DOCUMENT RESUME

ED 032 217 SE 007 470

By-Salton, Gerard
Information Storage and Retrieval, Scientific Report No. ISR-15.
Cornell Univ., Ithaca, N.Y. Dept. of Computer Science.
Spons Agency-National Science Foundation, Washington, D.C.
Pub Date Jan 69
Note-199p.
EDRS Price MF-$1.00 HC-$10.05
Descriptors-*Computer Science, *Information Retrieval, *Information Science, *Information Storage

Several algorithms were investigated which would allow a user to interact with an automatic document retrieval system by requesting relevance judgments on selected sets of documents. Two viewpoints were taken in evaluation. One measured the movement of queries toward the optimum query as defined by Rocchio; the other measured the retrieval experienced by the user during the feedback process. Both methods indicated that relevance feedback was effective. When only relevant document vectors were used, the algorithms provided equally good retrieval. Algorithms using nonrelevant document vectors for feedback improved the retrieval obtained by these users without requiring additional relevance judgments. No single feedback strategy was found to give superior retrieval for all queries. However, all relevance feedback algorithms tested improved the average retrieval obtained. (RR)

DEPARTMENT OF COMPUTER SCIENCE

CORNELL UNIVERSITY

INFORMATION STORAGE AND RETRIEVAL

Scientific Report No. ISR-15

to

The National Science Foundation

Ithaca, New York
January 1969

Gerard Salton
Project Director

Department of Computer Science

Cornell University

Ithaca, New York 14850

Scientific Report No. ISR-15

INFORMATION STORAGE AND RETRIEVAL

to

The National Science Foundation

Ithaca, New York

January 1969

Gerard Salton

Project Director

Dedication

This thesis is gratefully dedicated to my husband, Robert Ide, whose help and patience made my graduate work possible.

Acknowledgments

The author wishes to acknowledge the programming support of Robert Williamson, and the advice and encouragement of Professor Gerard Salton and Michael Keen. The free computer time supplied by Hydra Computer Corporation is also appreciated.

Summary

Many automatic document retrieval systems represent documents and requests for documents as numeric vectors that indicate the subjects treated by the document or query. This study investigates relevance feedback, a process that allows user interaction with such a retrieval system. The user is presented with a small set of possibly relevant documents, and is asked to judge each as relevant or nonrelevant to his request. The numeric vectors representing the judged documents are used to modify the numeric vector representing the query, and the new query vector is used to retrieve a more appropriate set of documents. The relevance feedback process can be iterated as often as desired. Several feedback algorithms are investigated in a collection of 200 documents and 42 queries. Two distinct viewpoints are taken in evaluation; one measures the movement of the query vector toward the optimum query defined by Rocchio, the other measures the retrieval experienced by the user during the feedback process. Several performance measures are reported from each evaluation viewpoint. Both evaluation methods indicate that relevance feedback is an effective process.

All algorithms tested that use only relevant document vectors for feedback provide equally good retrieval. Such algorithms should supply additional documents to any user who judges every document presented for feedback to be nonrelevant. Algorithms using nonrelevant document vectors for feedback improve the retrieval obtained by these users without requiring additional relevance judgments.

The relevance feedback algorithms tested are based on the assumption that the vectors representing the documents relevant to a query are clustered in the same area of the document space. The conclusion that no tested relevance feedback algorithm is completely appropriate for the experimental environment is supported by a hypothesis that explains the observed contrasts between the behavior of strategies using only relevant documents for feedback and that of strategies using nonrelevant documents. This hypothesis states that for most queries, some relevant document vectors are separated from others by one or more nonrelevant document vectors. The implications of this result for future research in relevance feedback, partial search or multi-level strategies, multiple query strategies, request clustering, and document vector modification are discussed, and useful evaluation measures and new algorithms for these areas are suggested.

Table of Contents

Table of Contents (contd)

List of Figures

List of Figures (contd)

List of Figures (contd)

Synopsis

The present proliferation of technical and scientific books and articles is too much of a good thing. The so-called "information explosion" has created a need for new methods of document classification and retrieval with the following characteristics:

1) The number of expert indexers and librarians obtainable is completely inadequate for processing such a volume of information, even in a cursory manner. An automatic computerized system is needed. [1]

2) Even with high speed computers, the retrieval operation must be simple in order to minimize time and cost.

3) The user should not be required to understand the detailed operation of the system. For this reason, the system should ideally respond to information requests in natural language.

This study deals with an experimental automatic document retrieval system that meets these requirements. Within this experimental system, several retrieval algorithms that permit user interaction with the search process are evaluated. All of these techniques employ user judgment of the relevance of certain selected documents to his request, and are called "relevance feedback" algorithms.

Section I of this report describes several methods of automatic document retrieval, and details the experimental retrieval system used in this study (the SMART system [3]). Section II examines several means of user interaction with an automatic retrieval system, and summarizes the results of some prior experiments with user interaction in the SMART system. In Section III the results of earlier experiments with several relevance feedback algorithms are presented. Section IV details the features of the experimental environment of this study. In Section V, means of evaluating the performance of an information retrieval system are discussed, and the evaluation measures and statistical tests used in this study are described. Section VI contains the results of relevance feedback experiments in five areas. In Section VI-A relevance feedback results in two document collections are compared. Section VI-B compares feedback algorithms that use only information from the documents judged "relevant" by the user, and Section VI-C examines the effect on these algorithms of the number of documents used for feedback. Section VI-D investigates strategies that use information from relevant and nonrelevant documents, and Section VI-E further studies the comparative usefulness of these strategies for different types of queries. In Section VII, the last section of this study, the results of these relevance feedback experiments are used to support recommendations for interactive document retrieval systems, and to suggest guidelines for future experiments with regard to relevance feedback algorithms, evaluation of feedback performance, partial search strategies, multiple query strategies, request clustering, and permanent document vector modification.

Relevance Feedback

In An Automatic Document Retrieval System

Chapter I

Automatic Document Retrieval Systems

The conventional library classifies documents by numeric subject codes which are assigned manually (Dewey decimal system, Library of Congress system). Cross-indexing is provided in a card file by "subject" title, and author. Both the numeric index and the "subject" cross-indexing may be inadequate for retrieval. A book on the intersection of two subjects (e.g., "The Aerodynamics of Birds") or on a new subject (e.g., automata theory --- is it mathematics, computer science, logic?) is hard to classify and therefore hard to find, unless the librarian is asked where he filed it. A fully automatic system must duplicate the function of this librarian by extracting information from a natural language request and retrieving precisely those documents most likely to be needed by the requestor.

One method of subject classification that is used in automatic retrieval systems assigns to each document a list of subject identifiers, often called "keywords". This list can be treated as a binary vector by associating a position in the vector with each possible keyword in the retrieval system. The value in a vector position is one if the associated keyword is assigned to the document described by the vector, zero otherwise. Retrieval systems operated by NASA and by the National Library of Medicine (Medlars) use this type of subject classification [2]. In both of these automatic retrieval systems, keywords are assigned to documents manually by subject experts.

An extension of keyword indexing represents each document as a positive weighted concept vector rather than a binary vector. Each classification concept is weighted to indicate its importance in the document.

In the SMART retrieval system, these concepts and weights are assigned by automatic processing of the natural language text of each document or abstract [3].

The user's query in an automatic information retrieval system can take several forms. The user may be asked to formulate his query using a restricted language. This language usually includes the set of keywords defined for the collection and sometimes the Boolean operations "and", "or", and "not". In the NASA system the user can assign values to each keyword instead of using the logical operations. Cross-referencing and hierarchal relationships among keywords can be used in both the NASA and Medlars systems to refine or expand the user's initial query. Both systems require the user to understand the indexing system in order to formulate effective search requests.

In the SMART retrieval system, the user is asked to phrase his query in natural language. The query is then processed in the same way as the document abstract, and a query concept vector is created. Logical relationships are not used in the query analysis.

