. In this case, it is suggested that the high-precision ranking strategy is used for the first k documents, and then the remaining $n-k$ documents are ranked using the standard vector space method. Thus, if the filters block out too many documents, those documents that did not meet the constraints are still allowed to take part in the ranking, albeit lower down in the list.

In summary, we suggest the following retrieval strategy:

1. The documents are divided into subtopic-sized units in advance.

2. The user specifies the query as a list of topics, one per line.

3. The system converts this into a Boolean query; a conjunct of disjuncts.

4. The system imposes a proximity constraint of one subtopical segment over this Boolean formula.

5. The system filters out documents that do not have at least one subtopical unit with representatives from all components of the conjunct.

6. The system ranks the remaining documents according to a standard vector space model.

7. If the resulting ranked list is too short, the system appends the highest-ranked documents that have not yet appeared in the results until the required number of documents has been reached.

4 Experiments

This section reports on two experiments that evaluate the performance of the algorithm described in Section 3. The first experiment uses queries created by the author and the second uses queries derived from those formulated by subjects in a separate study [11]. The next subsections describe the information access system used, the collection and topics, results using the author-generated queries and results for the subject-generated queries.

4.1 The System

For these experiments we used the PARC TDB (Text DataBase) system [7], implemented in Common LISP and CLOS. TDB provides a standard vector space weighting and ranking scheme (similarity search), similar to that reported in [3], and standard Boolean search. It

[1] Hersh *et al.* [16] compare performance of subjects using Grateful Med to those using a system that accepts natural language input, finding no difference between the two, but the each system was tested using different query formulations.

also has an extension to the vector space ranking which we call stuctured similarity search. This provides support for TileBars and the experiments described here. Structured similarity search ranks documents in the same way as does standard similarity search, but it additionally returns a list of term offsets that correspond to each component of the structured query discussed above.

4.2 The Collection

We experimented over the very large (> 3 MB) standard reference collection, the TIPSTER collection [9] (consisting largely of newswire, magazine articles, and government documents). Associated with this collection is a set of topic descriptions (referred to here as queries), and relevance assessments based on judgments made by information access experts. These were originally created for the Text REtrieval Conference (TREC) sponsored by NIST [9]. Queries taken from TREC-2 and 3 are evaluated against disks 1 and 2 of the collection, and queries taken from TREC-4 are evaluated against disks 2 and 3, as these disks correspond to the available relevance judgments. Each disk contains approximately 1 gigabyte of text data; thus the results of these experiments can be considered to scale. However, collections consisting of more specific subject areas, e.g., medical or legal text collections, might yield different results, such as a preference for different kinds of constraints.

4.3 Experiment 1

4.3.1 The Author-Generated Queries

The first experiment was run using 30 TREC queries. Since the data from each year of TREC has somewhat differing characteristics, ten queries were chosen from each of TREC-2, 3 and 4 in order to control for any differences in query length, number of possible relevant documents, etc. Queries were not selected based on any particular features; rather we simply used the first 10 queries from each year (choosing 10 queries from each TREC year at random would also have been appropriate).[2]

The TREC queries were converted into a form compatible with that described above. Some queries were assigned words that do not appear in the original TREC topic description. Appendix A shows the queries as formulated for the runs reported here. The vector space rankings, which act as baselines for these experiments, make use of exactly the same words, without the Boolean constraint.

The queries used in these experiments averaged 9.87 words in length. By comparison, the full queries[3] for TREC-2 and TREC-3 averaged 82.7 words per query. The unaltered TREC-4 queries, however, only contain 15.2 words per query (many of these words are not "content" words, e.g., question words such as *how* and *why*, and closed-class words). In TREC-4, the queries were designed to be shorter in order to be more "realistic."

As mentioned above, the query construction was done by the author in, on average, an estimated 90 seconds per query. Development of the constraints was done on a different set of queries with only three queries overlapping with the test set shown here.

Several of the queries call for particular two-word phrases, e.g., *surrogate mothers* and *social security*, and so the constraints are used here partially to force some kind of cooccurrence, where a simple phrasal proximity constraint could have been used instead. However, this specification is still more flexible than a strict phrase requirement, since the second termset for Query 70 is *mother motherhood* instead of *mother* alone, thus allowing for hits on variations of the phrasing of *mother*, and thus loosening what may otherwise be too tight a Boolean constraint.

For these experiments, subtopic segments were simple fixed-length contiguous blocks of 100 tokens each, since TextTiles are not integrated into the current version of the TDB system.

The precision of the results of each is evaluated at several document cutoff levels. In some cases, not enough documents are found that pass through both constraints to allow for evaluation at the cutoff level. To remedy this, when-

[2] Actually, only 29 queries appear in the results and Appendices; after the experiments were run, the TREC organizers realized they had to throw out query number 201.

[3] Using only the Description and Narrative parts, omitting the Summary and Concept parts which appear in TREC-2 queries but not in TREC-3 queries.

