Improving Relevance Feedback in the Vector Space Model

Carol Lundquist
Department of Computer Science
George Mason University
Fairfax, VA 22030
clundqui@osf1.gmu.edu

David A. Grossman
Office of Information Technology
3W16 Plaza B
Washington, DC 20505
dgrossm1@osf1.gmu.edu

Ophir Frieder
Department of Computer Science
Florida Institute of Technology
Melbourne, FL 32901
ophir@cs.fit.edu

Abstract

Since the use of relevance feedback in information retrieval to improve precision and recall was first proposed in the late-1960's, many different techniques have been used to improve the results obtained from relevance feedback. Since most information retrieval systems performing relevance feedback use combinations of several techniques, the individual contribution of each technique to the overall improvement is relatively unknown. We discuss several techniques to improve relevance feedback including calibrating the number of top-ranked documents or feedback terms used for relevance feedback, clustering the top-ranked documents, changing the term weighting formula, and scaling the weight of the feedback terms. The impact of each technique on improving precision and recall is investigated using the Tipster document collection. We compare our work to a commonly accepted approach of using 50 words and 20 phrases for relevance feedback and show a 31% improvement in average precision over the commonly accepted approach when 10 feedback terms (either words or phrases) are used. In addition, we have identified a method which shows promise in predicting those queries which benefit from relevance feedback.

1. Introduction

As increasing amounts of text is available in electronic format, users are presenting information retrieval (IR) systems with a wide variety of information retrieval requirements. Since the users may have relatively little knowledge regarding the specific contents of the various information collections, there is a growing need to be able to modify the original user query into one which is optimized for a particular information collection and is more likely to retrieve documents considered relevant by the user. Relevance feedback offers a method to modify the original query based on characteristics of a particular information collection to improve precision and recall in the retrieval results.

Relevance feedback in IR systems was first proposed in the late-1960's (Rocchio71). Since then, many techniques have been proposed to tailor and improve the relevance feedback results. However, most systems using relevance feedback incorporate a variety of techniques; so the contribution to the overall improvement from any single technique is relatively unknown. Most of the previous work done on comparing different relevance feedback improvements techniques has been done using small, special purpose document collections (Salton90, Harman92), and there has been no systematic comparison of the techniques done against a large, standard collection of research data. To better understand the interaction between the different relevance feedback techniques, we focus on a variety of relevance feedback improvement techniques and determine the impact each technique has on improving precision and recall in an IR system.

Our IR system is based on the vector space model and implemented using a relational database management system (RDBMS). Until recently, relevance feedback had not been implemented in the relational model (Lundquist97). Using the relational model as the basis for an IR system not only allows complex retrieval techniques such as relevance feedback to be implemented, but it also can automatically and transparently provide parallelism when implemented in a parallel RDBMS.

This work has been done using the Tipster data collection which contains several gigabytes of documents obtained from various news feeds and other electronic document collections. The TIPSTER data collection is used as the basis of the TREC conference each year; so standard collections of queries along with listings of their relevant documents are also available. The combination of data, queries, and known results makes this collection suitable for experimentation.

2. Prior Work

Relevance feedback is a process through which a query is selectively modified to retrieve more relevant documents from a collection than the unmodified original version. The query can be modified by either adjusting the term weights (i.e., increasing or decreasing the weight of a term through the use of a multiplying factor), by adding new terms or by using the combination of these two approaches. New terms are generally selected based on the most important terms from a set of documents deemed relevant to the query. Key factors in the relevance feedback process are the determination of the relevant documents and the selection of new terms. There are two main types of relevance feedback: one which is dependent upon user input to determine relevant documents, and one that performs independently of any user intervention by assuming that the top-ranked documents are relevant to the query.

a. User-Dependent Methods

Much of the initial work in relevance feedback was conducted by Rocchio in the late-1960's (Rocchio71). Rocchio's early work was subsequently expanded upon by Ide and both Rocchio and Ide used the vector space model as the basis for their IR system (Ide71). Relevance feedback in Rocchio's original algorithm was accomplished by conducting an initial query which returned a small number of documents. A user evaluated each document and made a judgment as to whether or not the document was relevant. The vectors for all relevant documents were summed and normalized into a new vector by dividing by the number of relevant documents. A similar process was done for the non-relevant documents. The new vectors could be further modified by multiplying by a weight adjustment value. The original query vector was modified by adding the new

CIKM 97 Las Vegas Nevada USA

vector for the relevant documents and subtracting the new vector for the non-relevant documents (Rocchio71). This whole process could be repeated through multiple iterations until the user was satisfied with the results.

Ide expanded on Rocchio's work by eliminating the vector normalization and during a series of experiments on a small test collection, Ide discovered that the use of non-relevant documents for feedback seemed to raise the ranks of fairly high-ranking relevant documents and, at the same time, lower the ranks of some low-ranking relevant documents (Ide71).

Subsequent work verified Rocchio's and Ide's original results that relevance feedback can produce improvement in retrieval accuracy (Salton90, Harman92). Salton and Buckley reported in 1990 that the best overall relevance feedback method was the Ide dec-hi method, where terms are directly added to the queries and only one non-relevant item is used in the process (Salton90).

The use of relevance feedback has also been explored using IR systems which incorporate the probabilistic model rather than the vector space model. Much work in this area has been done by Harper, Harman, Croft, Sparck Jones, and van Rijsbergen (Harper78, Harman92, Croft79, Sparck Jones79, van Rijsbergen77, Salton90). Harman's experiments on relevance feedback using the probabilistic model showed that while it was possible to obtain improvements in the query response, the vector space model showed good feedback performance on most collections whereas the probabilistic model had problems with some collections (Harman92).