SMART provides a fully automatic information storage and retrieval system of relatively simple form. Document abstracts are analyzed to construct representative concept vectors which are stored in the computer. When a user types a natural-language request into the system, it is converted to a concept vector representation in the same manner. Several types of automatic text-to-vector conversions have been used with the SMART system [3]. A list of common words to be ignored in constructing the concept vector is provided. A suffix dictionary is used to reduce all words to word stem form. A word stem thesaurus which treats each distinct word stem as a concept in the concept vector is used as an experimental standard. Frequency characteristics may be used to eliminate some concepts ("partial stem thesaurus"), for example, words occurring less than five or more than 100 times in a given collection may be eliminated. This study uses a thesaurus which was constructed semi-automatically for the subject area of aeronautical engineering. This "regular thesaurus" recognizes synonyms; that is, it converts words of the same meaning to the same concept, providing better retrieval performance than the stem thesaurus [4].

The degree of relationship between a query and a document is determined in the SMART system by some "distance function" of the query and document concept vectors. The most effective of the distance functions tested in SMART appears to be the cosine correlation, which measures the angle between concept vectors in n-dimensional space [4,5]. The cosine coefficient of two concept vectors ranges from 0 to 1, and is found by the formula

$$\cos(r,s) = \frac{\sum_{1}^{n} r_i s_i}{\sqrt{\sum_{1}^{n} r_i^2 \sum_{1}^{n} s_i^2}}$$

The process of determining the relationship of each document in the collection (or some subset thereof) to the user's query is called a "search" operation. The distance function is used to assign to each document a correlation coefficient indicating the relationship between the concept vector for that document and the query vector. The document identification numbers are then ordered by correlation coefficient and are assigned ranks from one to N (number of documents being searched) for evaluation purposes. The document cost closely related to the user's query is assigned the rank 1 (considered the "highest" rank).

The retrieval algorithm is the goal of any information retrieval system. For each query, the system must produce a set of documents relevant to the requestor's need. In the SMART system the retrieval algorithm can be varied experimentally. The retrieval algorithm applies the search operation; it may select subsets of documents to be searched, and it may conduct several search operations in response to one user request. The details of the retrieval operation are the primary concern of this study.

One of the simplest retrieval algorithms, here called the "full search" algorithm, performs one search operation using the entire document collection, and selects for retrieval the highest n documents in the ranked list resulting from the search operation. In an operating system each user could select this n; in the SMART system n is an experimental parameter.

Chapter II

User Interaction With An On-Line Retrieval System

The full search retrieval algorithm returns to the user the n documents with concept vectors "closest" to the query vector as measured by the angle between vectors (cosine correlation). If the user's original query is an accurate and complete description (in "concepts") of his need, and if the documents relevant to the user are clustered "close" together in the space of concept vectors, this algorithm can isolate these few relevant items from a large collection of irrelevant material. However, neither of these conditions is common in practice. It is evident from experiments with the SMART system that a user familiar with the subject area but unaware of vocabulary and word frequency effects on the search process is unlikely to formulate an initial query that provides optimum retrieval [6]. It is unreasonable, however, to expect each user to understand the fine details of the document classification system.

Further, there is evidence in the experimental document collection used here that the documents judged relevant by the users are not always clustered neatly in the concept vector space. Even with full knowledge of the document collection it is often impossible to formulate a single query that will rank all relevant documents above all nonrelevant documents. This may indicate flaws in the text-to-vector mapping used for this study. However, the needs of the human users of document collections are so diverse that a subject classification system appropriate for all queries may not exist, or may be impractical to implement.

Since the user's original query is often inadequate, some sort of user interaction with the retrieval operation is desirable. The user of a manual retrieval system such as a library might at first ask a general and unclear question. The librarian, using his knowledge of the document collection, might then ask the user a few questions and show him a few books in an attempt to pinpoint his needs. Recent technological developments encourage the investigation of similar types of user feedback in automatic retrieval systems. Large capacity random access memory devices allow the storage of natural language document titles and abstracts. On-line low speed terminals and time-sharing techniques may be used to provide real time interaction with many users at once, at several convenient locations.

Two major considerations arise in such an on-line system. In the present batch-processing systems, such as NASA and Medlars [2] immediate response to the user is not necessary. In an interactive system the computer time required to process a single query takes on a new importance. The low input-output speeds of those terminals appropriate for interactive applications introduce a second limitation [7]. For example, typing out a single document abstract on a typewriter terminal could easily consume more time than the computer retrieval operation. An interactive document retrieval system therefore requires an efficient retrieval algorithm and a minimum of necessary interactive input and output.

Several methods of user interaction have been tested in the SMART system using the document collection employed in this study (the 'Cranfield 200' collection described in Section IV). Results of this investigation [6] are summarized below.

The interactive strategies tested can be divided into pre-search and post-search algorithms. In pre-search interaction, information is presented to the user and a new query is constructed by him before the search operation takes place. The "repeated concepts" algorithm asks the user to choose one or more of his query terms to be repeated for emphasis. The "word frequency" technique displays for the user the frequencies with which his query terms occur in the document collection. The user is then invited to eliminate or change query terms that are too common or too rare to be useful for retrieval. Both of these displays help the uninformed user to take advantage of the effects of word frequency in a retrieval system using frequency-weighted vectors for document classification. The "thesaurus display" supplies synonyms and terms related to the terms of the initial query from a stored thesaurus appropriate to the subject area. The thesaurus used for this display in reference 6 is the "regular thesaurus" described in Section I of this report. Since the same thesaurus can be incorporated automatically into the SMART system, manual and automatic thesaurus procedures are compared in reference 6. The automatic application of the thesaurus to document and query vectors gives better retrieval results than the manual thesaurus display, except at low recall levels. The "source document display" exhibits concepts assigned to a relevant document known to the user before retrieval. When this display is used in addition to the automatic thesaurus, results are better than with automatic thesaurus alone.

Post-search techniques display the partial results of an initial search operation so that the user can reformulate his query and request another search. These algorithms may be iterated as often as the user desires. All post-search algorithms share a common disadvantage, the computer time required for several search operations. The time is well spent, however, for all post-search techniques investigated give better retrieval than automatic thesaurus display. "Title display" which displays the titles of the first n (in this reference n=5), documents retrieved by the initial query, provides better retrieval than thesaurus display except at high recall. "Abstract display", which displays n full abstracts, requires more output time and more time for user thought, but gives consistently better performance than title display. A variation of "relevance feedback", the technique investigated in this study, gives retrieval results nearly comparable to abstract display. Moreover, this report gives more effective variations of the relevance feedback algorithm than the version used by Lesk and Salton.* When pre-search and post-search information is combined, manual thesaurus display followed by abstract display gives better retrieval than either method alone. Adding word frequency information to the combination is helpful when the word stem thesaurus is used.

Estimates of the search cost per query show that abstract display, which gives the best overall performance of the methods tested, is the most expensive. The other post-search algorithms, title display and relevance feedback, are more costly than any pre-search method. Relevance feedback requires the least user effort of any post-search strategy. Lesk

*Lesk and Salton use the Q_o Strategy with N equal to 5. See Sections VI-C and VI-D for more effective algorithms.

and Salton [6] recommend the following algorithms:

a) For normal users needing high recall, automatic thesaurus followed by automatic relevance feedback.

b) For highest precision when high recall is not required, word stem matching followed by title display.

c) For experienced and patient users needing maximum performance, thesaurus display plus frequency information followed by abstract display.

The Lesk and Salton study shows that relevance feedback is one of the most effective user interaction techniques. In relevance feedback, the user is given a small set of items retrieved using his original query. He is then asked to judge which items of this set are relevant to his needs. This information is used to automatically produce a new query for another search. This feedback process can be iterated as often as desired. Relevance feedback has a definite psychological advantage over abstract display; the user is not required to make sophisticated decisions in rephrasing his own query. Instead, he can supply much information to the retrieval system at little effort by saying in effect "I want documents on the same subject as this document". The stored abstract of a chosen relevant document contains a more detailed description of the subject than a user would care to type as a query. In the experimental collection, the document vectors have approximately ten times as many concepts as the query vectors, so the user submits a ten times more detailed "query" simply by typing a document identifying number.

The disadvantages of relevance feedback should be reiterated. Like abstract display, relevance feedback requires the system output of document abstracts or information of comparable detail. Also, multiple searches of the document collection are made. Designers of retrieval systems must decide whether the extra output time and computer time is justified by the retrieval improvements obtained.