	Cut-off	No Constr (Baseline)	Bool	% Impr	p	Bool-Prox	% Impr	p
TREC-2	5	.56	.58	4	.6	.70	**25**	.009
	10	.52	.59	**13**	.009	.61	17	.2
	20	.49	.52	**6**	.05	.55	12	.1
	30	.44	.50	**14**	.003	.51	16	.08
	100	.34	.39	**15**	.03	.37	8	.2
TREC-3	5	.28	.36	29	.2	.68	**143**	.008
	10	.37	.42	**14**	.05	.60	**62**	.005
	20	.40	.45	13	.07	.55	**38**	.03
	30	.36	.44	**22**	.04	.50	**39**	.04
	100	.30	.37	**23**	.04	.35	16	.09
TREC-4	5	.38	.33	-13	.6	.29	-23	.3
	10	.30	.31	3	.7	.34	13	.3
	20	.24	.33	**38**	.03	.32	**33**	.04
	30	.23	.35	**52**	.0006	.26	13	.3
	100	.19	.21	11	.4	.16	-16	.3

Table 1: Results for Experiment 1 showing average precisions and percent improvements over the baseline for Tables 3 and 4. ***No Constr*** **indicates no constraints applied to the vector space ranking, and acts as the baseline.** ***Bool*** **indicates vector space ranking after applying the simple Boolean filter (a conjunct of disjunts), and** ***Bool-Prox*** **indicates vector space ranking after applying the Boolean proximity constraint. Percentage improvements over the baseline that are significant (paired t-test) are shown in bold.**

ever there are not enough documents for a given cutoff level, the top-ranked documents (according to the vector space model) that have not yet appeared are appended to the ranking, as in Step 7 of the algorithm in Section 3. These cases are notated in the per-query results (Tables 3 and 4).

4.3.2 Results

Table 1 summarizes the results for each set of queries in the form of averages and percent improvement. Tables 3 and 4 show the results in detail at various document cutoff levels for the three sets of queries. Each of these tables shows the precision for three different ranking strategies. The first score is the result of using all of the terms as shown in Appendix A but using no constraints; the retrieved documents are simply ranked according to the standard vector space formula. The second score is the result of first applying the Boolean constraint and ranking the documents that pass through this filter with the same vector space ranking. The third score is similar to the second, except the Boolean-proximity constraint is applied instead of the Boolean constraint alone.

Looking first at the averages and percent improvement for TREC-2 and 3, we see that there is a consistent improvement at all cutoff levels, and in many cases this percentage improvement is significant ($p < .05$). The TREC-3 queries see very large improvements, over 100% at one cutoff level; such strong percentage improvements are difficult to achieve for TREC-related experiments.

The table below compares the TREC-3 scores to those of the corresponding full queries, which have an average length of 85 words:

Cut-off	Full Query	No Constr	Bool	Bool-Prox	% Impr
5	.48	.28	.36	.68	42
10	.47	.37	.42	.60	28
20	.47	.40	.45	.55	17
30	.44	.36	.44	.50	14
100	.32	.30	.37	.35	9

We see a consistent improvement using the two constraints even though the full queries have on average more than eight times the number of words.[4]

Turning now to the TREC-4 runs, we see less encouraging results. The scores in general are lower for TREC-4 than for TREC-2 and 3, and the constraints degrade the results at some cutoff levels (although they significantly improve the results at cutoff 20). The problem seems to be that the Boolean constraint was too strong; seven out of nine of the the TREC-4 queries were formulated with three termsets, but most of the TREC-2 and TREC-3 queries had only two termsets. In the few cases where three termsets were used, especially notable improvements in precision seem to result: for example, queries 66, 152, and 159 all see remarkable improvements over the baseline when the constraints are employed. However, this does not seem to be the case for the TREC-4 queries.

As will be seen in Experiment 2, the constraints can be shown to significantly improve the scores for TREC-4, using even shorter query descriptions than in Experiment 1, when making use of only two termsets. Futhermore, the averages for Experiment 1, shown in Table 1, are evaluated over only nine queries; in Experiment 2 we make use of 25 queries.

4.4 Experiment 2

4.4.1 The Subject-Generated Queries

One potential objection to Experiment 1 is that the queries were formulated by the author, and that naive users' queries might be shorter or in some other manner less amenable the filtering and ranking techniques employed. To address this issue, we performed another experiment which used "real" queries, derived from those supplied by subjects in another study [11]. In this study, subjects were asked to use a graphical user interface that included, among other display modalities, a version of the TileBar display. Subjects were requested to enter their queries, those used for the TREC-4 interactive track, in the line-by-line format as described above. The subjects consisted of four University of California Berkeley graduate students, each of whom executed 13 queries. These consisted of 12 of the 25 required for the interactive track of TREC-4, as well as one extra query given to all four participants.