So while relevance feedback in both the vector space and probabilistic models has been shown to improve query responses, the major drawback to all of these algorithms is their dependence upon user relevance judgments during their intermediate stages. In many of today's IR systems, it is impractical or not feasible to ask users to review and evaluate intermediate query results. The most promising approach to solving this difficulty is through the use of automatic relevance feedback.

b. User-Independent Methods

Many of the researchers in relevance feedback describe their processes as "automatic" or "pseudo" relevance feedback. By this, most are referring to the process of automatically reformulating the original query, but they are still dependent upon user relevance judgments. In true automatic relevance feedback, a pre-determined number of documents retrieved by the original query are assumed to be relevant. Terms from these relevant documents are used to modify the original query using a relevance feedback algorithm. The formulae used for the automatic relevance feedback are variations of the Rocchio/Ide formulae, but with the concept of non-relevant documents ignored and eliminated from the computations.

Salton and Buckley conducted some of the first experiments in automatic relevance feedback in 1990 using several small document collections (the number of documents ranged from 1033 - 12,684). In these experiments they set the weight adjustment multiplier in the Rocchio relevance feedback formula equal to zero for the non-relevant documents. Even though the improvement in the query response was less than other experiments using the non-relevant documents, there still was significant improvement over query responses without relevance feedback (Salton90).

A number of participants in the TREC conferences have used relevance feedback and some of the best results in automatic relevance feedback were obtained by Buckley, et al, in 1995 when they used the original Rocchio relevance feedback algorithm in the SMART system on the Tipster document collection. During the experiment, they expanded the queries and reweighted the original query terms by adding the most frequently occurring 50 single terms and 10 phrases from the top 20 documents. The component for non-relevant documents in Rocchio's formula was dropped. The results obtained from this method were above the average for most of the queries when compared to the other participants (Buckley95).

3. Relevance Feedback Improvements

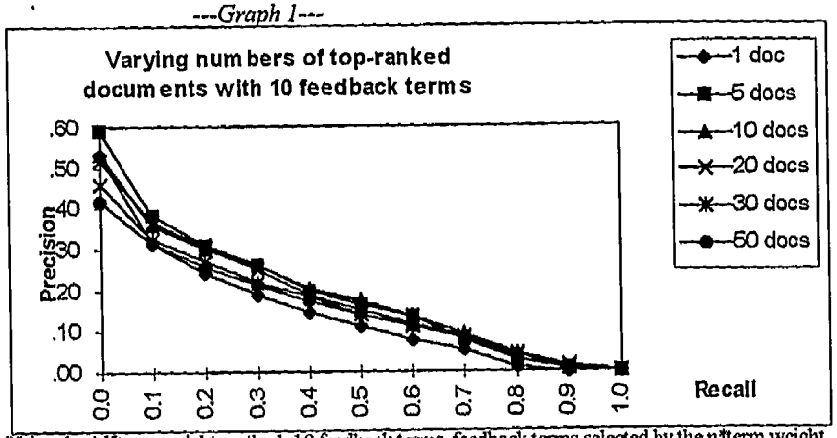
Our IR system is based on the vector space model and implemented in a relational database using unchanged SQL. Details of implementing an IR system using unchanged SQL are found in (Grossman96, Grossman97). A four processor Teradata DBC/1012 parallel processing computer is used as the platform for the IR system. The Teradata DBC/1012 Database Computer is a special purpose machine designed to run a relational database management system using standard SQL. As part of the tuning of our IR system, we have conducted a series of calibrations using various portions of the TREC queries and the Tipster data collection. The following sections describe our calibration experiments and the improvements in retrieval effectiveness demonstrated by the different relevance feedback techniques.

a. Number of Top-Ranked Documents

One of the issues in relevance feedback concerns the optimal number of top-ranked documents to use as the source of the feedback terms. To investigate the impact on precision and recall from the number of top-ranked documents used, we conducted experiments using the short versions of the 50 TREC-4 queries and disk 2 of the Tipster collection. In these experiments, the *idf* term weight method (described in section 3e) was used and either 10 or 20 feedback terms were selected from 1, 5, 10, 20, 30, and 50 top-ranked documents. Similar experiments were done by Harman in 1992 but with the Cranfield document collection and using the probabilistic model (Harman92). To identify the relationship between the exact precision and the number of top-ranked documents used for relevance feedback, we calculated the correlation coefficient. The correlation coefficient is a measure of the strength of the linear relationship between the exact precision averaged over the 50 queries and the number of top-ranked documents used for relevance feedback. We determined that the correlation coefficient is equal to -0.3915. The negative correlation coefficient implies that as the number of top-ranked documents used for relevance feedback increases, the exact precision averaged over the 50 queries will tend to decrease. Graphs 1 and 2 illustrate this relationship by showing how the levels of precision and recall are impacted when the number of top-ranked documents used for relevance feedback is varied.

The results shown in graphs 1 and 2 indicate that the greatest increases in precision and recall are obtained when between 5 and 20 top-ranked documents are used for use in relevance feedback. The results also indicate that selecting a smaller number of documents or a larger number of documents for use in relevance feedback produces less than optimal levels of precision. The results also show that there was a negligible difference when using 10 or 20 feedback terms. The following