Chapter III

Prior Investigations Of The Relevance Feedback Retrieval Algorithm

Rocchio [8,9,10] suggests an algorithm for relevance feedback based on the properties of the distance function used. If the set of relevant documents is known, the query that will be "closest" to this set of documents and furthest from the set of non-relevant documents can be formed. If the cosine correlation is used as a distance function, this ideal query is

$$q = N_s \sum_{1}^{N_r} \frac{r^i}{\sqrt{(r^i)^2}} - N_r \sum_{1}^{N_s} \frac{s^i}{\sqrt{(s^i)^2}}$$

where each r^i is the vector describing a document relevant to the user's query, and each s^i is the vector describing a document not relevant. Thus N_r is the number of documents in the collection that are relevant to the request, and N_s is the number of non-relevant documents, or the remainder of the collection.

This ideal query is useless for retrieval, because if the documents relevant to each request were known, a retrieval operation would not be needed. Rocchio suggests that the ideal query might be approached by iteration. The user is asked to make relevance judgments on a small retrieved set of documents, and this set is used to update the former query as follows:

$$q_{i+1} = n_r n_s q_i + n_s \sum_{1}^{n_r} \frac{r^i}{\sqrt{(r^i)^2}} - n_r \sum_{1}^{n_s} \frac{s^i}{\sqrt{(s^i)^2}} \qquad \text{(A)}$$

where n_r and n_s are the numbers of relevant and non-relevant documents retrieved by the previous search operation.

Rocchio investigated relevance feedback using formula A and the SMART retrieval system [9]. A set of seventeen natural language search requests and a collection of 405 abstracts of articles published in IRE Transactions on Electronic Computers (March-September, 1958) were indexed using a SMART regular thesaurus (Section II). Relevance judgments for the sample queries were constructed by a manual search of the entire document collection. Average retrieval results for the collection described are improved by two iterations of the relevance feedback process described in formula A. Rocchio suggests construction of multiple queries when the documents desired are not clustered in the document vector space.

Another investigation of a relevance feedback system was based on the "ADI collection", a collection of 82 documents presented at a conference on documentation. Thirty-five queries were constructed for this collection, and the documents considered relevant to those requests were specified by the two originators of the queries. The investigation of relevance feedback in the ADI collection was conducted by Riddle, Horwitz, and Dietz [11]. They used 22 of the 35 queries and studied a slightly different algorithm for modifying the search query. Their formula is:

$$Q_i+1 = Q_i + \alpha \sum_{1}^{n_r} r_i \qquad \text{(B)}$$

Three differences from Rocchio's formula are immediately apparent:

a) The descriptor vectors are not normalized by their length. In Rocchio's formula, the change to the weight of concept a in the query depends not only on the weight assigned to concept a in a retrieved document vector but also on the length of that document vector; that is, on the number of other concepts and on the magnitudes of weights in the document vector. This is not the case in the Riddle, Horwitz, and Dietz formula. When the latter formula is used, for instance, a document with generally highly weighted concepts changes the query more than does a document with generally lower weighted concepts, the number of concepts being equal. Weight magnitudes being roughly equal, a document with more concepts changes the query more than one with fewer. Rocchio's formula compensates for these effects.

b) The parameter α, which is the one variable in the above formula, is constant for all queries. Rocchio's formula uses a different multiplier for each query; the multiplier being dependent on the numbers of relevant and non-relevant documents retrieved (n_r and n_s).

c) The non-relevant documents retrieved on the previous iterations are not used to update the query. However, Riddle, Horwitz, and Dietz tested a "negative heuristic strategy" which uses the two non-relevant documents first retrieved (the two which the system falsely judges most relevant to the query) to update those queries that retrieve no further relevant documents on the first feedback iteration. For such queries the formula becomes:

$$Q_{i+1} = Q_i + \alpha \sum_{1}^{n_r} r_i - \sum_{1}^{2} s_i \qquad \text{(C)}$$

The feedback algorithm of Riddle, Horwitz, and Dietz produces an improvement in performance on most of the queries tested. The three experimenters recommend that the variable α in their formula be set to 1 for the first iteration and then increased by 1 for each subsequent iteration (called "increasing alpha strategy"). They also recommend their negative heuristic strategy (formula C).

Crawford and Melzer [12] have tested a relevance feedback strategy that ignores the original query after the initial search if a relevant document is found. Their algorithm is:

If at least one relevant document is retrieved within the first n documents, the original query is ignored and one additional relevant document is used for each iteration:

$$Q_{i+1} = \sum_{1}^{i} r_i \quad .$$

But if no relevant documents are retrieved within the first n

$$Q_2 = Q_1 - S_1 \quad .$$

On iterations after the first, the second formula of their strategy is not used.

Using the Cranfield 200 document collection (Section IV), Crawford and Melzer found their strategy superior to formula B when $\alpha = 1$.

Steinbuhler and Aleta [13] have tested Rocchio's algorithm in the ADI collection, for the 'worst case' when only non-relevant documents are available for feedback after the first search operation. The information from non-relevant documents alone, when used to modify the query according to formula A, gives better retrieval than the initial query.

Kelly [14] proposes an addition to the relevance feedback algorithm when no relevant documents are retrieved. He points out that in these cases no new concepts are added to the query, and recommends adding concepts that occur frequently in the document collection. He tested this recommendation on sets of artificially constructed 'query' and 'document' vectors with success. However, Steinbuhler and Aleta [13] found that adding frequent concepts to the query degraded performance in the ADI collection.

The results reported in Section VI of this study give further insight into the problems investigated by the five earlier studies cited. Section VII uses both the earlier studies and the present report to support recommendations for document retrieval systems.

Chapter IV

Environment Of The Reported Experiments

The present study of relevance feedback compares several related formulas for query modification. The following general formula is available to the experimenter:

$$Q_i+1 = \pi Q_i + \omega Q_o + \alpha \sum^{\min(n_a, n_r)} r_i + \mu \sum^{\min(n_b, n_s)} s_i \qquad (D)$$

where $n_r + n_s$ (see formula A, Section 1) equals N, the number of documents retrieved for feedback.

The experimental variables are α, ω, π, μ, n_a, n_o, and N. The parameter α is positive, and weights all incoming relevant documents relative to other contributors to the query (previous query, initial query, non-relevant documents). The parameter π permits the previous query to be increased in weight relative to the incoming documents. Q_o is the initial query, as opposed to the query of the previous iteration; ω permits the initial query to be used as part of the new query (see Section 3B). The parameter μ should theoretically be negative, as it permits some significance to be attached to the non-relevant documents retrieved. The parameter n_a (n_b) permits some specific number of relevant (non-relevant) documents to be used in the query even if n_r (n_s) is larger. It is assumed that the r_i and s_i are indexed in order of decreasing relevance (as determined by the system) to the query; that is, the n_a relevant documents (or n_b non-relevant documents) used in the new query will be those closest in the descriptor space to the previous query. The flexibility of this formula permits the investigation of several feedback strategies.

The system also provides the following formula to simulate Rocchio's algorithm:

$$Q_{i+1} = \pi \; n_r n_s Q_i + \omega \; Q_c + n_s \sum^{\min(n_r,\, n_a)} r_i - n_r \sum^{\min(n_s,\, n_b)} s_i \qquad \text{(E)}$$

Formula E does not normalize the vector lengths as is done in Rocchio's algorithm (formula A).

The document collection used in this study (the "Cranfield" collection) contains 200 documents from the field of aerodynamics, chosen from a library of 1400 documents. For this collection, there are 42 queries, constructed by some of the authors of the 1400 documents; these requestors are also responsible for the relevance judgments.

The concept vectors describing document and queries are quite sparse for the "Cranfield" collection. The maximum number of concepts used to describe one document is 85, out of a possible 552 concepts. The largest weight given to any concept in any document descriptor is 288. The query description vectors are sparser by one order of magnitude and shorter than the document descriptors. The maximum number of concepts used in a single query vector is 13; the largest weight in any query vector is 24. The largest number of documents relevant to a single query is 12, or six percent of the collection. The comparative brevity of the query vectors in this collection is typical in technical document retrieval, because document abstracts generally contain more detailed information than user queries.

The characteristics of an experimental document collection determine the extent to which experimental results typify retrieval behavior in 'real' document collections. The three collections described in this study are much too small to require sophisticated retrieval techniques. However, the Cranfield 200 document collection is more realistic than the ADI and IRE collections for three reasons.

First, more queries are available for the Cranfield 200 collection. The number of queries available is important in judging the statistical significance of experimental results.

