XI. Document Indexing Based on Relevance Feedback

T. L. Brauen, R. C. Holt and T. R. Wilcox

Abstract

A relatively simple method is proposed in this study for using accumulated relevance feedback information in an automatic information retrieval system to improve precision and recall. This method involves the dynamic modification of relevant document vectors after each retrieval run. Experiments on a collection of 425 documents indicate that this method does improve precision and recall significantly. More experiments must be conducted to perfect the method for large scale practical applications.

1. Introduction

User feedback techniques have been developed in an attempt to achieve higher precision and recall in automatic retrieval systems such as SMART. The user gives the system an indication of the relevance to his query of certain documents. This information about what the query should retrieve is used by the system to modify query and document vectors so that the desired documents are retrieved. It is hoped that other documents in the same area of the document space will also be relevant to the query.

Current proposals use this relevance information only in the processing of one query. There is, however, reason to believe that this information will also be helpful in the processing of similar queries from other users. It seems quite reasonable to assume that a number of knowledgeable users submitting similar queries may select roughly the same set of documents as relevant to their query. If this is indeed the case, it will be of advantage to the system to save relevance information for each of the processed queries. This information can be used as a first approximation to future users' responses in the feedback process. As the relevance information for a given type of query builds up, this first approximation should become a very good one and thus reduce the number of feedback iterations necessary to satisfy the users' requests. The time saved can be used to continue searching for more relevant documents, thereby improving the recall.

Given that feedback information should be retained by the system, it is clear that normal query modification will not achieve this aim since queries are not a permanent part of the system. It is, therefore, natural to focus one's attention on the document vectors since they are a permanent part of any retrieval system.

Some work has been done in the area of document vector modification. Friedman, Maceyak and Weiss [1] warp the documents closer to the query and thus improve the chances of a better recall performance. This modification is, however, only temporary. Davis, Linsky and Zelkowitz [2] also modify the entire document space in an effort to improve recall. The modification is permanent. These methods are disadvantageous in that they are time consuming, the modifications being complicated, and requiring the manipulation of the entire document space. These two methods would probably not be feasible in a large retrieval system.

2. Method

The problem is to find a method which retains the feedback information, but which is practical in the sense that it uses limited amounts of time and space regardless of the size of the document collection. To achieve these ends the authors propose the following procedure. The original query vector q_o is repeatedly modified, using any of the existing relevance feedback techniques, until a query vector q_n is obtained which retrieves a set of documents D acceptable to the user. The terms and weights of the document vectors in D are now modified to decrease the angle between each document vector d_i in D and the original query vector q_o. Since the cosine correlation (formula (1))

$$\cos\,(q_o,d_i) = \sum_{j=1}^{n} q_o^j\, d_i^j \Big/ \sqrt{\sum_{j=1}^{n} (q_o^j)^2 \cdot \sum_{j=1}^{n} (d_i^j)^2} \qquad (1)$$

(where d_i^j denotes the j^{th} term weight of vector d_i)

is used to determine the correlation between the vectors q_o and d_i, a decrease in the angle between vectors increases their correlation.

For the original query vector q_o and its relevant documents d_i in D defined above, the document modification process takes place in two steps. For each relevant document vector, the query vector is first normalized to the "length" of the document vector by making the sum of weights of q_o equal to the corresponding sum for d_i. If A_q is the sum of the weights of the query vector, and if A_d is the sum for the document vector, then the normalized query vector is defined as:

$$\hat{q}_o = q_o (A_d/A_q). \tag{2}$$

The weights of the relevant document vector d_i are then modified according to the formula:

$$\hat{d}_i^j = d_i^j + \alpha(\hat{q}_o^j - d_i^j) \quad \text{where } 0<\alpha<1. \tag{3}$$

By normalizing the query vector as in (2) the above modification is based on vectors of equal length. Since (3) is a linear transformation, this means $\hat{d}_i$ will have the same length as d_i and prevents document vectors from shrinking.* With $0<\alpha<1$, the term $\alpha(\hat{q}_o^j - d_i^j)$ may be fractional. In this case, the fraction is rounded to the nearest integer.

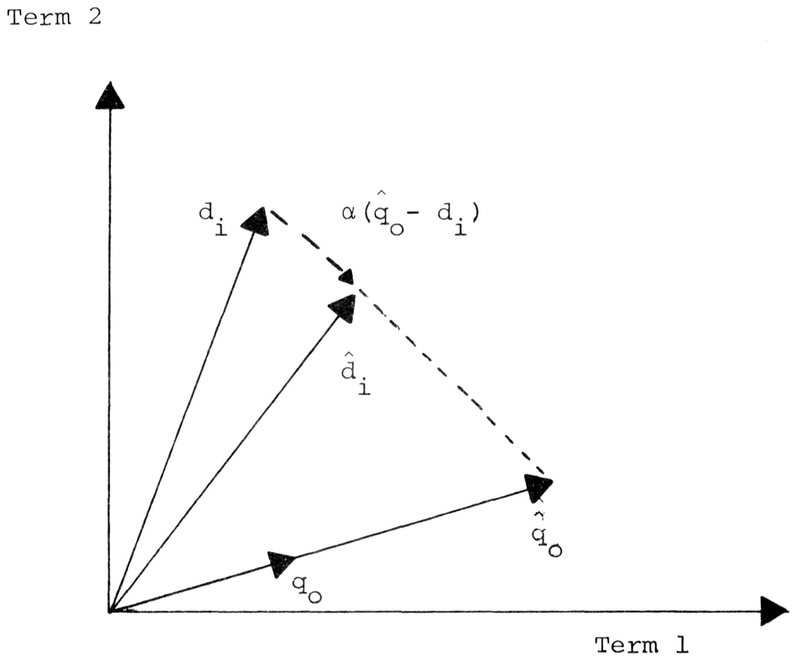
Of course, in actual systems, a document term whose weight d_i^j is zero is not stored in the document vector. Thus, if a term appears with a non-zero weight in q_o but is absent in d_i, the term is added to the document vector with a weight proportional to its corresponding weight in the query vector. Terms appearing in both the query and document vectors lead to modifications based directly on (3), while terms appearing in neither vector are not considered. Fig. 1 illustrates the modification process for a q_o and d_i in two-space.