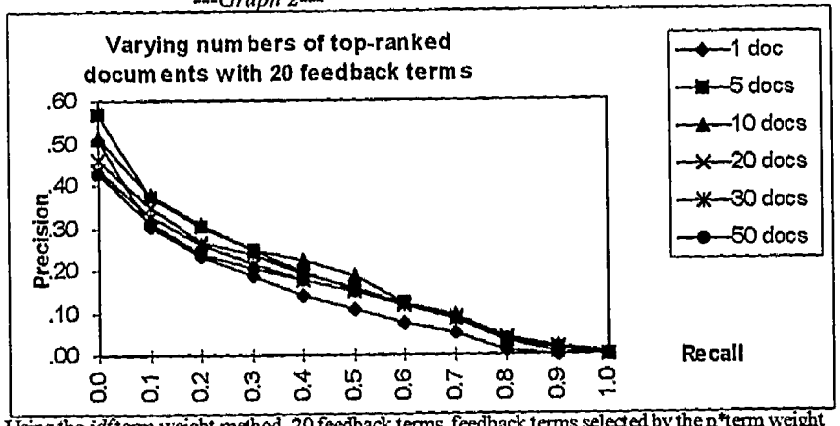
---Graph 1---



Using the *idf* term weight method, 10 feedback terms, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Number of top-ranked documents	Average Precision	Percent Change	Exact Precision	Percent Change
1	.1339	—	.1764	—
5	.1769	+32%	.2162	+18%
10	.1755	+31%	.2220	+26%
20	.1719	+28%	.2000	+13%
30	.1520	+14%	.1835	+4%
50	.1479	+10%	.1810	+3%

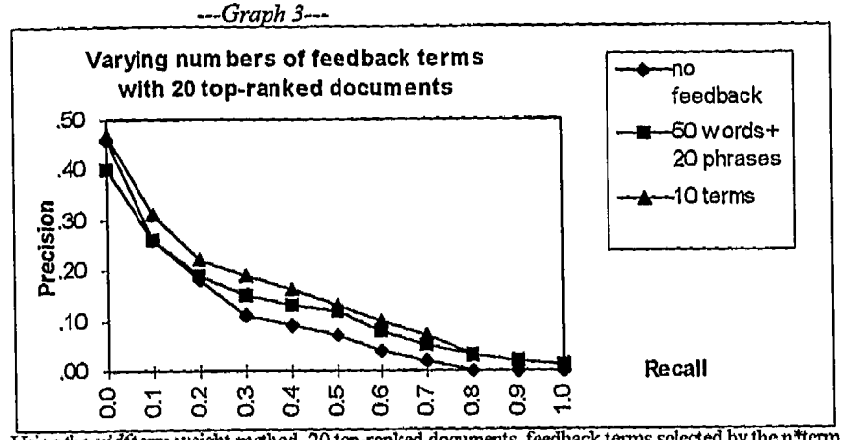
---Graph 2---



Using the *idf* term weight method, 20 feedback terms, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Number of top-ranked documents	Average Precision	Percent Change	Exact Precision	Percent Change
1	.1283	—	.1722	—
5	.1721	+34%	.2157	+25%
10	.1802	+40%	.2206	+28%
20	.1599	+25%	.1978	+15%
30	.1517	+18%	.1822	+5%
50	.1441	+12%	.1772	0%

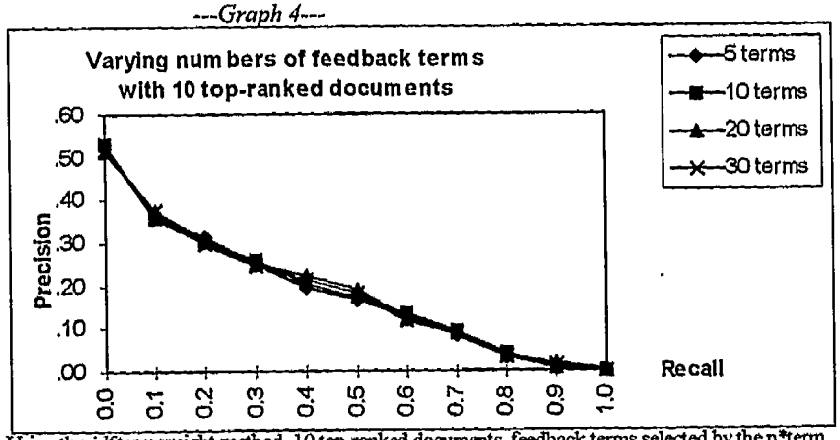
---Graph 3---



Using the *nidf* term weight method, 20 top-ranked documents, feedback terms selected by the n*term weight method, and subset of documents from Tipster disks 2 and 4 with the TREC-5 queries.

Number of feedback terms used	Average Precision	Percent Change	Exact Precision	Percent Change
0 (feedback not done)	.0914	—	.1306	—
50 words + 20 phrases	.1147	+25%	.1486	+14%
10 terms (either words or phrases	.1400	+53%	.1755	+34%

---Graph 4---



Using the *idf* term weight method, 10 top-ranked documents, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Number of feedback terms used	Average Precision	Percent Change	Exact Precision	Percent Change
5	.1713	—	.2153	—
10	.1755	+2%	.2220	+3%
20	.1802	+5%	.2206	+2%
30	.1746	+2%	.2109	-2%

section provides more detail on the impact on relevance feedback when varying numbers of feedback terms are selected.

b. Number of Feedback Terms

To evaluate the impact of the number of terms selected on improving precision and recall, we conducted two sets of experiments. The first set of experiments was conducted using the short versions of the 50 TREC-5 queries and a subset of the documents from disks 2 and 4 of the Tipster data collection. In the first set of experiments, we identified the 20 top-ranked documents for each query and used the *n * nidf* feedback term selection technique to select 10 terms in one experiment and 50 words+20 phrases for the second experiment. Results shown in graph 3 compare the precision and recall levels when the two different numbers of feedback terms are used and shows how much improvement over the baseline is obtained by both methods. Graph 3 illustrates that using a smaller number of feedback terms is clearly better for the short TREC-5 queries than using a larger number of feedback terms although many groups in TREC have used 50 words+20 phrases for relevance feedback (Buckley95, Ballerini96).