Second, the documents in the Cranfield 200 collection were chosen from a more typical environment. The ADI collection consists of short papers all presented at the same conference. The papers in the IRE collection were all published in the same magazine within a seven-month period. By contrast, the 1400 documents in the full Cranfield collection were in effect selected by knowledgable authors from the field of aerodynamics, and the 200 documents in the small collection were chosen to represent the larger collection.

Third, the queries and relevance judgments in the ADI and IRE collections were constructed by a small number of information retrieval experts, while those in the full Cranfield collection were constructed by 182 authors of recent papers in aerodynamics.

Concept vectors for both the ADI and Cranfield collections were constructed automatically using a regular subject-area thesaurus. In all experiments reported here, the cosine correlation (Section I) is used as the distance function.

Chapter V

Evaluation Of Retrieval Performance

A. Measures of Performance

Several average measures of the performance of the tested retrieval algorithms on the 42 Cranfield queries are used in this report. Each measure is based on the concept of "recall" and "precision". In evaluating an information retrieval system, an arbitrary cut-off point, such as rank ten or cosine correlation 0.75, is often employed. Documents above this cut-off point in the ranked list resulting from a search operation are considered "retrieved". With such a cut-off, recall is the precentage of documents relevant to the user that are retrieved, and precision is the percentage of retrieved documents that are relevant.

An ideal retrieval system would provide recall and precision of 100%, indicating that all relevant documents are retrieved and no non-relevant documents are retrieved. In SMART experiments an inverse relationship between recall and precision is observed, such that high recall implies low precision and vice-versa.

The 'document curves' used in this report are graphs of recall and precision at several cut-off points based on rank; that is, recall and precision after x documents are retrieved, for several values of x. The other measures used are not based on specific cut-off points, but in a sense measure retrieval performance over the entire document collection.

Normalized recall and normalized precision are two measures proposed by Rocchio [9], that take the average recall and precision obtained for all possible cut-off points. If N is the number of documents in the collection, R_j is the recall at a cut-off of j documents (rank j) and P_j is the precision at a cut-off of j documents, normalized recall and precision are defined as follows [15]:

$$NR = \frac{1}{N} \sum_{j=1}^{N} R_j$$

$$NP = \frac{1}{N} \sum_{j=1}^{N} P_j$$

For automatic calculation, the following approximations are used in the SMART system [15]:

$$NR = 1 - \frac{\sum_{i=1}^{n} r_i - \sum_{i=1}^{n} i}{n\ (N-n)}$$

$$NP = 1 - \frac{\sum_{i=1}^{n} \ln r_i - \sum_{i=1}^{n} \ln i}{\ln \left(\frac{N!}{n!\ (N-n)!} \right)}$$

where r_i is the rank of the i^{th} relevant document in the collection and n is the number of relevant documents in the collection for the given query. A normal overall measure of retrieval performance has been suggested [15], but is not explicitly displayed in this report: Normal overall measure = 1 - 5 NR + NP. The factor of 5 gives equal weight to the two component measures.

Another overall measure used in many studies of retrieval performance is the recall-precision curve, an average plot of precision at each 5% or 10% of recall. Each query is averaged into each point of the plot. To accomplish this averaging process, an interpolation procedure is needed, since, for example, a query with two relevant documents can only achieve uninterpolated recall levels of 50% and 100%.

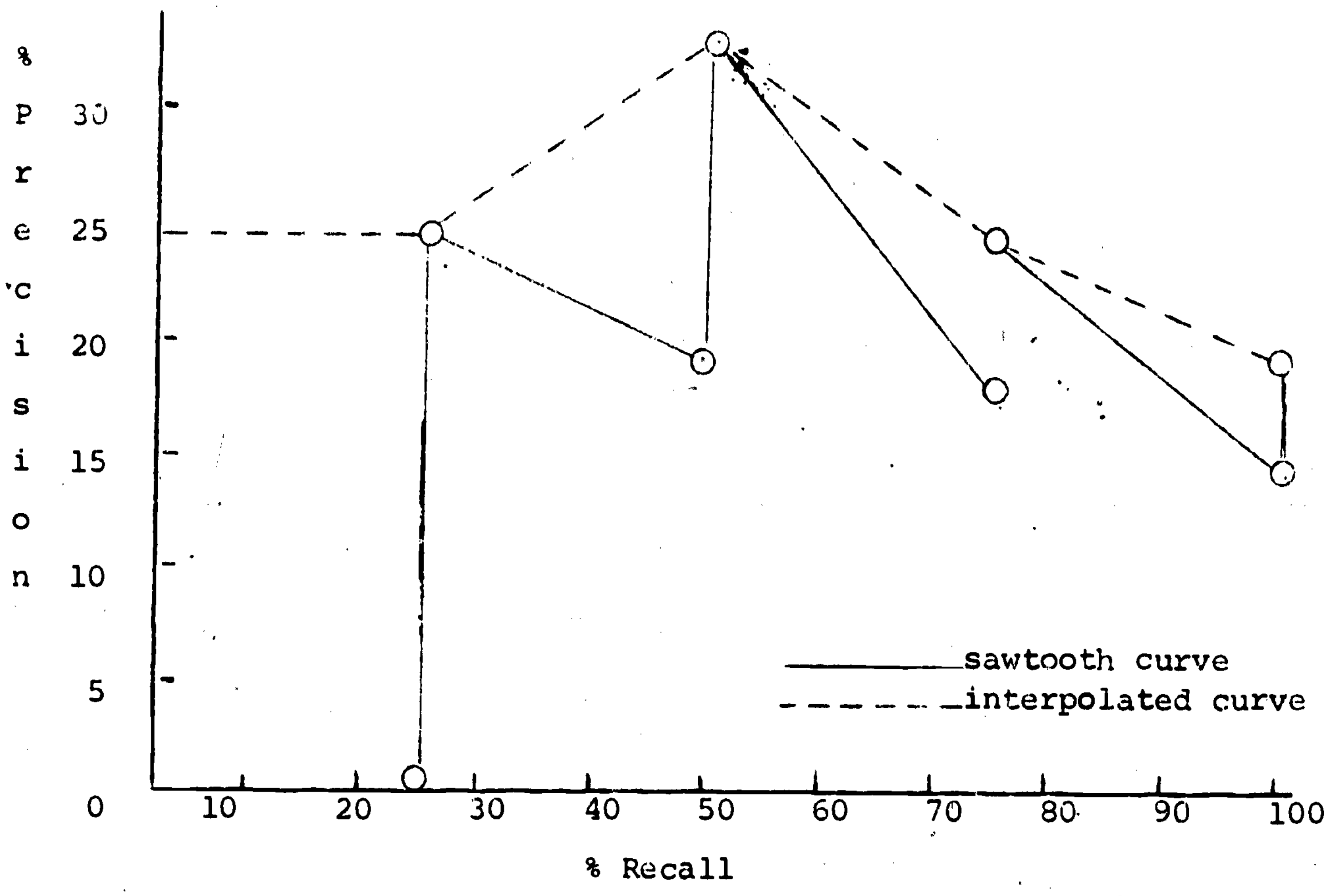
Two types of recall-precision curve are used in this study. They are distinguished by the method of interpolation used. Both the Quasi-Cleverdon interpolation used in several previous studies and the Neo-Cleverdon interpolation now used for all evaluation of the SMART system are described below.

Figures 1 and 2 show two graphs for a hypothetical query having 4 relevant documents. The relevant documents are assumed to be retrieved with ranks of 4, 6, 12 and 20. Thus, at 25% recall, the precision is 25%, a 50% recall, the precision is 33%, and so on. However, these values correspond actually to the highest possible precision points, since they are calculated just after a relevant document is retrieved. In this example, after 3 documents are retrieved, the precision is 0%, after 5 documents, the precision is 20%, and so on. This range of precision for each recall level is indicated by the top and bottom points in Figures 1 and 2 at 25%, 50%, 75%, and 100% recall. The solid sawtooth line connecting these points is not used for interpolation; it is intended to indicate the drop in precision between the actual recall levels for this query as more non-relevant documents are retrieved.