However, there was a critical difference in the way the termsets were used in that study. The query was **not** cast into conjunctive normal form. Rather, the entries in the termsets were treated as one bag-of-words and used as input to the same vector space search and ranking method as used above. The termsets were used in two ways: (i) to show the hits in the TileBar format and (ii) to rerank the results of the vector space search when showing the documents to the subject in TileBar mode. They were not used as a filter and no Boolean constraints were imposed.

In most cases the subjects created four termsets (the average was 3.84 over 97 query formulations), most likely because there were four entries available and the more termsets used, the more detailed the TileBar display. The queries were quite short, consisting of 5.38 words on average. For comparison, the unaltered TREC-4 queries averaged 16.84 words per query (or 11.52 words per query when words on the TDB stoplist are removed, since these queries consist of full sentences in most cases). For comparison, the queries generated by the author for the nine TREC-4 queries averaged 8.25 words and 2.75 termsets per query.

So at an average of 5.38 words and 3.84 termsets per query, direct application of the conjunct of disjuncts becomes almost a straight conjunction, which is often too restrictive. For this reason, and because the queries were not originally devised to satisfy the constraints explored here, they were modified very slightly. One subject-specified version of each query was chosen arbitrarily and modified so it consisted of exactly two termsets. Every query had at least two termsets already, and those having more were modified by the author by combining pairs of adjacent termsets. No reordering was done, even though at times it appeared that reordering would help the results by combining related terms together. This was not done in order to minimize the amount of interference to the original query.

For example,

(AND
    {OR *firearm gun weapon*}

[4] The percent improvements are significant ($p < .02$) when all are compared, but at each cutoff level they are not.

Doc Cutoff	No Constr (B1)	Full (B2)	Bool	% B1	p	% B2	p	Bool-Prox	% B1	p	% B2	p
5	.22	.30	.30	36	.07	0	1.0	.43	**96**	.0006	**43**	.01
10	.22	.30	.32	**46**	.008	6	.7	.41	**86**	.0005	**37**	.009
20	.25	.26	.31	**24**	.008	**19**	.05	.35	**40**	.01	**35**	.02
30	.23	.24	.29	**26**	.008	**21**	.05	.31	**35**	.01	**29**	.05
100	.17	.17	.22	**29**	.01	**29**	.05	.20	18	.2	18	.3

Table 2: Results for Experiment 2. Precision and percent improvements over the baselines for 25 queries generated by subjects (average length 5.38 words) at five cutoff levels. *Baseline1* shows the precision for vector space ranking with no constraints using the user-specified query. *Baseline2* shows the precision for vector space ranking using the original full TREC topic description (average length 16.84 words) and no constraints. % B1 and % B2 show the percentage improvements over the indicated baselines, for which the constraints combined with ranking yield significantly better results. Percentage improvements over the baseline that are significant (paired t-test) are shown in bold.

{OR *crime criminal*}
{OR *ammunition sale* }
{OR *correlation connetion* })

was transformed to

(AND
{OR *firearm gun weapon*}
{OR *crime criminal ammunition sale correlation connetion* })

(retaining original spelling errors) even though it would make more sense to group *ammunition* with *firearm gun weapon*.

The queries in their original and modified forms appear in Appendix B.

4.4.2 Results

Table 2 shows the overall results of this experiment. There are two baselines. The first is based on the results of the vector space ranking using the slightly modified user queries. The percent improvement for application of (i) Boolean constraint only and (ii) the Boolean and the proximity constraint are shown. Again, the improvements over the baseline are substantial and significant, ranging from 18 to 96%. As the document cutoff level increases from the top 5 documents to the top 100, there is a decrease in percentage improvement over the vector space ranking. At the 100-document cutoff level, the single constraint performs better than both constraints together.

The second baseline is based on the scores obtained when using the vector space ranking on the query as originally specified in TREC-4. These are on average almost twice as long as the user-specified queries (11.52 *vs.* 5.38 content words). Without constraints the short queries fare far worse at lower cutoff levels and about the same as the full queries at higher cutoffs. The shorter but more constrained queries perform the same as or better than the full queries at all cutoff levels, with percent improvement ranging from 18 to 43%.

As can be seen from these results, the short queries derived by subjects in an interactive setting can be made to yield much stronger results using the algorithm suggested here than using the query as originally formulated. At this point we can discuss the poor showing in Experiment 1 on the TREC-4 queries. As mentioned above, the culprit seems to be queries whose Boolean specification is too specific. Below is shown the results of the first nine TREC-4 queries using the subject-specified formulations as in Table 2.