The second set of experiments was conducted using only disk 2 of the Tipster collection, the short versions of the 50 TREC-4 queries, and the *idf* term weighting method. During these experiments we varied the number of feedback terms used when 10, 20, and 30 top-ranked documents were selected for relevance feedback. Graphs 4, 5, and 6 show how the precision and recall levels varied when 5, 10, 20, and 30 feedback terms were used. These graphs show that the highest levels of precision and recall are obtained when between 10 and 20 top-ranked documents are selected for relevance feedback and when between 10 and 20 new terms are added to the short versions of the TREC-5 and TREC-4 queries with terms consisting of either words or phrases.

c. Feedback Term Selection Techniques

The first step of the automatic relevance feedback process identifies the *n* top-ranked documents. These documents are assumed to be relevant and are used as the source of the feedback terms. The issue then becomes one of identifying the particular terms, and only those terms, which will improve precision and recall results for the query. If "good" terms are chosen, the query will find more relevant documents. However, if "bad" terms are chosen, the feedback terms can potentially re-focus the query onto a topic different from the original query topic and fewer relevant documents will be identified after relevance feedback. In 1992, Harman experimented with several different feedback term selection techniques using the Cranfield document collection (Harman92). According to her results, the best feedback term selection techniques were those which incorporated information about the total frequency of a term in the collection rather than just the frequency of the term within a document.

To determine the impact on precision and recall from the type of feedback term selection method used, we developed and tested several different feedback term selection methods. These experiments were conducted using various combinations of documents from disks 2 and 4 of the Tipster data collection along with the short versions of queries from TREC-4 and TREC-5. The feedback term selection methods are described below:

1) ***n * nidf*** (where *n* = number of top-ranked documents containing the term and *nidf* = the weight of the term in the collection.)
2) **TermWeight** (where termwt = (*nidf* * ((1 + log(termcnt))/logavgtf)))
3) ***n * idf*** (where *n* = number of top-ranked documents containing the term and *idf* = the weight of the term in the collection.)
4) **Min3Docs** (where the top terms occurring in at least 3 of the top-ranked documents are ordered by their *idf* values)
5) **Min4Docs** (where the top terms occurring in at least 4 of the top-ranked documents are ordered by their *idf* values)
6) **SUM(termcnt * *idf*)/reldoc** (where reldoc = number of top-ranked documents chosen for relevance feedback)
7) *idf* * log(*n*) (where *idf* = the weight of the term in the collection and *n* = number of top-ranked documents containing the term)

The most successful term selection technique we have identified so far is the (*n * term weight*) sort method regardless of whether *idf* or *nidf* term weights are used.

d. Feedback Term Scaling

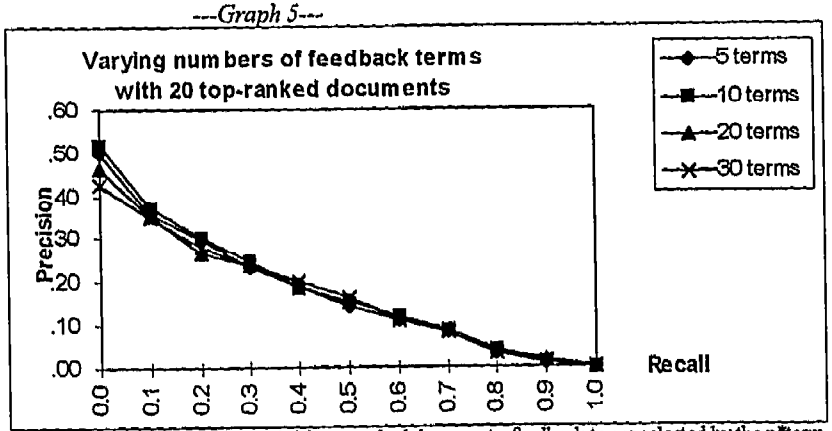
Rocchio's original formula contained a scaling factor which allowed the relative weights of feedback terms to be either increased or decreased with respect to the original query terms. To investigate the impact of scaling the weights of the feedback terms, we conducted a series of experiments using documents from disk 2 of the Tipster collection along with the short versions of the 50 TREC-4 queries. Using 10 or 20 feedback terms chosen from the 20 top-ranked documents, we adjusted the scaling factor of the feedback terms relative to the original query terms by 0.2, 0.4, 0.6, 0.8, and 1.0. Graphs 7 and 8 show the impact scaling has on the overall precision and recall for the queries and show that a scaling factor between 0.4 and 0.6 produces the greatest improvement in precision and recall.

e. Term Weighting

In the vector space model, each component of the vector represents a term in the document. In a binary term weighting scheme, a "1" indicates the presence of a particular term in the document while a "0" indicates its absence. The major drawback to this scheme is that all terms have the same weight, i.e., a "1", and no distinction is made between terms of high importance such as names or places and terms of lower importance such as adjectives. One of the ways to reflect the relative importance of terms within documents is to assign a specific weight to each term in the document. In the vector space model, this is accomplished by having each component in the document vector represent the weight of the term in the document.

One of the more commonly used methods of term weighting is the "*idf*" (inverse document frequency) method (Salton89). The *idf* term weights, however, do not take into consideration the overall length of the document. This means that shorter documents with fewer terms will be at a disadvantage when compared to the longer documents with more terms. For example, if one document contained 10 terms and another contained 100 terms and both documents had two terms that matched the query, the *idf* term weights would weight the two terms equally in both documents even though the two terms are of more relative importance in the document with 10 terms than the document with 100 terms.