The Quasi Cleverdon interpolation uses a straight line between peak points of precision, as indicated by the dashed line in Figure 1. It has been argued that this interpolation is artificially high, since it lies at all points above the saw-tooth curve, and thus, does not reflect in any way the precision drop as more non-relevant documents are retrieved. The Neo-Cleverdon interpolation of Figure 2 projects a horizontal line leftward from each peak point of precision, and stops when a higher poi't of precision is encountered. This new interpolation curve (the dashed line in Figure 2) does not lie above the saw-tooth curve at all points. When the precision drops from one recall level actually achieved to the next, an immediate drop in precision after the first point to the level of the next point is indicated. For example, in Figure 2, the precision value at 50% recall is 33%, but at 55% recall, the interpolated value used for the new averages is 25% precision. When the precision rises from one recall level to the next, however, the first precision point actually achieved is ignored for purposes of interpolation. The achieved precision of 25% at 25% recall in the example of Figure 2 is ignored, and for all recall levels from 0 to 50%, an interpolated precision of 33% is used for the new averages. The proponents of the new interpolation argue that this method indicates in all cases a precision that the user could actually achieve, if he were to use clairvoyance to retrieve exactly the right number of documents.

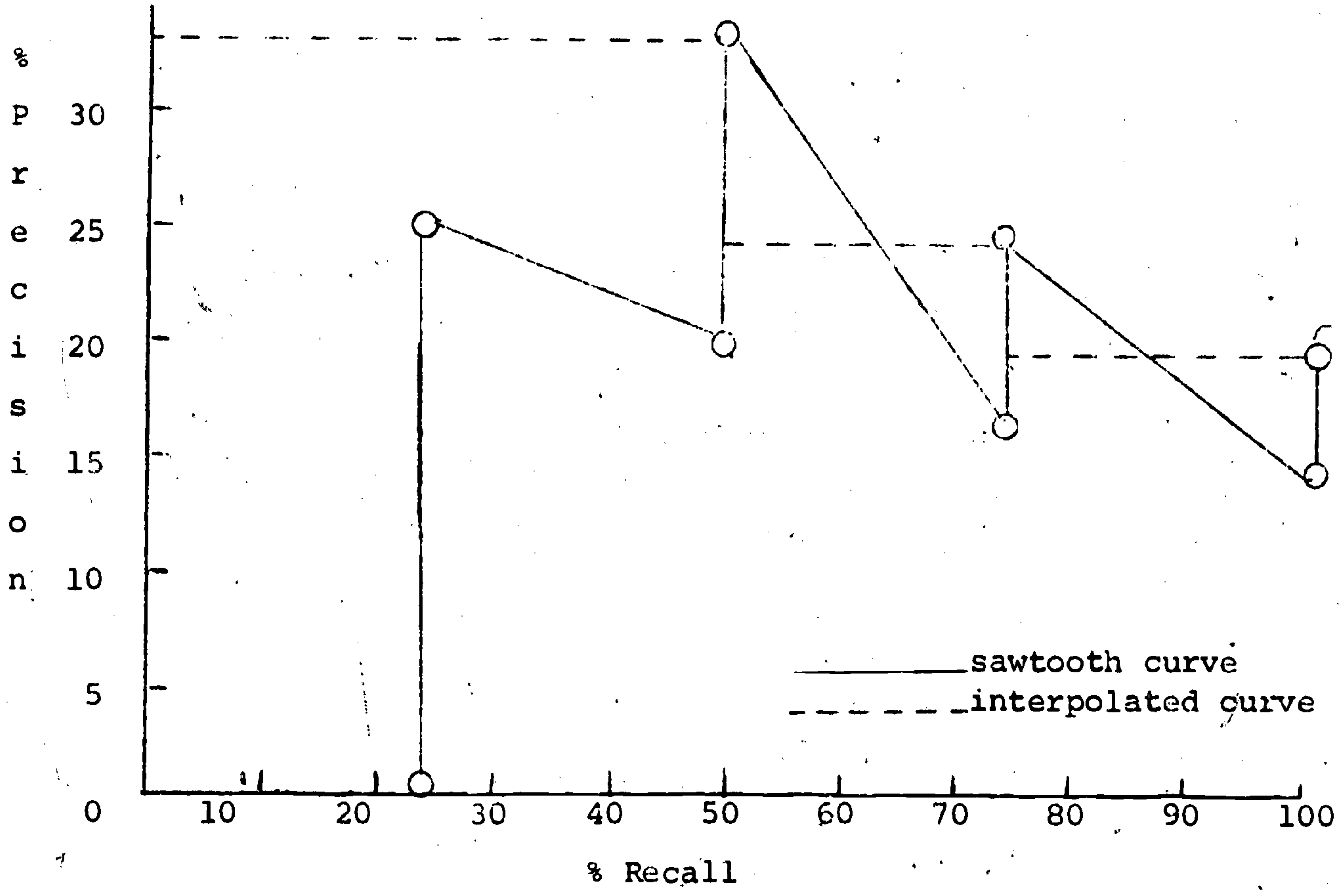
B. Statistical Significance Tests

Several statistical tests are reported here using as input the



An Illustration of the Interpolation Method Used for the Quasi-Cleverdon Recall-Precision Averages

Figure 1



An Illustration of the Interpolation Method Used for the Neo-Cleverdon Recall-Precision Averages

Figure 2

rank recall, log precision, normalized recall, normalized precision and 10 points from the feedback effect (Section V-C) recall-precision curve with the Neo-Cleverdon interpolation. The statistical tests are intended to measure the "significance" of the average difference in values of these measures obtained for two iterations or two distinct search algorithms. The test results are expressed as the probability that the two sets of values obtained from two separate runs are actually drawn from samples which have the same characteristics. A small probability value thus indicates that the two curves are significantly different. If this probability for one measure is, for example, 5%, the difference in the two average values of that measure is said to be "significant at the 5% level".

Choice of a statistical method for calculating this probability is important. The present study uses three statistical tests, the familiar T-test, the Wilcoxon Signed-Rank Test (WSR), and the Wilcoxon Rank Sum Test (WRS) [16].

The T-Test and the Wilcoxon Signed-Rank Test are used in this report to compare the retrieval of one feedback iteration to another or of one algorithm to another, using all queries. The T-Test takes account of the magnitude of the differences, and assumes that the measures tested are normally distributed. The WSR test does not make this assumption. Moreover, the WSR test takes account only of the ranks of the differences, ignoring their magnitude. Because this test does not assume normality of the input and because it ignores some information (magnitudes of differences), the WSR test is more conservative than the T-Test. It is therefore less prone to the error of calling a result "significant" when it is not. Because information retrieval provides discrete rather than continuous data, and because only 42 data points (42 queries) are provided, the more conservative WSR test is preferable for the present evaluation.

The Wilcoxon Rank Sum Test can be used to test unpaired observations, and is used in Section VI-E of this study to compare one subgroup chosen from the 42 queries to a contrasting subgroup of queries. Like the WSR test, the WRS test ignores the magnitudes of the results and does not assume a normal distribution.

C. The Feedback Effect in Evaluation

The assignment of ranks to documents retrieved for feedback is a key factor in the evaluation of retrieval performance. Two methods of assigning these ranks have been proposed, and both are used in the present study. Hall and Weiderman [17] compare and evaluate these two methods. In previous feedback investigations, all documents in the collection received new ranks after each iteration and the top-ranked N documents were used for feedback. Hall and Weiderman point out that evaluation of this retrieval technique takes into account two effects, which they call "ranking effect" and "feedback effect".

Relevance feedback in effect uses information from one or more document descriptors to modify the query descriptor. The relevant documents used for this purpose will be ranked higher by the modified query than previously, and the non-relevant documents used will be ranked lower.

The effect of these rank changes in "retrieved" documents is termed the "ranking effect". If the ranking effect is included in an overall performance measure, the measured change in performance between feedback iterations is quite impressive.

This large change in "total performance" (including both ranking and feedback effect) indicates the extent to which the initial query has been perturbed toward the centroid of the relevant documents, and strongly supports Rocchio's theory.

Hall and Weiderman state that in an environment where the user must actively supply relevance judgments for feedback, changes in the ranks of documents which the user has already seen are of no interest to him. The user in such an environment is concerned primarily with the "feedback effect"; that is, the effectiveness of the modified query in bringing new relevant documents to his attention. They conclude that, though total performance is a valid measure of the effectiveness of relevance feedback in approaching tne "ideal query", the feedback effect should be isolated and examined as well.