*The length of any document vector d_i equals the sum of its term weights $\sum_j d_i^j$. From (3)

$$d_i^j = d_i^j + \alpha(\hat{q}_o^j - d_i^j)$$

$$\sum_j \hat{d}_i^j = \sum_j d_i^j + \alpha(\sum_j \hat{q}_o^j - \sum_j d_i^j)$$

$$\sum_j \hat{d}_i^j = \sum_j d_i^j \quad \text{since} \quad \sum_j \hat{q}_o^j = \sum_j d_i^j \quad \text{from (2).}$$



Document Modification Process

Fig. 1

Two features of the modification process should be noted. First, only the vectors of documents judged relevant to the query are modified. Second, if the same query is submitted by another user, the documents judged relevant by the first user will be retrieved more quickly because the modification has increased their correlation with the query vector. Ideally, most of these documents will also be relevant to the second user, and retrieval performance will therefore be improved.

Since queries are phrased in precise terms, similar queries may reasonably be expected to map to similar query vectors. If user B submits a query similar to, but not the same as, a query previously submitted by user A, the documents judged relevant by user A should be retrieved quickly for user B. This is true because the modification process increases the correlation between user A's query vector and his relevant document vectors, and because user B's query vector is similar to user A's query vector.

The possibility of improved retrieval performance for resubmitted identical queries under the modification process is obvious, but is of limited value. However, the fact that improved performance may also be expected for queries that are somewhat similar,but not identical, gives this method potential for practical application. This improvement for queries that are only similar is demonstrated in the experimental results given below.

The method takes little time to execute, and so is not costly to implement. Modification of vectors in large systems does, however, raise the possibility of necessary reclustering for search purposes. This is not as large a concern as may perhaps appear. In large systems, the percentage of document vectors modified for any query will be quite small, and if α is small, any changes made for a query will be slight. Thus, reclustering need not be undertaken often, and may well be necessary no more frequently than the reclustering required by the addition of new document vectors.

3. The Experiment

The method was tested using two document collections — the Cranfield 200 collection, consisting of 200 documents in aerodynamics, and 42 queries for which relevance judgments are available, and the Cranfield 400 collection, consisting of 425 documents and 155 queries with relevance judgments. The latter is the Cranfield 1400 collection compressed to include only those documents relevant to at least one query.

The following is a summary of the test procedure. The set of query vectors, along with a list of their relevant documents, and the set of document vectors are read into the system of testing programs. The query vectors are ordered in a random sequence and divided into two groups. The first group is used by the system to modify the document vectors while the second group is saved for later comparison.

For each query from the first group, the query vector and all its relevant document vectors are recovered from the query and document vector spaces. The a priori knowledge of relevant documents for the query simulates the results of relevance feedback iterations. Each relevant document vector is then modified using the normalized query vector according to formula (3). The modified document vectors are then returned to the document vector space.

When all relevant document vectors for all queries in group one have been modified, the second group of queries is used to test the effectiveness of the modification. Each query from the second group is submitted to a program which correlates it with vectors in the modified document space. The ten highest correlating documents are printed out along with the ranks of the relevant documents.

For each query submitted, global (normalized) precision and global (normalized) recall measures are computed. If there are n relevant documents in a space of N documents, and if r_i is the rank of the i^{th} relevant document, the normalized precision is given by

$$P_{norm} = 1- \left[\left(\sum_{i=1}^{n} \log r_i - \sum_{i=1}^{n} \log i \right) / \log \left(N!/(N-n)\,!n! \right) \right] \qquad (4)$$

and normalized recall is given by

$$R_{norm} = 1- (1/n) (\sum_{i=1}^{n} (r_i - i) / (N-n)). \qquad (5)$$

These normalized measures are used because they reflect the rankings of the relevant documents over the whole collection rather than rankings up to some arbitrary point. This eliminates the need for a subjective choice of cut-off point which may bias the results. When all queries from group two have been processed, an average normalized precision and recall are computed for all queries in the group.

The second group of queries is also correlated with document vectors in the original, unmodified document vector space, and the same normalized precision and recall measures are computed. The average global precision and the average global recall results for the two sets of correlations are compared using a t-test. From this, it is determined whether the differences in the measures observed over the two sets of runs are significant.

4. Experimental Results

A number of experiments were performed using the Cranfield 200 collection. In each run, the set of 42 queries was divided into a group of 37 queries, used to modify the document vectors, and a group of 5 queries used to test the effect of these modifications. Comparison of normalized recall and precision results for the 5 testing queries submitted to the modified and unmodified spaces shows no significant change in performance. These results are caused by the small size of the collection. With 37 queries modifying about 4 relevant documents each, each document is modified an average of 0.75 times. Thus, significant modification to the document vector space is not achieved, and results are much the same as those for the original document vector space. It also appears that few queries have overlapping relevant documents. As a result, modifications by the first 37 queries have only a small effect on the relevant documents of the 5 test queries.