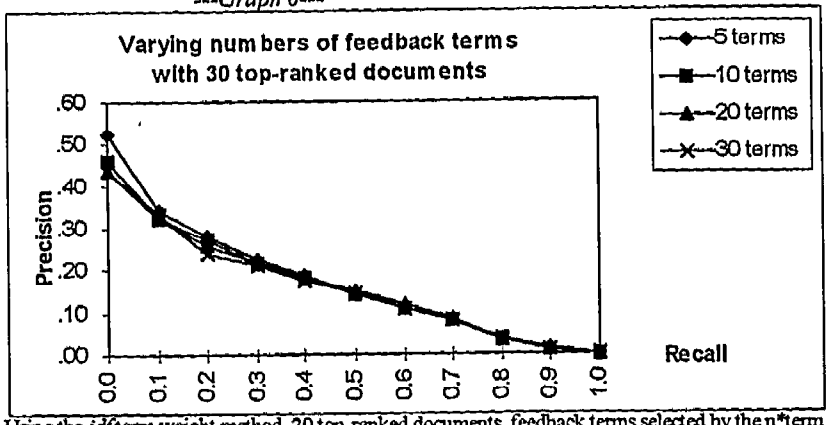
---Graph 5---



Using the *idf* term weight method, 20 top-ranked documents, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Number of feedback terms used	Average Precision	Percent Change	Exact Precision	Percent Change
5	.1626	---	.2097	---
10	.1719	+6%	.2000	-5%
20	.1599	-2%	.1978	-6%
30	.1623	+0%	.2053	-2%

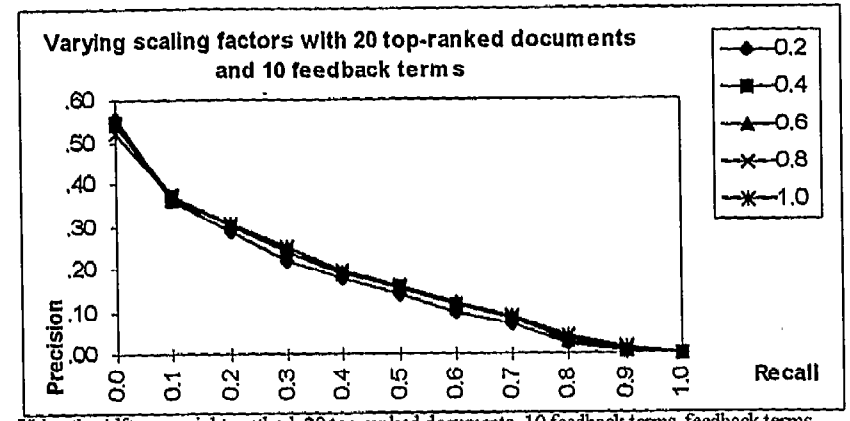
---Graph 6---



Using the *idf* term weight method, 20 top-ranked documents, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Number of feedback terms used	Average Precision	Percent Change	Exact Precision	Percent Change
5	.1592	---	.2000	---
10	.1520	-5%	.1835	-8%
20	.1517	-5%	.1822	-9%
30	.1473	-7%	.1804	-10%

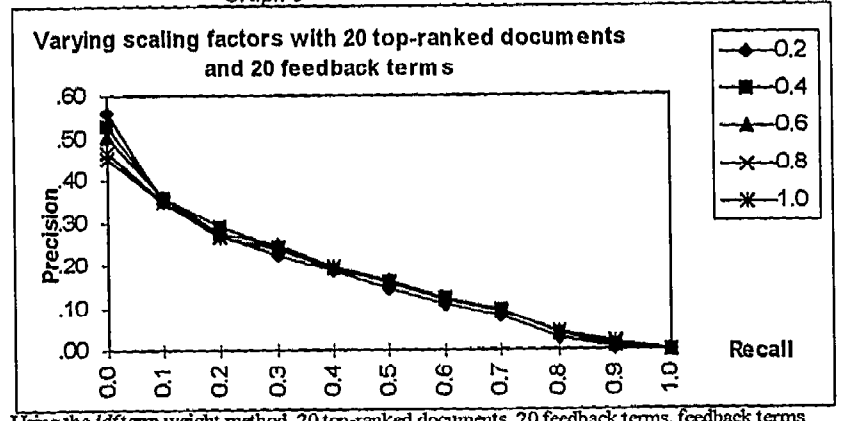
---Graph 7---



Using the *idf* term weight method, 20 top-ranked documents, 10 feedback terms, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Feedback term scale	Average Precision	Percent Change	Exact Precision	Percent Change
0.2	.1580	---	.2023	---
0.4	.1696	+7%	.2074	+3%
0.6	.1741	+10%	.2075	+3%
0.8	.1741	+10%	.2055	+2%
1.0	.1719	+9%	.2000	-1%

---Graph 8---



Using the *idf* term weight method, 20 top-ranked documents, 20 feedback terms, feedback terms selected by the n*term weight method, and documents from Tipster disk 2 with the TREC-4 queries.

Feedback term scale	Average Precision	Percent Change	Exact Precision	Percent Change
0.2	.1584	---	.2079	---
0.4	.1671	+5%	.2093	+1%
0.6	.1660	+5%	.2082	+1%
0.8	.1625	+3%	.2008	-3%
1.0	.1599	+1%	.1978	-5%

To incorporate this information and normalize for document lengths, Buckley, et al, developed the "lnu" term weight equations described in (Buckley95, Singhal96). To reflect the weight of a term within a query, we adopted the approach proposed in (Knaus95, Ballerini96) which extends the lnu term weight method to create the *"nidf"* term weighting method.

The experiments described in graphs 9 and 10 were conducted using the short versions of the 50 TREC-5 queries and a subset of the documents from disks 2 and 4 of the Tipster data collection. Graph 9 shows the precision and recall values resulting from the *idf* and *nidf* term weights for the initial queries and before relevance feedback. Graph 10 shows the precision and recall values for the different term weights after relevance feedback has been done on the queries. It is interesting to note that while the *nidf* term weighting method performed slightly worse than the *idf* term weighting method before relevance feedback, the *nidf* term weighting method showed substantial improvement over the *idf* term weighting method after relevance feedback .