The present study evaluates total performance and also measures feedback performance in the manner suggested by Hall and Weiderman, discarding the ranking effect and presenting only the feedback effect. The ranks of the top N documents retrieved in each iteration (the documents used for feedback) are "frozen" in all subsequent iterations, and only the remainder of the collection is searched using the modified query. Thus, in feedback effect evaluation, the N documents retrieved on any iteration are guaranteed to be N new documents; that is, documents not used for feedback on any previous iteration. Moreover, the performance measures for the first (second, third) iteration are calculated from a ranked document list in which the top N (2N, 3N) documents are the same as those retrieved previously. Only the changes in the ranks of documents not yet seen by the user is measured.

Feedback effect evaluation gives overall results that are deceptively low. Because the top ranks are frozen, no newly retrieved document can achieve a rank higher than that of any previously retrieved document. With a constant feedback strategy, therefore, on the first (second, third) iteration, the highest possible rank for a new document is N+1 (2N+1, 3N+1). For this reason, the feedback effect evaluation is a misleading measure of the overall performance of the retrieval system, and should be used in conjunction with other evaluation methods. Isolation of the feedback effect is primarily useful to compare different feedback strategies from the viewpoint of a user in an interactive retrieval environment. Figure 35 in Section VII-B compares total performance and feedback effect evaluation of similar feedback algorithms.

However, one feature of feedback effect evaluation is psychologically essential to a realistic relevance feedback system; the guarantee that the N documents retrieved on any iteration have not previously been seen by the user. For this reason, new evaluation methods that provide this guarantee without severely limiting the attainable retrieval performance should be investigated. Several such methods are discussed in Section VII-B.

The results reported in this study include:

Total Performance:

1. Normalized recall and precision.
2. Recall-precision curves with Quasi-Cleverdon interpolation.

Feedback Effect:

1. Normalized recall and precision.
2. Recall-precision curves with Neo-Cleverdon interpolation.
3. Document curves at several cut-off points.
4. T-tests and Wilcoxon Signed Rank tests of the normalized measures and of recall-precision curves.
5. Wilcoxon Rank Sum tests of normalized recall and precision.

Chapter VI

Experimental Results

The results of this study are presented in five general sections:

a) A comparison of the improvement in retrieval performance observed for the Cranfield 200 document collection with that obtained for the ADI 82 document collection used by Riddle, Horwitz, and Dietz [11].

b) An investigation of strategies that use only R', the set of relevant documents retrieved, to update the query. The different algorithms are obtained by varying the parameters π, ω, and α in the query-update formula. The "increasing alpha strategy" of Riddle, Horwitz, and Dietz is included among the methods tested.

c) An investigation of the effect of the number of documents given to the user for feedback on each iteration.

d) An investigation of strategies that use both relevant and non-relevant documents retrieved to update the query.

e) An investigation of the retrieval characteristics of selected subgroups of queries.

A. Comparison of the Cranfield and ADI Collections

The initial search results, before feedback, for the two collections are essentially the same except at the ends of the recall-precision curves. Below 30% recall, the precision of the ADI initial search is from 2 to 7% better than that of the Cranfield initial search. Above 80% recall the precision in the Cranfield initial search is from 2 to 6% better.

This result is interesting because there is reason to expect that performance in the Cranfield collection would be worse. Cleverdon and Keen point out that in a collection with a higher "generality number", that is, with a higher ratio of relevant documents to collection size, performance is better with respect to precision [18]. The average generality number of the ADI collection is over twice that of the Cranfield collection. The generality number in a collection of practical size would be even lower than that of the Cranfield collection.

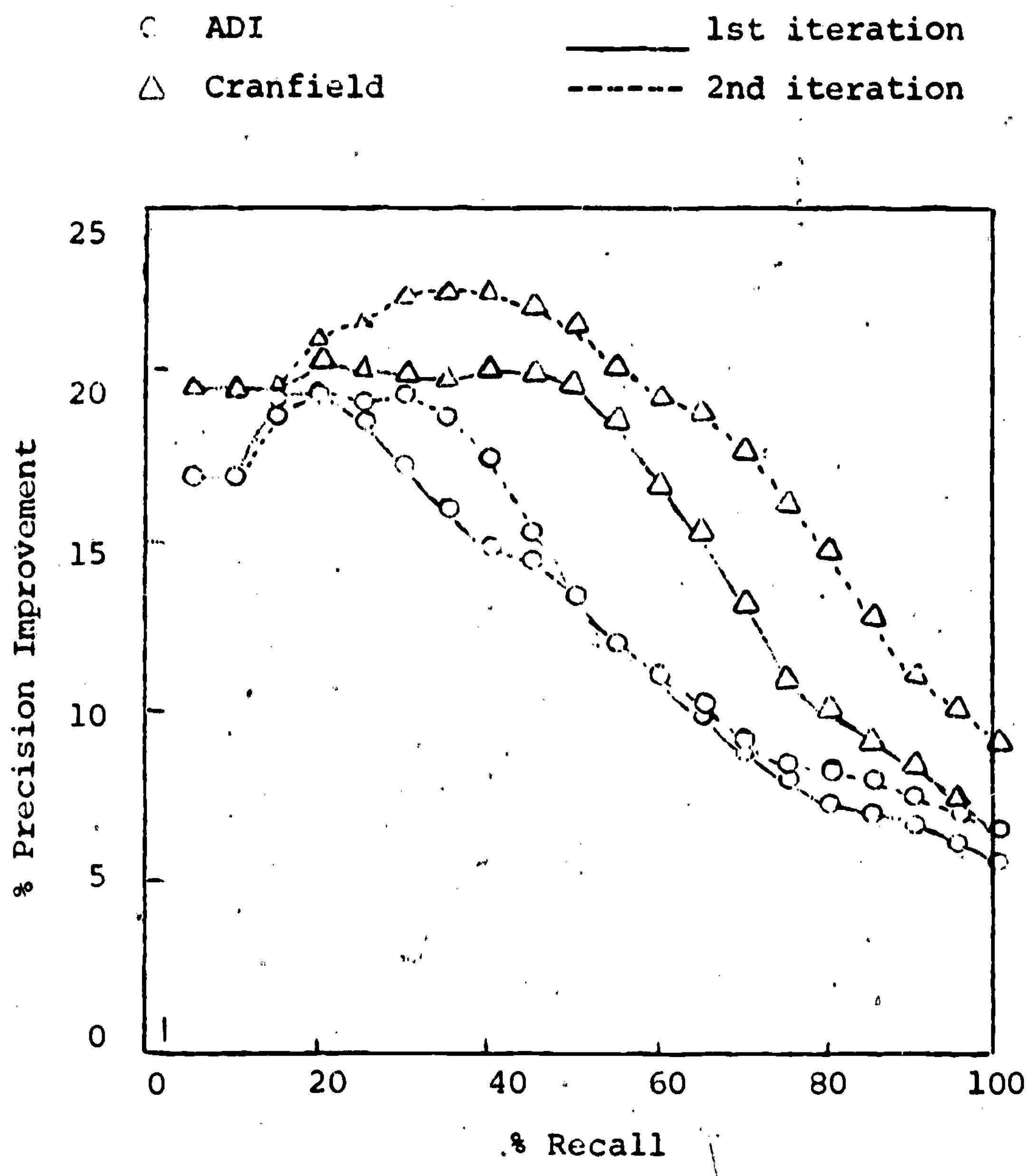
Because the initial search results differ, the total performance improvement caused by feedback in the recall-precision curve is used for comparison of the two collections. All thirty-five queries are used to search the ADI collection. The "increasing alpha strategy" of Riddle, Horwitz, and Dietz is the update formula, and five documents are given the user on each iteration.

Figure 3 shows the differences in total performance precision for all recall levels between the initial search and the first and second feedback iterations for each collection. In the Cranfield collection, relevance feedback causes greater improvement that in the ADI collection. Also, the second iteration results in a greater improvement over the first in the 200 document collection. The difference in generality between the collection would be expected to cause less improvement in the larger collection [18]. The greater effect of relevance feedback in the Cranfield collection could be due to any or all of the following factors:

a) The difference in subject and in the language of the subject. It is possible that the terminology of aerodynamics is 'harder', that is, more limited and precise, than the vocabulary of the newer field of computer science. Retrieval of documents from a harder subject area would

N=5

"increasing alpha strategy"



Improvement Over Initial Search
ADI and Cranfield Collections

Total Performance Recall-Precision Curves

Figure 3

expected to be better.

b) The difference in collection scope. The ADI collection covers a wider subject area within computer science than does the Cranfield collection within aerodynamics. A narrower subject area should provide better retrieval.

c) The difference in variability within the collections. The 200 documents were chosen from 1400 documents concerned with aerodynamics. The 82 document collection consisted of short papers presented at a single conference. Since the Cranfield 200 documents vary more in such parameters as vector length and terminology, relevant documents might be easier to distinguish from non-relevant documents.

d) The difference in query construction and relevance judgments mentioned in Section III. It is encouraging to find that in the more realistic Cranfield environment, relevance feedback causes more rather than less improvement in performance.