A number of experiments performed using the Cranfield 400 collection produces more favorable results. In each run, the set of 155 queries was partitioned into a group of 124 queries, used to modify the document vectors and a group of 31 queries, used to test the effect of the modifications. Experiments were run using two randomly generated partitions, designated A and B and four values of α, ranging from 0.05 to 0.4. In one special experiment, designated III-A', modifications were made to a null space in which all document vectors were initialized to zero, instead of to the Cranfield document vectors. The results of these experiments are shown in Table 1.

Run No.	α	Ave Normed Precision			Ave Normed Recall		
		Unmod Space	Mod Space	% Change	Unmod Space	Mod Space	% Change
IA	.05	.6274	.6788	+8.2	.8891	.9100	+2.4
IIA	.10	.6274	.7073	+12.7	.8891	.9231	+3.8
IIB	.10	.5434	.5766	+6.1	.8439	.8624	+2.2
IIIA	.25	.6294	.7182	+8.1	.8891	.9309	+4.7
~~*IIIA'~~	~~.25~~	~~.6274~~	~~.5522~~	~~-8.4~~	~~.8891~~	~~.8784~~	~~-1.1~~
IIIB	.25	.5434	.5960	+9.7	.8439	.8671	+2.8
IVB	.40	.5434	.5963	+9.8	.8439	.8595	+1.8

Normalized Precision and Recall Results

Table 1

[*] For run IIIA' modifications were made to the null space.

For the first part of this discussion of results, special experiment IIIA' is ignored. In all runs, the average normalized recall and precision values for the modified document space are considerably higher for the modified document space than the corresponding values for the unmodified space. A t-test applied to the results presented in Table 1 indicates a significance level of 0.01 or better in all cases. Standard deviations over the global precision and recall values for the 31 testing queries were calculated for each run. The standard deviations after the modification were essentially the same as before. This indicates that the modifications provided a uniform improvement across the document space.

As can be seen from Table 1, varying α from 0.05 to 0.4 has no significant effect on the amount of improvement. It appears that an increase of α from 0.05 to 0.4 produces two conflicting effects on the search results. For small values of α only, small changes are made in relevant document vectors and therefore the modifications have only a small effect on the search results. The fact that significant improvement is nevertheless achieved indicates that the modifications are mostly beneficial to the test queries. In the case of large α rather drastic modifications are made to the document vectors, and therefore the modifications have a large effect on the retrieval performance. At the same time, the drastic extent of the modifications causes a more sporadic and less consistent improvement. That is, beneficial modifications become more beneficial to the document space while any non-beneficial modifications become more harmful. The net result is that variations in α in the experiments had little effect.

The normalized recall values for the unmodified document space, as given in the table, are close to 0.9. Since the maximum possible global recall is 1.0, a large improvement in the measure cannot be expected. In contrast, the global precision for the unmodified space is about 0.6, leaving room for considerable improvement. In the experiments, normalized recall is improved by three per cent on the average while normalized precision is improved by nine per cent on the average. It would appear that this difference between 3 and 9 per cent reflects characteristics of the measures and the collection rather than a characteristic of the method.

The special experiment IIIA' is different from the rest in that the document vectors in the modified space are derived entirely independently of the Cranfield document vectors. Experiment IIIA' is identical with experiment IIIA, except that a dummy document space is substituted for the Cranfield document space. The effect of zero document vectors is simulated in the dummy space by initializing the document vectors to a single large term weight which is not used in the Cranfield collection. This simulation is necessary so that during normalization, the query vectors would not be set to zero. Using the normalized recall measure in experiment IIIA', the performance of the document vectors derived solely from the 124 modifying queries is seen to be surprisingly good — only 8.4 per cent below the performance of the unmodified Cranfield document vectors. One is tempted to hypothesize that, given a few hundred more queries, the performance of the original Cranfield vectors would be surpassed. Experiments with larger collections are necessary to test this hypothesis.

The most important overall result of these experiments is the fact that in all cases the modified Cranfield document vectors performed significantly better than the unmodified vectors. From Table 1, improvements in normalized precision range from 6.1 to 12.7 per cent, while improvements in normalized recall range from 1.8 to 4.7 per cent. This seems to indicate that in an actual automatic retrieval system, where a very large number of queries are available for modifications and α can be made small, this method could be of practical value. These experiments tend to support the assumption that over a period of time similar query vectors can be expected to be received requesting similar sets of documents.

5. Discussion

Several interesting aspects of this method should be mentioned. As non-zero term weights are added to a document vector, the space required to represent the vector increases. In the experiments described here, the modifications added enough non-zero terms so that the storage required for the document space increased by over 25 per cent. In an operating information retrieval system, a routine could be used to compress the modified document space by setting small terms to zero.