Cut-off	No Constr	Bool	% Impr	Bool-Prox	% Impr
5	0.29	0.29	0	0.44	52
10	0.26	0.28	8	0.42	62
20	0.28	0.33	18	0.37	32
30	0.27	0.31	15	0.32	19
100	0.20	0.22	10	0.18	-10

As can be seen in this table, up to the 100-level cutoff, use of the two constraints yields strong improvements in precision. The comparison between these results and those of the TREC-4 queries in Experiment 1 (see Table 4) indicates a possible need for a modification of the algorithm to incorporate some kind of backing-off strategy that favors satisfaction of more termsets over fewer, but gracefully recovers when representatives from all topics are not present. This is related to the notion of *quorum-level* search [26].

5 Discussion

5.1 Relation to Other Work

Jing and Croft [17] experimented with how to improve queries by making use of term expansion with phrasal collocations. Two of their results are of interest to this work. First, they found that they did not get strong improvement using term expansion on the full (long) TREC topic descriptions. To see the benefits of the term expansion, they found it necessary to restrict the TREC topic descriptions to use only the short Description part. Second, they found strong improvements when they simply duplicated some of the phrases in the query, quite independent of adding new terms. Furthermore, the most effective phrases were single-word phrases. In other words, they improved results by assigning greater weight to important terms and thus force their system to produce a better ranking.

It may be the case that such duplication yields a similar effect to the Boolean constraint imposed here; identification of important terms are made and then highly ranked documents must contain those terms.

Some independent work related to our study has been reported by Clarke *et al.* [5] on the manual ad hoc track in TREC-4. For this track, the researchers running the experiment are allowed to spend time formulating the query manually; in this case the researchers spent 15 to 45 minutes per query. The queries were specified as an ordered list of topics, called subqueries, and most subqueries were specified as disjuncts. Each subquery was ranked separately, and then the ranks were combined according to the ordering of the subquery list, with the results of a particular subquery ranked before the results of subsequent subqueries. This yields what might be thought of as a psuedo-Boolean filter. The proximity constraint is somewhat different – instead of being fixed, the subqueries were ranked according to the inverse of the distance between the elements that compose the subqueries. Thus elements that appear closer together yield a higher ranking. The results obtained using this strategy for the manual ad hoc using this approach were quite strong; however, the average query length was long at 67 words per query. The use of the inverse distance of the terms satisfying subparts of the query is a clever idea and might prove to be a good alternative to the fixed length proximity constraint used in the algorithm described here.

As discussed above, other researchers have looked at alternative ways to rank the results of Boolean queries [29, 28]. However, here we have suggested something different, namely, that a Boolean query be used as an initial filter, but then vector space or other ranking mechanism used as usual.

5.2 Summary and Future Work

Interactive information access systems intended for use by casual users should produce effective results given only very short, simple queries, and (in most cases) should focus on ensuring high precision among the first few documents shown, rather than emphasizing full recall. This paper has introduced a method to enhance short user-specified queries, and results obtained over 45 TREC queries show significant, consistent improvements over ranking the same queries in a standard way. The main idea behind the method is to find documents that contain at least subtopical discussions of all of the important components of the user query, and the method makes use of standard search algorithms. Because the results are strong for very short queries, they have important ramifications for interactive systems intended for casual users which should not require long, complex queries for effective results.

In future work we plan to investigate the impact of other kinds of filters having to do with term distribution and overlap. It may be the case that these attributes may be best used as part of the ranking, rather than as binary filters, and we plan to use a logistic regression to ascertain how best to weight the various attributes. As described in Section 4, we may need to loosen the Boolean constraint by using an ordered partial match strategy wherein documents with all k termsets are ranked higher than those with $k-1$, and so on, in the spirit of a quorum-level strategy. We also plan to try to find a way to make use of the inverse distance measure of Clarke *et al.* [5] instead of the fixed proximity constraint. The effects of relevance feedback [27] on these results also remains to be evaluated.

It has been often observed that different queries respond better to different retrieval methods [2]. We have preliminary evidence that shows that different queries work better with different constraint types. This effect could be seen to some extent in the results presented in this paper, in that some queries are more strongly improved by the constraints than others, and a few were adversely affected. The difficult question remains, however, of how to automatically determine which kinds of constraints are most applicable for which queries. One solution to this problem is to provide the user with intuitive, descriptive interfaces that indicate the relationship between the query and the retrieval results. Thus, this kind of filtering mechanism should be used in conjunction with information access visualization tools, such as TileBars. These tools can help provide a framework for our contention that multiple ranking and display options should be available to the user in an interactive system.

Acknowledgements

Jan Pedersen and David Hull provided very useful feedback on the ideas presented in this paper. Additionally Jan Pedersen provided many suggestions that greatly improved the presentation, Arnie Rosenthal provided a useful suggestion, and two anonymous reviewers provided helpful suggestions for improvements and additional references.

References

[1] P. Anick, J. Brennan, R. Flynn, D. Hanssen, B. Alvey, and J. Robbins. A direct manipulation interface for boolean information retrieval via natural language query. In *Proceedings of the 13th Annual International ACM/SIGIR Conference*, pages 135–150, Brussels, Belgium, 1990.