B. Strategies Using Relevant Documents Only

Two of the experiments of Riddle, Horwitz, and Dietz [11] are repeated for the Cranfield collection. To simulate their experiments with equation D of Section IV, the parameter α is varied; π is kept equal to 1, and μ and ω equal to 0. Both the "increasing alpha" and "constant alpha" strategies are employed.

Figure 4 clarifies the effect of the "increasing alpha strategy" and the "constant alpha strategy" for the first, second, and third iterations of a feedback run, evaluating total performance. The R^0 column shows the factors which multiply a relevant document retrieved on the initial search, R^1 shows the multipliers affecting a relevant document which is not retrieved until the first iteration, and R^2 shows the multipliers affecting a document not retrieved until the second iteration. Figure 4 assumes that a document once retrieved is retrieved on all succeeding iterations; in the experimental system this assumption is generally correct.

It is clear that both the constant and increasing alpha strategies give a document retrieved on an earlier iteration more significance in later queries. On the third iteration, the constant alpha strategy assigns to a document retrieved on the initial search three times the significance it gives a second iteration document (the respective multipliers are 3 and 1). The increasing alpha strategy assigns to an initial search document twice the significance of a second iteration document (the respective multipliers are 6 and 3). This effect stems from the use of the previous query Q_i as an element in the equation. To assign the same significance to relevant documents whenever they are retrieved, it is necessary to substitute Q_o for Q_i in the formula: that is, to let $\pi = 0$ and $\omega = 1$ in equation B. This is called the "Q_o strategy" in Figure 2.

Riddle, Horwitz, and Dietz [11] report that for the 82 document collection, the "increasing alpha strategy" performs somewhat better than the constant alpha strategy. In the Cranfield collection, the three strategies shown in Figure 2 give essentially the same results when $N = 5$. Using the Q_o strategy with different relative values of ω and α also does not change performance. Query update parameters (in equation D) for the six experiments performed are shown in Figure 5. Among all six experiments, the differences in normalized precision and recall are less than 0.75% for all iterations.

In total performance, six strategies using only relevant documents differ very little. Three additional 'relevant only' algorithms are compared using feedback effect evaluation. One of these strategies sets α and π equal to 1 in formula D. This strategy, called Feedback Increment, is not equivalent to the constant alpha strategy because the feedback effect evaluation provides new documents for feedback on each iteration. Figure 5 shows that the Feedback Increment strategy gives the same weighting effects as does the Q_o strategy. These two strategies are identical on the first iteration, but on subsequent iterations the feedback effect evaluation may retrieve different documents.

Another strategy using feedback effect evaluation, called Q_o+, gives added weight to the original query on each iteration by setting ω equal to 4 (α=1, π=1). A third strategy is Rocchio+, the Rocchio strategy without non-relevant documents. In effect, α equals $n_r n_s$ and π equals n_s, so α and π vary with each query.

Differences in feedback effect among these three methods are trivial. For the two overall measures and the recall-precision curves, the largest difference is 1.25 percent. The document curves are more sensitive in general to performance differences, especially in recall. The largest difference is 3% in recall at a 40 document cut-off. Most differences in all measures favor the Q_o+ strategy.

TOTAL PERFORMANCE	iter	Q_o	R^o	R^1	R^2
"increasing alpha strategy"	1	1	1	0	0
	2	1	3	2	0
	3	1	6	5	3
"constant alpha strategy"	1	1	1	0	0
	2	1	2	1	0
	3	1	3	2	1
"Q_o strategy"	1	1	1	0	0
	2	1	1	1	0
	3	1	1	1	1
FEEDBACK EFFECT					
"feedback increment"	1	1	1	0	0
	2	1	1	1	0
	3	1	1	1	1

Effects of 'Relevant Only' on the Multipliers
of Documents Retrieved on Three Successive Iterations

Figure 4

$N=5$, $n_a=N$, $n_b=0$, $\mu=0$

TOTAL PERFORMANCE	π	ω	α
	1	0	1
increasing alpha			2
			3
constant alpha	1	0	1
Q_o strategy	0	1	1
Q_o weighting query double	0	2	1
Q_o weighting query half	0	1	2
Q_o weighting query six times	0	6	1
FEEDBACK EFFECT			
Feedback increment	1	0	1
Feedback Q_o+	1	4	1
Feedback Rocchio +	$n_r n_s$	0	n_s

Query Update Parameters for Relevant Only Strategies
Using Only Relevant Documents

Figure 5

The 200 document collection seems quite insensitive to variations in the parameters π, ω, and α. The considerations mentioned in section VI-A are probably relevant here also. This insensitivity indicates that perhaps the performance for the Cranfield collection is more stable in general than for the ADI collection. Evidence of comparative stability is also reported by Lesk and Salton [19]. The performance differences between automatic use of the word stem thesaurus and a regular subject-area thesaurus (see Section II) are less pronounced in the Cranfield 200 collection than in the ADI collection.

It is evident from the reported experiments that the weight assigned to the original query has little effect on retrieval. This finding tends to support the conclusion of Crawford and Melzer [12] that the original query is not needed after the initial search (Section III). The advantage of their strategy over equation B is probably not caused by the omission of the initial query when relevant documents are found, but by the non-relevant document feedback used when no relevant documents are found (see Section VI-D).