With continual modification of the document vectors, the original indexing information contained in the document vectors is lost. Equation (3) can be rewritten as follows:

$$\hat{d}_i^j = (1-\alpha)\ d_i^j + \alpha \hat{q}_o^j \ . \qquad (6)$$

From equation (6) it is obvious that after one modification, a document term $\hat{d}_i^j$ contains the fraction $(1-\alpha)$ of the original document term information (d_i^j) and the fraction α of query term information $(\hat{q}_o^j)$. Regardless of the values of the original document vector and the relevant queries, the fraction of original indexing information remaining in a document vector after n modifications is $(1-\alpha)^n$. For example, in experiment IVB, where $\alpha = 0.4$, the fraction of original indexing information remaining in a document vector after 2 modifications to this document vector is $(1-0.4)^2 = 0.36$. In an operating information retrieval system, α should be much smaller than 0.4, possibly on the order of 0.01, so that a few "noisy" queries cannot seriously damage the document space, i.e., so that the indexing reflects the judgment of a large number of users.

In the Cranfield 400 collection, each query is associated with about 8 relevant documents. Since 124 queries were used for modification, about 992 documents were modified. On the average, then, each document was modified 2 to 3 times. Thus, for $\alpha = 0.4$, the preceding argument indicates that less than about 0.36 of the original indexing information remains after the modifications. Thus, in experiment IVB, normalized precision and recall were improved by replacing most of the original indexing information in each document vector by information derived from 2 or 3 queries.

It may seem disturbing that the information in the original document vectors, which may have been constructed at considerable expense, may eventually be lost to the system. If desired, a simple modification to the proposed method, such as flagging the original document term weights, could insure that these terms would remain intact. However, the advantage of saving terms which are seldom if every used is not obvious.

The results presented above suggest the following areas for research and application. Information retrieval centers might collect queries with corresponding relevant documents. This information could form the basis for future indexing schemes.

The proposed method might be used in a bootstrapping operation. A minimal and inexpensive technique is used for the initial indexing of a document collection. As the system handles more and more queries, the indexing is constantly corrected in a direction which improves performance. Similarly, in a field where the vocabulary is changing, correction can automatically be made in the direction of the new vocabulary.

Further research is required to see which of these suggestions is of practical value.

References

[1] Friedman, S. R., Maceyak, J. A., and Weiss, S. F., A Relevance Feedback System Based on Document Transformation, Report No. ISR-12 to the National Science Foundation, Information Storage and Retrieval, Department of Computer Science, Cornell University, Ithaca, N. Y., Section X, June 1967.

[2] Davis, M., Linsky, M., and Zelkowitz, M., A Relevance Feedback System Employing a Dynamically Evolving Document Space, Computer Science 221 Term Project, Cornell University, Spring 1968.

Appendix

The Normal Precision and Normal Recall measures shown in Table 1 are global measures. It is often also helpful to visualize the effects of retrieval runs with the use of non-global measures. The usual recall and precision measures used for this purpose are defined as:

$$\text{Recall} = \frac{\text{No. of relevant documents retrieved}}{\text{No. of relevant documents}}$$

$$\text{Precision} = \frac{\text{No. of relevant documents retrieved}}{\text{No. of documents retrieved}}$$

The measures are computed as each document is retrieved for a given query.

In the test runs, a record was kept of the first ten documents retrieved for each query. The following graphs show recall-precision measures for four randomly chosen queries with respect to the first ten documents retrieved in the modified and the unmodified document spaces. For each query, three graphs are included, indicating the effects of document space modification by different values of alpha.

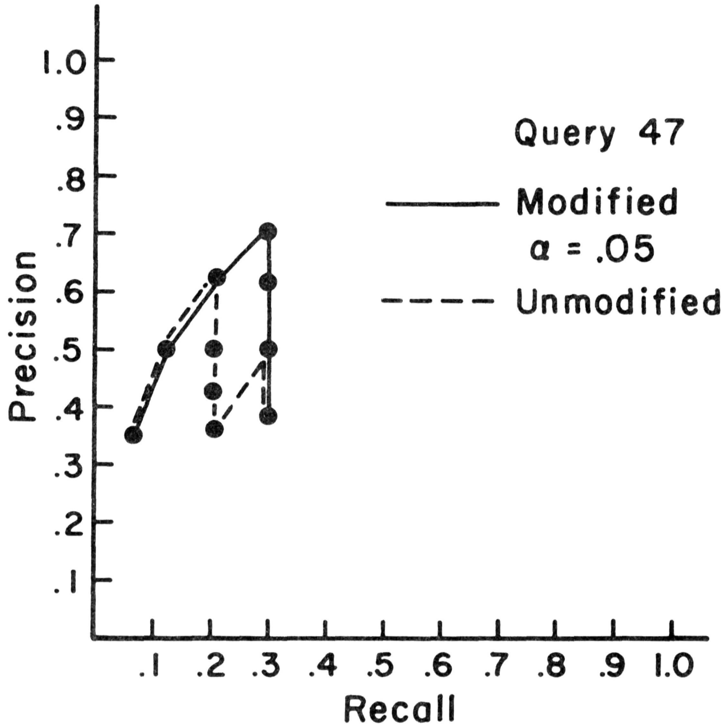

1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
Precision
Query 47
Modified
α = .05
Unmodified
.1 .2 .3 .4 .5 .6 .7 .8 .9 1.0
Recall