[2] N. J. Belkin and W. B. Croft. Retrieval techniques. In Martha E. Williams, editor, *Annual Review of Information Science and Technolgy*, pages 109–145. Elsevier Science Publishers, 1987.

[3] Chris Buckley, James Allan, and Gerard Salton. Automatic routing and ad-hoc retrieval using SMART: TREC 2. In Donna Harman, editor, *Proceedings of the Second Text Retrieval Conference TREC-2*. National Institute of Standards and Technology Special Publication 500-215, 1994.

[4] James P. Callan. Passage-level evidence in document retrieval. In *Proceedings of the 17th Annual International ACM/SIGIR Conference*, pages 302–310, Dublin, Ireland, 1994.

[5] Charles L. A. Clarke, Grodon V. Cormack, and Forbes J. Burkowski. Shortest substring ranking (multitext experiments for TREC-4). In Donna Harman, editor, *Proceedings of the Fourth Text Retrieval Conference TREC-4*. National Institute of Standards and Technology Special Publication, 1996. (to appear).

[6] W. Bruce Croft, Robert Cook, and Dean Wilder. Providing government information on the internet: Experiences with THOMAS. In *Proceedings of Digital Libraries '95*, pages 19–24, Austin, TX, June 1995.

[7] Douglass R. Cutting, Jan O. Pedersen, and Per-Kristian Halvorsen. An object-oriented architecture for text retrieval. In *Conference Proceedings of RIAO'91, Intelligent Text and Image Handling, Barcelona, Spain*, pages 285–298, April 1991. Also available as Xerox PARC technical report SSL-90-83.

[8] Stephanie W. Haas and Robert M. Losee Jr. Looking in text windows: Their size and composition. *Information Processing and Management*, 30(5):619–629, 1994.

[9] Donna Harman, editor. *Proceedings of the Third Text Retrieval Conference TREC-3*. National Institute of Standards and Technology Special Publication 500-225, 1995.

[10] Donna Harman, editor. *Proceedings of the Fourth Text Retrieval Conference TREC-4*. National Institute of Standards and Technology Special Publication, 1996. (to appear).

[11] Marti Hearst, Jan Pedersen, Peter Pirolli, Hinrich Schüetze, Gregory Grefenstette, and David Hull. Four TREC-4 Tracks: the Xerox site report. In Donna Harman, editor, *Proceedings of the Fourth Text Retrieval Conference TREC-4*. National Institute of Standards and Technology Special Publication, 1996. (to appear).

[12] Marti A. Hearst. Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd Meeting of the Association for Computational Linguistics*, June 1994.

[13] Marti A. Hearst. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Denver, CO, May 1995. ACM.

[14] Marti A. Hearst and Christian Plaunt. Subtopic structuring for full-length document access. In *Proceedings of the 16th Annual International ACM/SIGIR Conference*, pages 59–68, Pittsburgh, PA, 1993.

[15] Rolf G. Henzler. Free or controlled vocabularies: Some statistical user-oriented evaluations of biomedical information systems. *International Classification*, 5(1):21–26, 1978.

[16] William R. Hersh, Diane L. Elliot, David H. Hickam, Stephanie L. Wolf, and Anna Molnar. Towards new measures of information retrieval evaluation. In *Proceedings of the 18th Annual International ACM/SIGIR Conference*, pages 164–170, Seattle, WA, 1995.

[17] Yufeng Jing and W. Bruce Croft. An association thesaurus for information retrieval. In *Proceedings of RIAO*, pages 146–160, Rockefeller University, New York, 1994.

[18] E. Michael Keen. Term position ranking: some new test results. In *Proceedings of the 15th Annual International ACM/SIGIR Conference*, pages 66–76, Copenhagen, Denmark, 1992.

[19] F. Lancaster. *Vocabulary Control for Information Retrieval, Second Edition.* Information Resources, Arlington, VA, 1986.

[20] Ray R. Larson. Evaluation of advanced retrieval techniques in an experimental online catalog. *Journal of the American Society for Information Science*, 43(1):34–53, 1992.

[21] X. Allan Lu and Robert B. Keefer. Query expansion/reduction and its impact on retrieval effectiveness. In Donna Harman, editor, *Proceedings of the Third Text Retrieval Conference TREC-3*, pages 231–239. National Institute of Standards and Technology Special Publication 500-225, 1995.

[22] Karen Markey, Pauline Atherton, and Claudia Newton. An analysis of controlled vocabulary and free text search statements in online searches. *Online Review*, 4:225–236, 1982.

[23] G. McAlpine and P. Ingwersen. Integrated information retrieval in a knowledge worker support system. In *Proceedings of the 12th Annual International ACM/SIGIR Conference*, pages 48–57, Cambridge, MA, 1989.