Graphs 9 and 10 demonstrate that when the *nidf* term weighting method is used for relevance feedback, the results show a significant increase in precision and recall over results obtained using the *idf* term weighting method.

f. Document Clustering

When automatic relevance feedback is used, the top-ranked documents are assumed to be relevant to the query. If all of the documents are not actually relevant to the query, the overall precision and recall results could be significantly degraded by performing relevance feedback. To distinguish the actual relevant documents from the non-relevant documents, Lu has proposed that the top-ranked documents should be clustered under the assumption that if most of the top-ranked documents for a query are actually relevant to that query, they will form a cluster which excludes the non-relevant documents. The documents within the cluster will then have a higher similarity coefficient to other documents in the cluster than to the non-relevant documents outside of the cluster and documents having a low similarity coefficient with the other documents can be removed from the feedback process (Lu96). Lu, et al, demonstrated that document clustering on the top-ranked documents in relevance feedback improves the retrieval effectiveness for some queries.

We used the work done by Lu as a model and developed three simple document clustering algorithms.

1) **Highest pair** - This algorithm begins by identifying the pair of documents for each query that has the highest similarity coefficient. All documents related to these two documents with a similarity coefficient above the stated threshold are then grouped with the original document pair. This process is repeated once more and all documents related to documents in the group with a similarity coefficient above the stated threshold are also added to the group. Any documents remaining outside of the group are then excluded from the relevance feedback process.

2) **Largest group** - This algorithm begins by identifying the single document from among the top-ranked documents which is related to the highest number of the other top-ranked documents by a similarity coefficient above the stated threshold. The document and its related documents become grouped and any documents remaining outside of the group are excluded from the relevance feedback process. If a query has no document pairs with a similarity coefficient above the stated threshold, then the query will not undergo any relevance feedback.

3) **Largest group, expanded** - This algorithm is similar to the largest group algorithm except that documents not already in the group but which have a similarity coefficient to any group member document above the stated threshold are also included in the group. Essentially, a second pass is made against a similarity matrix table.

Graph 11 illustrates the impact on precision and recall when document clustering is used on the 20 top-ranked documents selected for relevance feedback. These experiments were conducted using the short version of the 50 TREC-5 queries, a subset of documents from disks 2 and 4 of the Tipster data collection, and the *nidf* term weight method.

When document clustering is used on the top-ranked documents, some queries show improvements in their retrieval results, however, the results for other queries decline. These results indicate that while clustering of the top-ranked documents will improve the relevance feedback results for some queries, it also worsens the relevance feedback results for other queries.

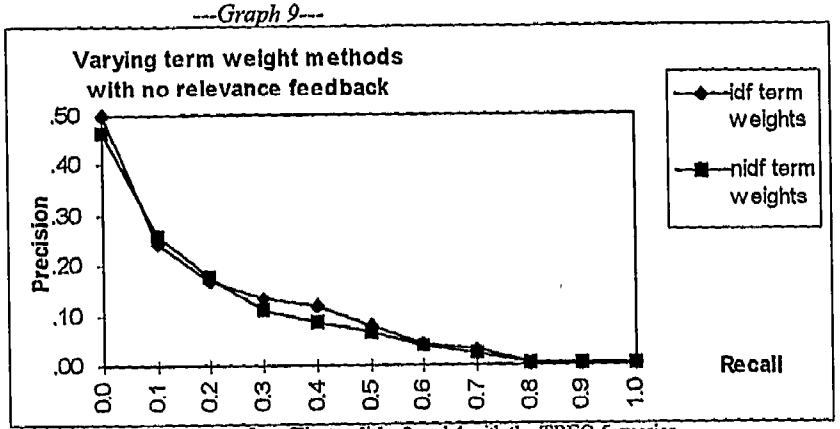
g. Relevance Feedback Thresholding

When we examined the precision and recall levels for individual queries, we identified a correlation coefficient of +0.24 between the percentage of improvement in exact precision seen during relevance feedback and the average of the *nidf* term weights of the words (not including any phrases) within the queries. This correlation implies that queries with an average *nidf* of their words below a certain threshold should not undergo relevance feedback. To test this hypothesis, we conducted several experiments using the short versions of the TREC-5 queries, a subset of disks 2 and 4 of the Tipster data collection, the *nidf* term weight method, the *n* * *nidf* feedback term selection method, 10 feedback terms, and 20 top-ranked documents. Our experiments determined that the maximum improvement to precision and recall is realized if we do not perform relevance feedback on queries having an average *nidf* < .2175. Further experiments showed that the average *nidf* for a query should be calculated on only the words having an *nidf* <= 0.4. (The 0.4 value roughly corresponds to the word occurring in less than 35,000 documents in the collection. Using these calculations, six of the fifty TREC-5 queries (#251, #263, #268, #270, #272, #293) did not undergo relevance feedback. As a result of relevance feedback thresholding, the average precision increased +1.4% from .1400 to .1421 and the exact precision increased +2.1% from .1755 to .1791. These results demonstrate that improvement, while not dramatic, can be achieved through relevance feedback thresholding. The relationship between the weight of the words in the query and the improvement from relevance feedback needs to be explored further.

h. Combination of Techniques

Graph 12 illustrates the improvement in precision and recall which is achieved when a combination of the techniques described above are used. The following experiments were conducted using the full set of documents from disks 2 and 4 of the Tipster collection along with the short versions of the 50 queries from TREC-5. The *nidf* term weighting method was used and 20 of the top-ranked documents were selected for relevance feedback. The *n* * *nidf* term selection method was used to select 10 feedback terms (either words or phrases) for