Fig. A1

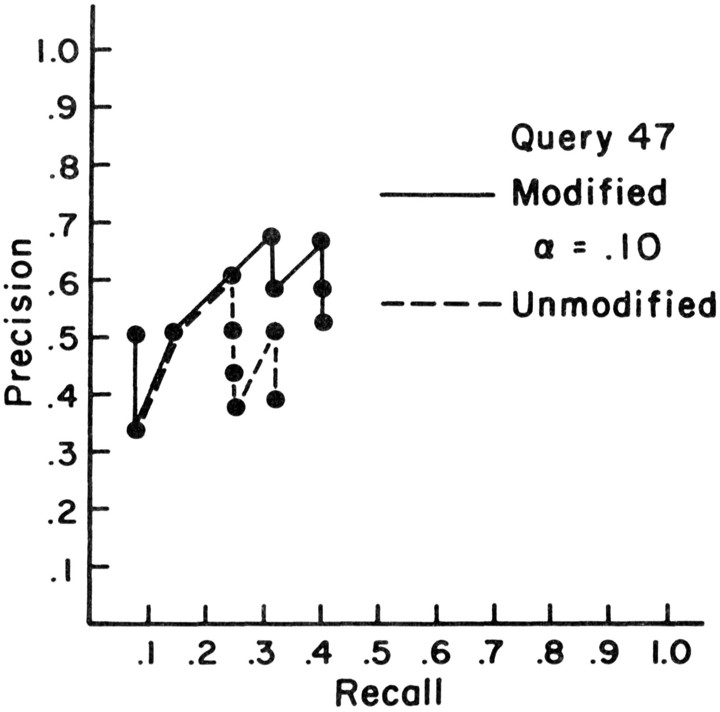

1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
Precision
Query 47
Modified
α = .10
Unmodified
.1 .2 .3 .4 .5 .6 .7 .8 .9 1.0
Recall

Fig. A2

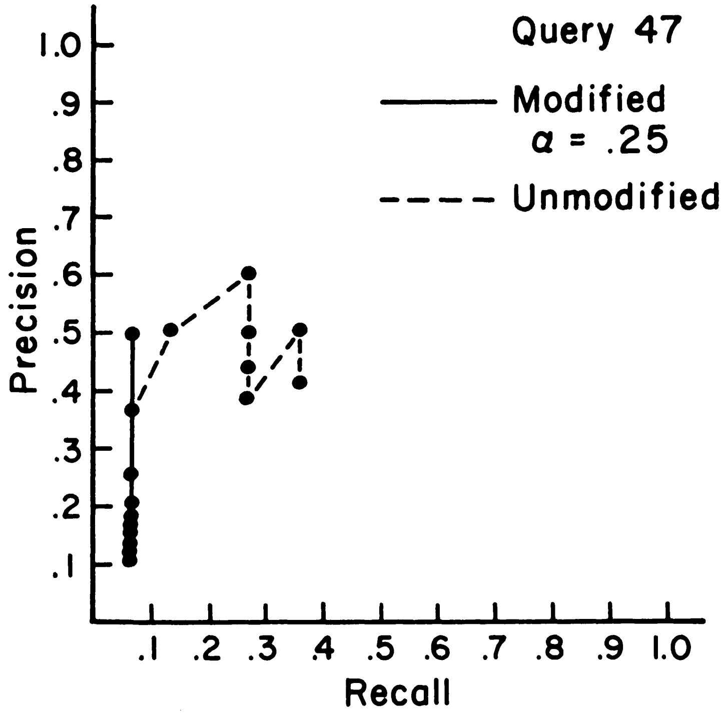

Query 47
Modified
α = .25
Unmodified
Precision
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
.1
.2
.3
.4
.5
.6
.7
.8
.9
1.0
Recall

Fig. A3

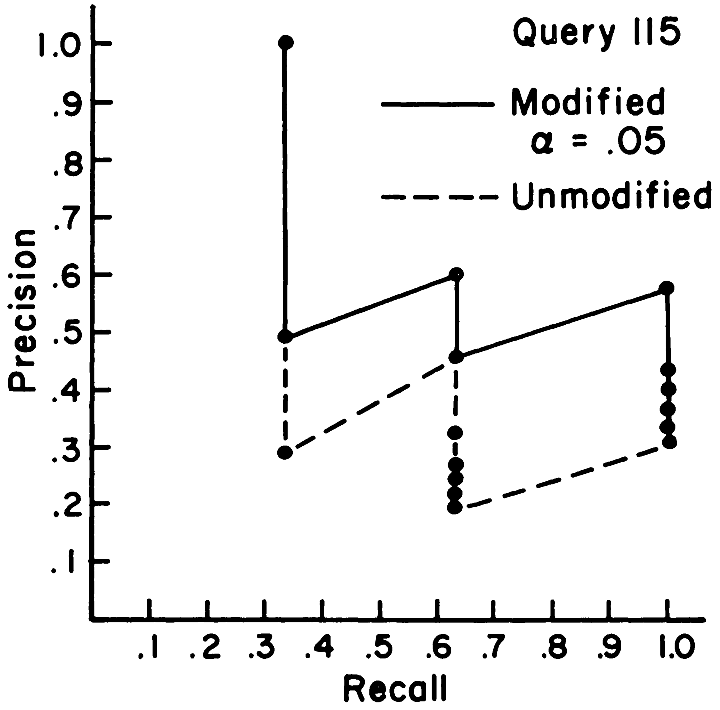

Query 115
Modified
α = .05
Unmodified
Precision
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
.1
.2
.3
.4
.5
.6
.7
.8
.9
1.0
Recall