[24] Elke Mittendorf and Peter Schäuble. Document and passage retrieval based on hidden markov models. In *Proceedings of the 17th Annual International ACM/SIGIR Conference*, pages 318–327, Dublin, Ireland, 1994.

[25] Alistair Moffat, Ron Sacks-Davis, Ross Wilkinson, and Justin Zobel. Retrieval of partial documents. In Donna Harman, editor, *Proceedings of the Second Text Retrieval Conference TREC-2*, pages 181–190. National Institute of Standards and Technology Special Publication 500-215, 1994.

[26] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer.* Addison-Wesley, Reading, MA, 1989.

[27] Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *JASIS*, 41(4):288–297, 1990.

[28] Gerard Salton, Edward A. Fox, and Harry Wu. Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022–1036, November 1983.

[29] Howard Turtle and W. Bruce Croft. Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems*, 9(3):187–222, 1991.

[30] Ellen M. Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of the 17th Annual International ACM/SIGIR Conference*, pages 61–69, Dublin, Ireland, 1994.

[31] Ross Wilkinson. Effective retrieval of structured documents. In *Proceedings of the 17th Annual International ACM/SIGIR Conference*, pages 311–317, Dublin, Ireland, 1994.

Doc Cutoff	Topic Num	No Constr	Bool	Bool Prox	Topic Num	No Constr	Bool	Bool-Prox
5	61	0.80	0.80	1.00	66	0.40	0.20	0.60
10	61	0.90	0.90	1.00	66	0.20	0.30	0.60
20	61	0.90	0.90	0.95	66	0.25	0.35	0.40
30	61	0.80	0.87	0.87	66	0.20	0.33	0.37
100	61	0.58	0.72	0.74	66	0.15	0.18	0.18
5	62	0.40	0.40	0.80	67	0.80	0.80	0.80
10	62	0.20	0.30	0.60	67	0.70	0.70	0.70
20	62	0.35	0.40	0.55	67	0.55	0.55	0.60
30	62	0.33	0.37	0.60	67	0.53	0.53	0.53
100	62	0.36	0.41	0.47	67	0.49	0.51	0.57
5	63	0.60	0.60	0.60	68	0.60	0.80	0.80
10	63	0.70	0.80	0.40	68	0.60	0.80	0.70
20	63	0.50	0.50	0.35	68	0.60	0.70	0.60
30	63	0.37	0.47	0.27	68	0.50	0.60	0.57
100	63	0.17	0.19	0.13	68	0.45	0.48	0.33
5	64	0.40	0.40	0.40	69	0.40	0.40	0.40
10	64	0.30	0.30	0.30	69	0.20	0.20	0.30
20	64	0.35	0.35	0.40	69	0.10	0.10	0.20
30	64	0.40	0.40	0.37	69	0.13	0.13	0.13
100	64	0.44	0.44	0.49	69	0.08	0.08	0.11
5	65	0.40	0.40	0.60	70	0.80	1.00	1.00
10	65	0.50	0.60	0.70	70	0.90	1.00	0.80
20	65	0.50	0.50	0.65	70	0.80	0.85	0.75
30	65	0.47	0.53	0.60	70	0.70	0.80	0.77*
100	65	0.33	0.35	0.35	70	0.34	0.50	0.35*

Table 3: Precision on ten TREC 2 Queries for Experiment 1 (author-specified queries). ***No Constr*** **indicates the baseline, i.e., the results of vector space ranking, applying no constraints.** ***Bool*** **indicates vector space ranking after applying the simple Boolean filter (a conjunct of disjunts), and** ***Bool-Prox*** **indicates vector space ranking after applying the Boolean proximity constraint (at least one representative from each disjunct must co-occur within at least one subtopic segment). Asterisks indicate those queries and cutoff levels at which not enough documents passed through the filters and so the remainder of the ranking was filled out by documents ranked highly by the vector space model and not yet accounted for.**