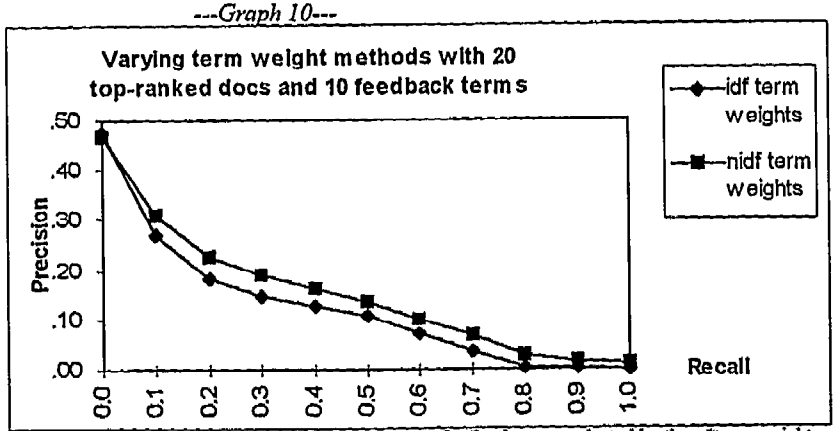
---Graph 9---



Using a subset of documents from Tipster disks 2 and 4 with the TREC-5 queries.

Term Weighting Method	Average Precision	Percent Change	Exact Precision	Percent Change
idf	.0966	---	.1410	---
nidf	.0914	-5%	.1306	-7%

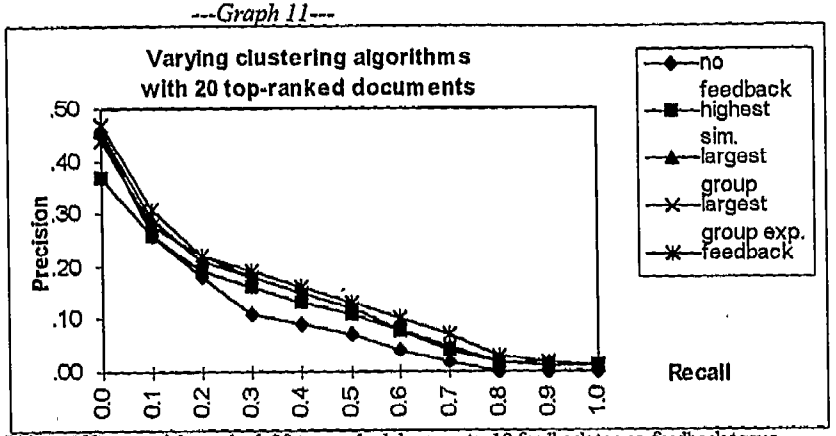
---Graph 10---



Using 20 top-ranked documents, 10 feedback terms, feedback terms selected by the n*term weight method, and a subset of documents from Tipster disks 2 and 4 with the TREC-5 queries.

Term Weighting Method	Average Precision	Percent Change	Exact Precision	Percent Change
idf	.1100	---	.1421	---
nidf	.1400	-27%	.1755	+24%

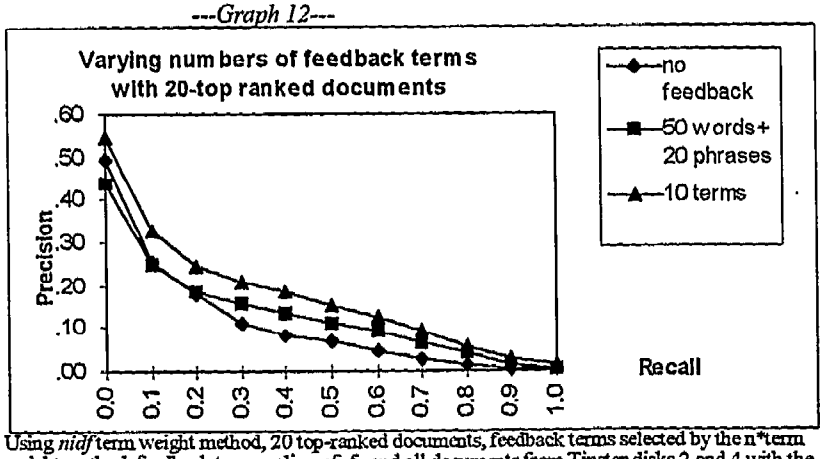
---Graph 11---



Using *nidf* term weight method, 20 top-ranked documents, 10 feedback terms, feedback terms selected by the n*term weight method, and a subset of documents from Tipster disks 2 and 4 with the TREC-5 queries.

Clustering Method	Average Precision	Percent Change	Exact Precision	Percent Change
No feedback or clustering	.0914	---	.1306	---
Highest Similarity	.1081	+18%	.1435	+10%
Largest Group	.1242	+36%	.1573	+20%
Largest Group Expanded	.1233	+35%	.1573	+20%
Feedback, no clustering	.1400	+53%	.1755	+34%

---Graph 12---



Using *nidf* term weight method, 20 top-ranked documents, feedback terms selected by the n*term weight method, feedback term scaling of .5, and all documents from Tipster disks 2 and 4 with the TREC-5 queries.

Number of feedback terms used	Average Precision	Percent Change	Exact Precision	Percent Change
0 (feedback not done)	.0928	---	.1346	---
50 words+20 phrases	.1199	+29%	.1499	+11%
10 terms (either words or phrases	.1570	+69%	.1859	+38%

one trial and 50 feedback words plus 20 feedback phrases for the second trial. All feedback terms were scaled by a factor of .5.

The results clearly demonstrate that a significant amount of improvement in precision and recall can be obtained when a combination of relevance feedback improvement techniques are used. They also show that using only 10 terms (either words or phrases) for relevance feedback produces a 31% improvement over the commonly used method of using 50 words and 20 phrases for relevance feedback.