Fig. A4

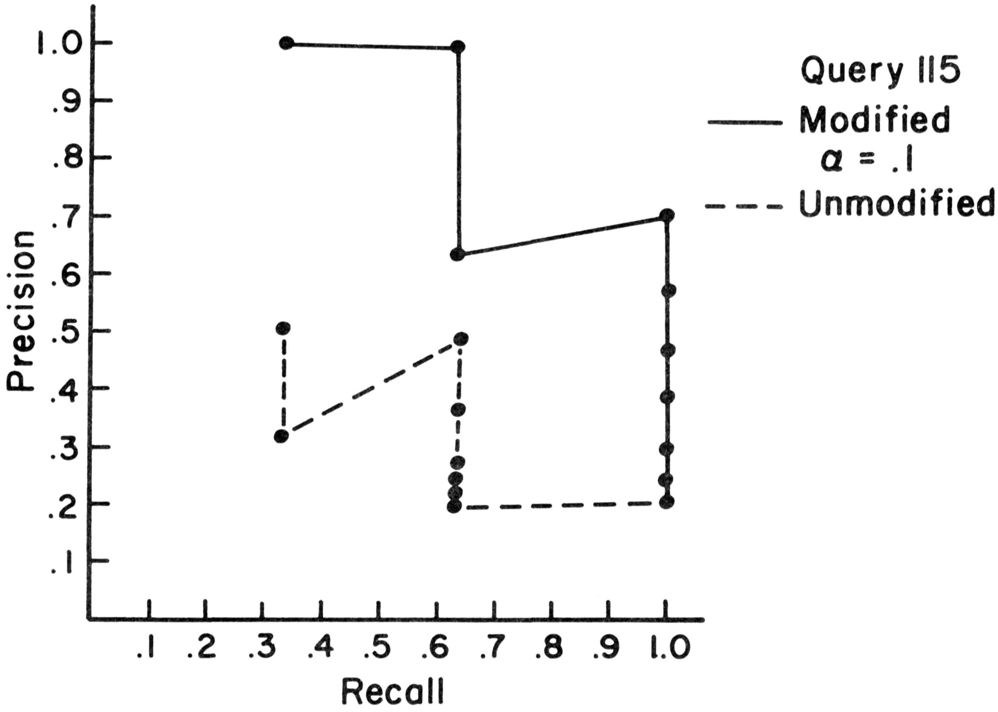

1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
Precision
.1
.2
.3
.4
.5
.6
.7
.8
.9
1.0
Recall
Query 115
Modified
α = .1
Unmodified

Fig. A5

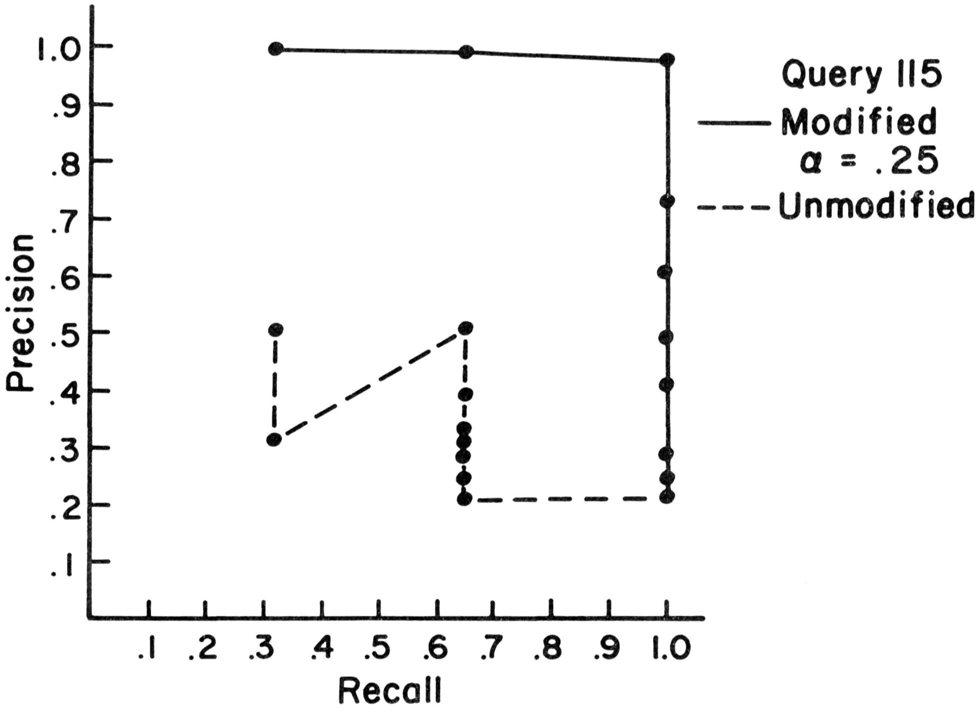

1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
Precision
.1
.2
.3
.4
.5
.6
.7
.8
.9
1.0
Recall
Query 115
Modified
α = .25
Unmodified