Doc Cutoff	Topic Num	No Constr	Bool	Bool Prox	Topic Num	No Constr	Bool	Bool-Prox	Topic Num	No Constr	Bool	Bool-Prox
5	151	0.80	0.80	0.80	158	0.40	0.40	1.00	206	0.00	0.00	0.20
10	151	0.80	0.80	0.80	158	0.40	0.50	0.90	206	0.00	0.00	0.20
20	151	0.75	0.80	0.70	158	0.40	0.45	0.80	206	0.00	0.00	0.15
30	151	0.70	0.73	0.67	158	0.30	0.37	0.63	206	0.00	0.07	0.10
100	151	0.37	0.45	0.27	158	0.19	0.24	0.29	206	0.00	0.03	0.04
5	152	0.00	0.00	0.40	159	0.20	0.60	1.00	207	0.80	0.80	0.80
10	152	0.10	0.20	0.40	159	0.50	0.70	0.90	207	0.70	0.70	0.80
20	152	0.25	0.35	0.60	159	0.55	0.70	0.50	207	0.60	0.80	0.75
30	152	0.23	0.43	0.63	159	0.50	0.67	0.33	207	0.60	0.77	0.60
100	152	0.33	0.49	0.55	159	0.31	0.39	0.28*	207	0.41	0.51	0.41*
5	153	0.00	0.00	0.00	160	0.00	0.00	1.00	208	0.00	0.00	0.00
10	153	0.00	0.00	0.00	160	0.20	0.20	0.60	208	0.00	0.10	0.10
20	153	0.00	0.00	0.00	160	0.30	0.25	0.45*	208	0.05	0.05	0.05
30	153	0.00	0.03	0.00	160	0.20	0.20	0.33*	208	0.03	0.03	0.03
100	153	0.02	0.01	0.00	160	0.11	0.11	0.14*	208	0.07	0.04	0.03
5	154	0.80	0.80	1.00	202	0.60	0.00	0.00	209	0.40	0.40	0.40
10	154	0.90	0.90	1.00	202	0.30	0.20	0.10	209	0.20	0.20	0.30
20	154	0.90	0.90	0.95	202	0.25	0.20	0.15	209	0.20	0.30	0.40
30	154	0.83	0.83	0.97	202	0.20	0.33	0.10	209	0.23	0.37	0.33
100	154	0.76	0.77	0.84	202	0.26	0.17	0.16*	209	0.20	0.24	0.20
5	155	0.00	0.00	0.00	203	0.20	0.40	0.00	210	1.00	1.00	0.80
10	155	0.00	0.00	0.00	203	0.10	0.30	0.10	210	0.90	0.90	0.90
20	155	0.00	0.00	0.10	203	0.05	0.30	0.10*	210	0.70	0.85	0.75
30	155	0.00	0.00	0.13	203	0.07	0.30	0.07*	210	0.60	0.73	0.63
100	155	0.02	0.04	0.08	203	0.04	0.11	0.04*	210	0.27	0.40	0.28
5	156	0.60	0.60	0.80	204	0.40	0.40	0.40				
10	156	0.70	0.70	0.90	204	0.50	0.40	0.50				
20	156	0.75	0.75	0.90	204	0.30	0.40	0.45				
30	156	0.80	0.80	0.93	204	0.33	0.50	0.40				
100	156	0.82	0.85	0.93	204	0.46	0.39	0.24				
5	157	0.00	0.40	0.80	205	0.00	0.00	0.00				
10	157	0.10	0.20	0.50	205	0.00	0.00	0.10				
20	157	0.10	0.25	0.55*	205	0.00	0.05	0.05				
30	157	0.07	0.33	0.40*	205	0.00	0.07	0.03				
100	157	0.04	0.36	0.12*	205	0.01	0.03	0.02*				

Table 4: Precision on ten TREC-3 and nine TREC-4 Queries for Experiment 1 (query 201 was subsequently thrown out by the TREC organizers). Notations are as in Table 3.

A Appendix: Author-Specified Queries

```
; TREC 2 Queries
(setq query61 `({israel israeli tel-aviv mossad} {iran-contra iran scandal diversion arms}))
(setq query62 `({coup d'etat} {attempt success successful overthrow surrender revolt oust}))
(setq query63 `({ machine translation translation} {software application marketing product
    develop implementation market implement prototype company}) )
(setq query64 `({hostage kidnap kidnapper hijack}
    {negotiate trade exchange swap release attempt political government}))
(setq query65 `({information retrieval ir}
    {storage query text interface software application marketing
    product develop implementation market implement prototype company}) )
(setq query66 `({natural language nlp}
    {linguistic feature capability parser analyzer semantic syntactic syntax}
    {technology marketing product develop inc prototype}) )
(setq query67 `({protest riot sit-in uprising} {political government dissident politics }))
(setq query68 `({asbestos fiber fine-diameter} {health hazard lung cancer osha harmful}))
(setq query69 `({salt treaty} {limit revive ratify senate ceiling lobbyists lobby}
    {reagan buildup starwars star}))
(setq query70 `({surrogate} {mother motherhood}
    {law legal judicial lawyer court custody hearing}))
; TREC 3 Queries
(setq query151 `({jail prison inmate correctional}
    {overcrowding overcroweded capacity taxpayer cost}) )
(setq query152 `({defense military military-industrial contract}
    {impropriety cheating fraud bribery}
    {service product supplier contractor developer}))
(setq query153 `({insurance coverage} {long term care}))
(setq query154 `({oil spill accident} {gallon ton ship offshore holding tank shipborn}))
(setq query155 `({christian coalition fundamentalism southern baptist bible belt}
    {political power influence voter turnout}))
(setq query156 `({gun firearm rifle semi-automatic}
    {control legislation brady law second amendment restrictions}))
(setq query157 `({sclerosis auto-immune}
    {cure cause treatment research drug therapy healing heal}))
(setq query158 `({term} {limit senate consecutive representative} ))
(setq query159 `({electric alternate energy} {car vehicle}
    {development design developing underway building selling sell}))
(setq query160 `({vitamins a b c}
    {cure disease ailment preventative prevent improve success progress}))
; TREC 4 Queries
(setq query202 `({nuclear proliferation} {treaty treaties}
    {status violations monitor monitoring} ) )
(setq query203 `({tire} {recycle recycling} {economic impact cost}) )
(setq query204 `({nuclear} {power plant} {location rate production kilowatts}) )
(setq query205 `({paramilitary para-military} {activity group meeting exercise}) )
(setq query206 `({third} {party perot} {likelihood viability success succeed win}) )
(setq query207 `({quebec canada} {independence separatists separation}) )
(setq query208 `({bioconversion biological} {reuse recycle waste energy fertilizer}
    {recent development research breakthrough product adoption adopt}) )
(setq query209 `({social} {security}
    {stability instability broke bankrupt measure proposal}) )
(setq query210 `({medical needle} {waste needle}
    {illegal disposal dumping dump anti-dumping pollution} ) )
```