4. Conclusions and Future Work

The experiments demonstrated that on the short versions of the TREC-5 and TREC-4 queries, performing relevance feedback by: 1) expanding the query with 10 to 20 new terms (either words or phrases) performed better than the commonly used process of expanding the query with 50 new words and 20 new phrases; and 2) using 5 to 20 top-ranked documents produced better results than more or less documents; and 3) using a scaling factor between 0.4 and 0.6 on the feedback terms appears to optimize the precision and recall levels for the queries. In addition, a direct comparison of two term weighting methods, *idf* and *nidf*, showed that while the *nidf* weights performed significantly better than the *idf* weights when relevance feedback was used, the *idf* weights performed better than the *nidf* weights when no relevance feedback was used. Finally, it appears that while clustering the top-ranked documents prior to selecting the feedback terms will improve the results for some queries, clustering will worsen the results for other queries.

One of the interesting results from these experiments is it appears that certain characteristics of a query (i.e., average *nidf* of words) are a predictor as to how likely the query is to improve under relevance feedback. Further work needs to be done to determine if other characteristics of the document collection or query will predict retrieval effectiveness. In addition, as the term weighting and relevance feedback term selection methods have been shown to have a significant impact on the retrieval effectiveness, additional enhancements to both term weighting and feedback term selection methods need to be explored. Finally, document clustering on the documents used for relevance feedback has been shown to improve the results for some queries while worsening the results for other queries. The relationship between document clustering and relevance feedback also needs further investigation.

Acknowledgments

This work was supported in part by matching funds from the National Science Foundation under the National Young Investigator Program under contract number IRI-9357785. Ophir Frieder is currently on leave from the Department of Computer Science at George Mason University.

References

(Ballerini96) Ballerini, J., M. Buchel, D. Knaus, B. Mateev, E. Mittendorf, P. Schauble, P. Sheridan and M. Wechsler, "SPIDER Retrieval System at TREC-5," to appear in Text Retrieval Conference, sponsored by National Institute of Standards and Technology and Advanced Research Projects Agency, November 1996.

(Buckley95) Buckley, Chris, Amit Singhal, Mandar Mitra and Gerard Salton, "New Retrieval Approaches Using SMART: TREC 4," Text Retrieval Conference, sponsored by National Institute of Standards and Technology and Advanced Research Projects Agency, November 1995.

(Croft79) Croft, W. B. and D. J. Harper, "Using probabilistic models of document retrieval without relevance information," *Journal of Documentation*, v. 35, pp. 285-195, 1979.

(Grossman96) Grossman, D, C. Lundquist, J. Reichert, D. Holmes and O. Frieder, "Using Relevance Feedback within the Relational Model for TREC-5," to appear in Text Retrieval Conference, sponsored by National Institute of Standards and Technology and Advanced Research Projects Agency, November 1996.

(Grossman97) Grossman, D., O. Frieder, D. Holmes, and D. Roberts, "Integrating Structured Data and Text: A Relational Approach," *Journal of the American Society of Information Science*, January 1997.

(Harman92) Harman, D., "Relevance Feedback Revisited," *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Ed. Nicholas Belkin, Peter Ingwersen and Annelise Mark Pejtersen, SIGIR Forum, June 21-24, 1992.

(Harman96) Harman, D., "Overview of the Fourth Text Retrieval Conference (TREC-4)," to appear in Text Retrieval Conference, sponsored by National Institute of Standards and Technology and Advanced Research Projects Agency, November 1996.

(Harper78) Harper, D. J. and C. J. van Rijsbergen, "An Evaluation of Feedback in Document Retrieval using Co-Occurrence Data," *Journal of Documentation*, v. 34, pp. 189-216, 1978.

(Ide71) Ide, E., "New Experiments in Relevance Feedback," Gerard Salton, Editor, The SMART Retrieval System, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.

(Knaus95) Knaus, D., E. Mittendorf, P. Schauble, P. Sheridan, "Highlighting Relevant Passages for Users of the Interactive SPIDER Retrieval System," Text Retrieval Conference, sponsored by National Institute of Standards and Technology and Advanced Research Projects Agency, November 1995.

(Lu96) Lu, A., M. Ayoub, and J. Dong, "Ad Hoc Experiments Using EUREKA," to appear in Text Retrieval Conference, sponsored by National Institute of Standards and Technology and Advanced Research Projects Agency, November 1996.

(Lundquist97) Lundquist, C., D. Grossman, O. Frieder, and D. Holmes, "A Parallel Implementation of Relevance Feedback using the Relational Model," *Proceedings of the World Multiconference on Systemics, Cybernetics, and Informatics*, July 1997.

(Rocchio71) Rocchio, Jr., J. J., "Relevance Feedback in Information Retrieval," Gerard Salton, Editor, The SMART Retrieval System, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.

(Salton89) Salton, G., Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer, Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1989.

(Salton90) Salton, G. and C. Buckley, "Improving Retrieval Performance by Relevance Feedback," *Journal of the American Society for Information Science*, v. 41, no. 4, pp. 288-297, 1990.

(Singhal96) Singhal, A., C. Buckley, and M. Mitra, "Pivoted Document Length Normalization," *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Ed. Hans-Peter Frei, Donna Harman, Peter Schauble and Ross Wilkinson, SIGIR Forum, August 18-22, 1996.

(Sparck Jones79) Sparck Jones, K. "Search Term Relevance Weighting given little Relevance Information," *Journal of Documentation*, v. 35, pp. 30-48, 1979.

(van Rijsbergen77) van Rijsbergen, C. J., "A Theoretical Basis for the use of Co-Occurrence Data in Information Retrieval," *Journal of Documentation*, v. 33, pp. 106-119, 1977.