Fig. A6

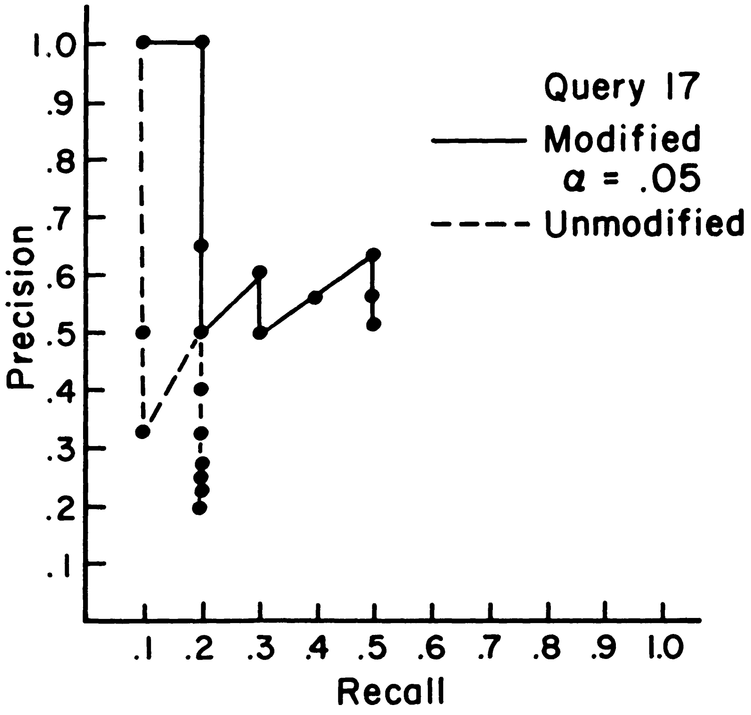

Precision
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
.1
.2
.3
.4
.5
.6
.7
.8
.9
1.0
Recall
Query 17
Modified
α = .05
Unmodified

Fig. A7

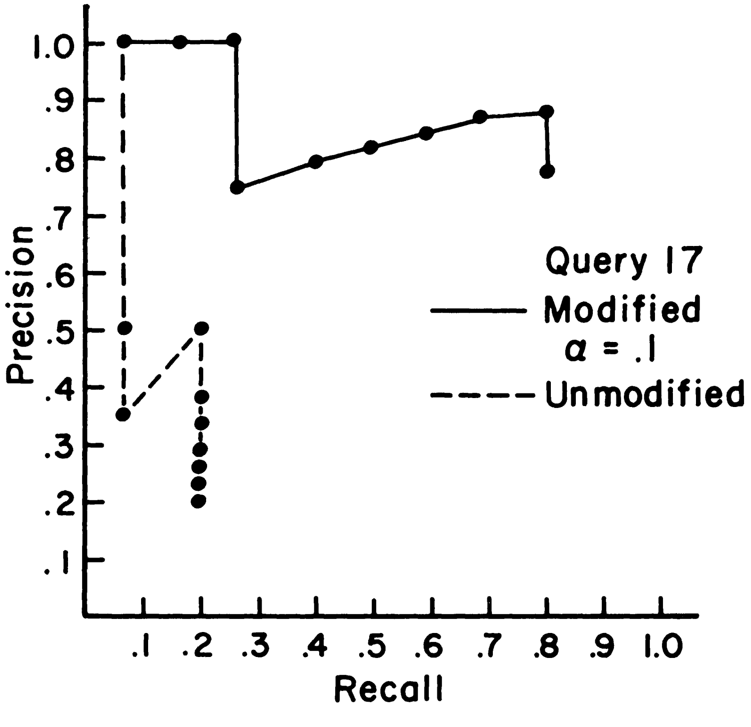

Precision
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
.1
.2
.3
.4
.5
.6
.7
.8
.9
1.0
Recall
Query 17
Modified
α = .1
Unmodified