B Appendix: Subject-Specified Queries

The original query appears first, followed by its modification to have only two termsets.

```
(setq query202 `({nuclear} {treaty} {proliferation} {monitor}) )
(setq query202b `({nuclear treaty} {proliferation monitor}) )
(setq query203 `({recycle environment} {tire} {economics save} {impact}) )
(setq query203b `({recycle environment} {tire economics save impact}) )
(setq query204 `({nuclear power plant} {location list} {unite state} {production rate}) )
(setq query204b `({nuclear power plant} {location list unite state production rate}) )
(setq query205 `({paramilitary} {unite state} {terrorist} {domestic}) )
(setq query205b `({paramilitary} {unite state terrorist domestic}) )
(setq query206 `({political} {party} {third} {unite state}) )
(setq query206b `({political party} {third unite state}) )
(setq query207 `({quebec canada} {separatist} {independence} {prospect}) )
(setq query207b `({quebec canada} {separatist independence prospect}) )
(setq query208 `({conversion recycle} {plant} {waste} {development}) )
(setq query208b `({conversion recycle} {plant waste development}) )
(setq query209 `({social} {security} {bankrupt} {proposal}) )
(setq query209b `({social security} {bankrupt proposal}) )
(setq query210 `({medical waste} {disposal} {illegal} {solution solve}) )
(setq query210b `({medical waste} {disposal illegal solution solve}) )
(setq query211 `({dwi dui} {death} {law} {drink drive}) )
(setq query211b `({dwi dui} {death law drink drive}) )
(setq query212 `({copyright} {violation} {international} unite state}) )
(setq query212b `({copyright} {violation international unite state}) )
(setq query213 `({dna gene} {test} {crime murder} {convict sentence}) )
(setq query213b `({dna gene} {test crime murder convict sentence}) )
(setq query214 `({hypnosis} {self-induce self} {technique} {trick}) )
(setq query214b `({hypnosis self-induce self} {technique trick}) )
(setq query215 `({infant} {mortality} {cause} {unite state}) )
(setq query215b `({infant} {mortality cause unite state}) )
(setq query216 `({osteoporosis} {treatment} {prevention} {research}) )
(setq query216b `({osteoporosis} {treatment prevention research}) )
(setq query220 `({crossword puzzle} {puzzle maker}) )
(setq query220b `({crossword puzzle} {puzzle maker}) )
(setq query223 `({microsoft} {bill gate} {computer industry} {history}) )
(setq query223b `({microsoft bill gate} {computer industry history}) )
(setq query227 `({u.s.  military} {friendly fire} {accident foul play} {death kill}) )
(setq query227b `({u.s.  military} {friendly fire accident foul play death kill}) )
(setq query232 `({near-death} {report} {experience} {evaluation}) )
(setq query232b `({near-death} {report experience evaluation}) )
(setq query236 `({law regulation} {sea ocean} {disagreement conflict} {coastal}) )
(setq query236b `({law regulation} {sea ocean disagreement conflict coastal}) )
(setq query238 `({park} {national} {maintenance management}) )
(setq query238b `({park} {national maintenance management}) )
(setq query239 `({cancer} {nation} {rate} {cause}) )
(setq query239b `({cancer} {nation rate cause}) )
(setq query242 `({affirmative action} {equal opportunity}
    {construction build} {industry company}) )
(setq query242b `({affirmative action equal opportunity}
    {construction build industry company}) )
(setq query243 `({fossil fuel energy} {private co industry}
    {government federal gov} {restrict bar}) )
(setq query243b `({fossil fuel energy}
    {private co industry government federal gov restrict bar}) )
(setq query250 `({firearm gun weapon} {crime criminal}
    {ammunition sale} {correlation connetion}) )
(setq query250b `({firearm gun weapon}
    {crime criminal ammunition sale correlation connetion}) )
```