Fig. A8

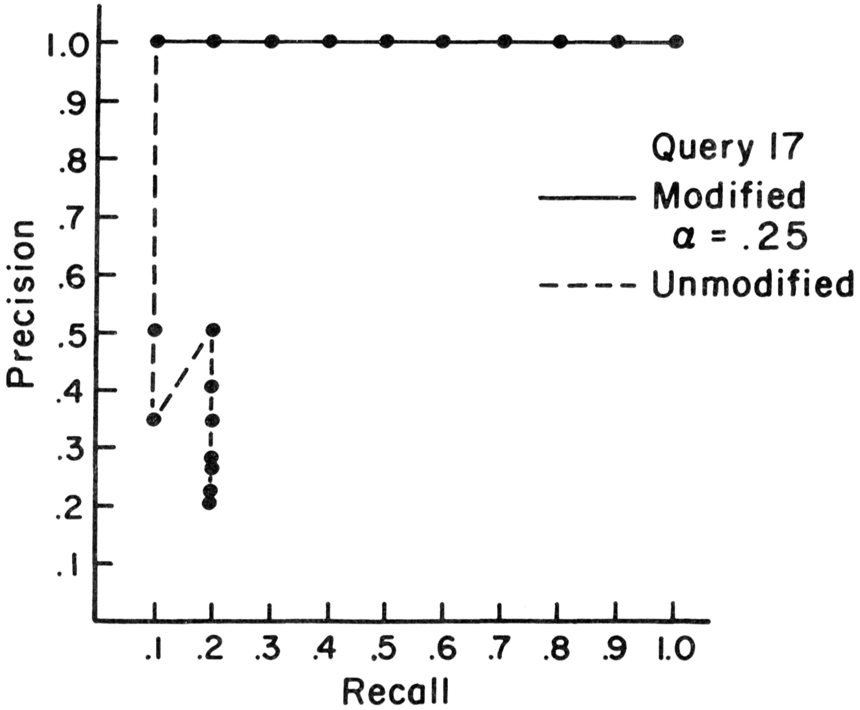


Fig. A9

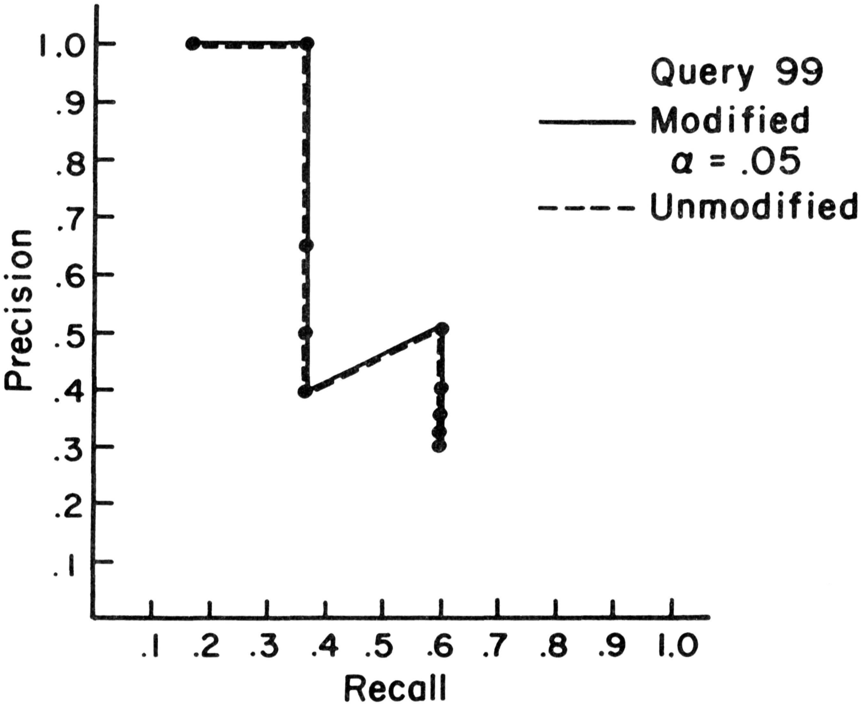


Fig. A10

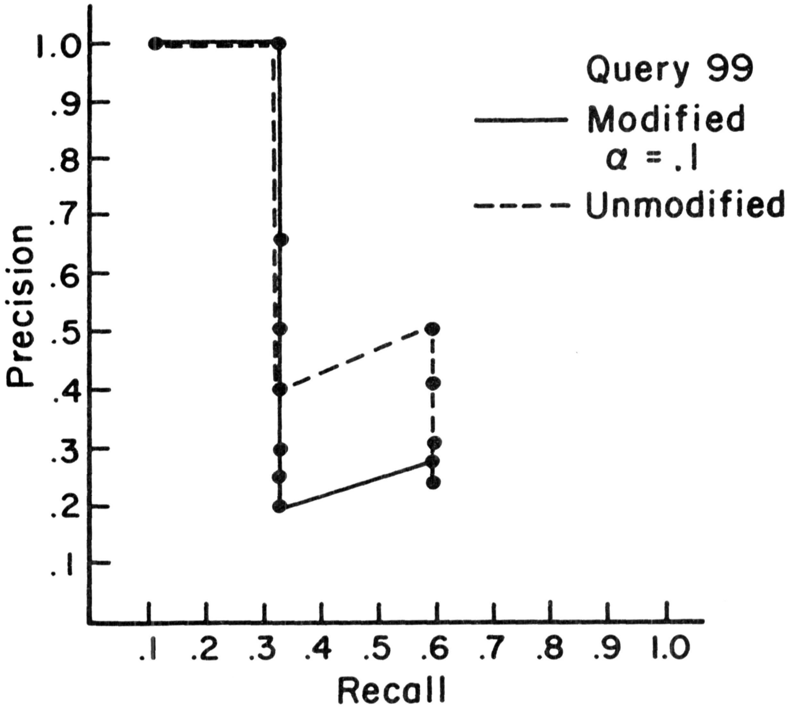


Fig. A11

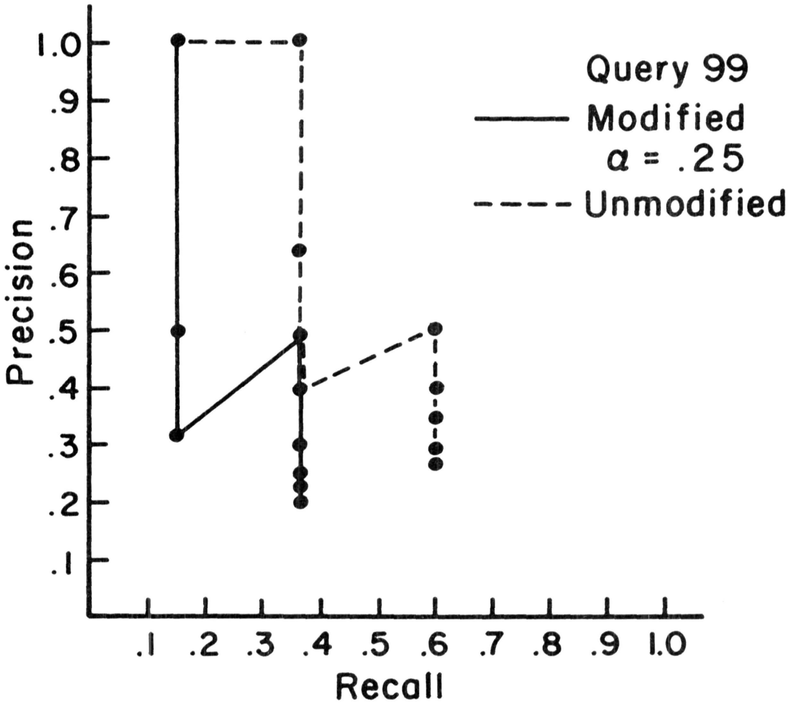


Fig. A12