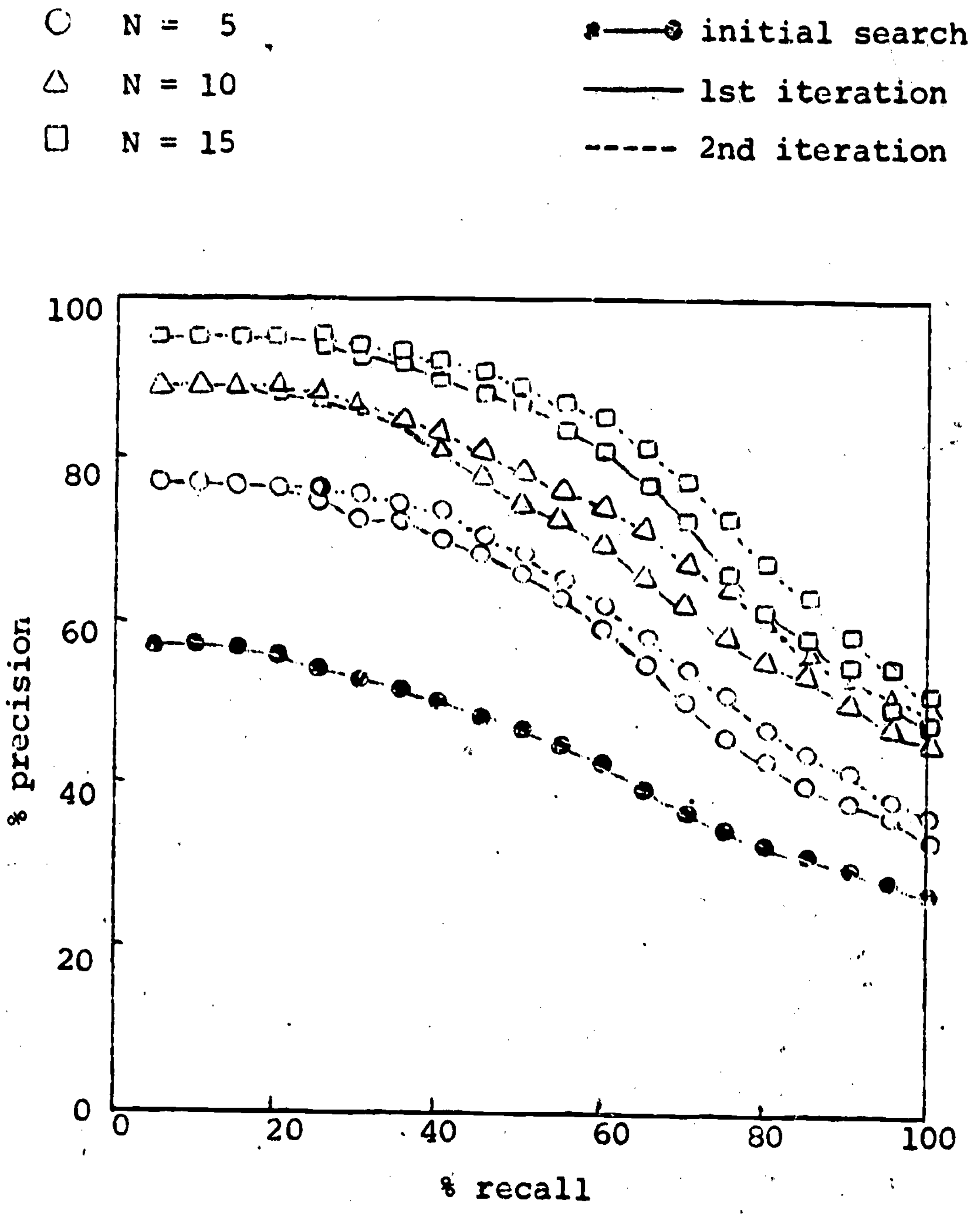
C. Amount of Feedback Output

The number of documents fed to the user is a critical parameter in a relevance feedback system. Of course, performance improves when the user supplies more information. This improvement must be evaluated in terms of the extra effort required of the user.

Figure 6 shows the total performance of the "increasing alpha strategy" when 5, 10, and 15 documents are fed to the user for relevance judgments. The total performance improvement between the N = 5 and N = 10 curves might justify doubling the number of relevance judgments the user must make; that is, a hypothetical "average" user might be willing to double his effort to achieve such an improvement. Tripling the feedback to produce the N = 15 curve might not be justified by total performance, especially at the high recall end of the curve.

Caution is necessary in interpreting the feedback effect evaluation when N is varied, because the feedback effect evaluation gives an unfair advantage to runs using few documents for feedback. When five documents are used for feedback, ranks 1-5 are frozen on the first iteration and ranks 1-10 on the second (see Section V-C). When ten documents are fed back, however, ranks 1-10 are frozen on the first iteration and ranks 1-20 on the second. The difference in results caused by increasing the number of documents fed back is therefore minimized by the feedback effect process.

Recall-precision curves for the feedback effect are not presented in this section, because the minimizing effect described above is averaged into different recall levels for different queries. For example, assume that query a has four relevant documents and query b has two. Each query retrieves the first relevant document with rank 8. When five documents are used for feedback, each retrieves the first relevant document with rank 6. When ten documents are used, of course rank 8 is 'frozen' by the feedback effect evaluation. Consider the effect of these queries on the



Number of queries (out of 42)
retrieving no relevant in the first N:

N = 5	N = 10	N = 15
11	5	2

Varying the Number of Feedback Documents

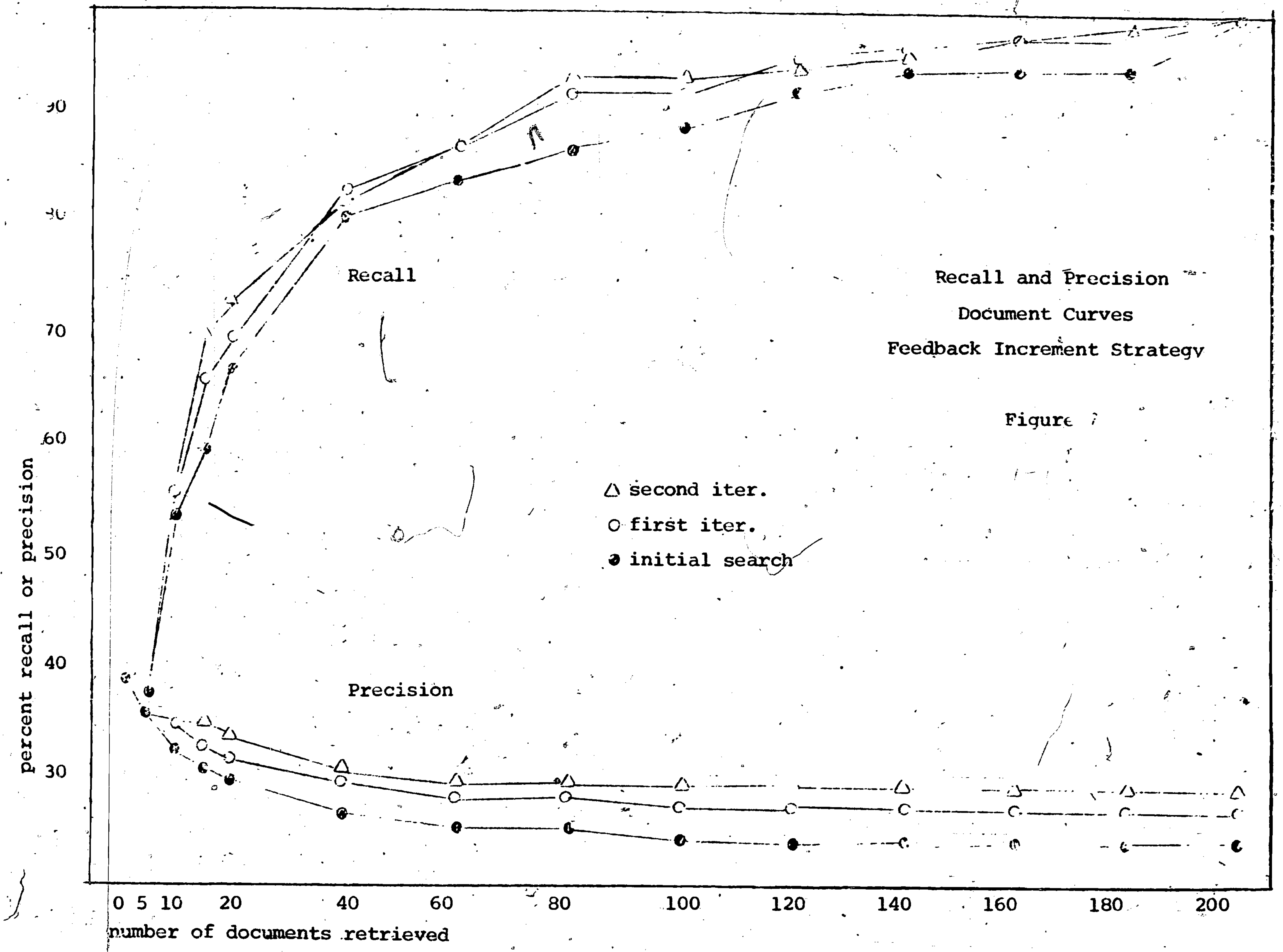
Total Performance Recall-Precision Curves

Figure 6

recall-precision averages for the first iteration. When ten documents are used for feedback, neither query improves these averages. When five are used, query a improves the recall-precision averages from 5% to 25% recall, but query b improves the recall-precision averages from 5% to 50% recall.

Thus it is hard to judge the significance of any difference in results caused by differences in feedback output. Figure 7 shows the document curves for two iterations of feedback with N equal to 5, using the Feedback Increment algorithm. Figure 8 uses these curves as a norm for comparison with the results of less feedback and of more feedback. The document curves of Figure 7 are represented in Figure 8 by the straight line at zero difference. The differences between N=10 and N=5 for two iterations and between N=2 and N=5 for one iteration are graphed. The N=2 differences, indicated by '△', are positive at first because ranks 3-5 are frozen for the N=5 curve. This initial advantage fades after 10 documents and the N=2 results are lower than the N=5 norm thereafter. The N=10 curves for both iterations are affected by the feedback effect evaluation. The first iteration gains higher performance than N=5 after 13 documents. The second iteration curves cross the N=5 norm after 15 documents, even with ranks 1-20 frozen. After 40 documents have been retrieved, the differences in both recall and precision for the first iteration are slight; the N=10 advantage on the second iteration is slight but consistent.

After 20% of the collection has been searched, the differences in

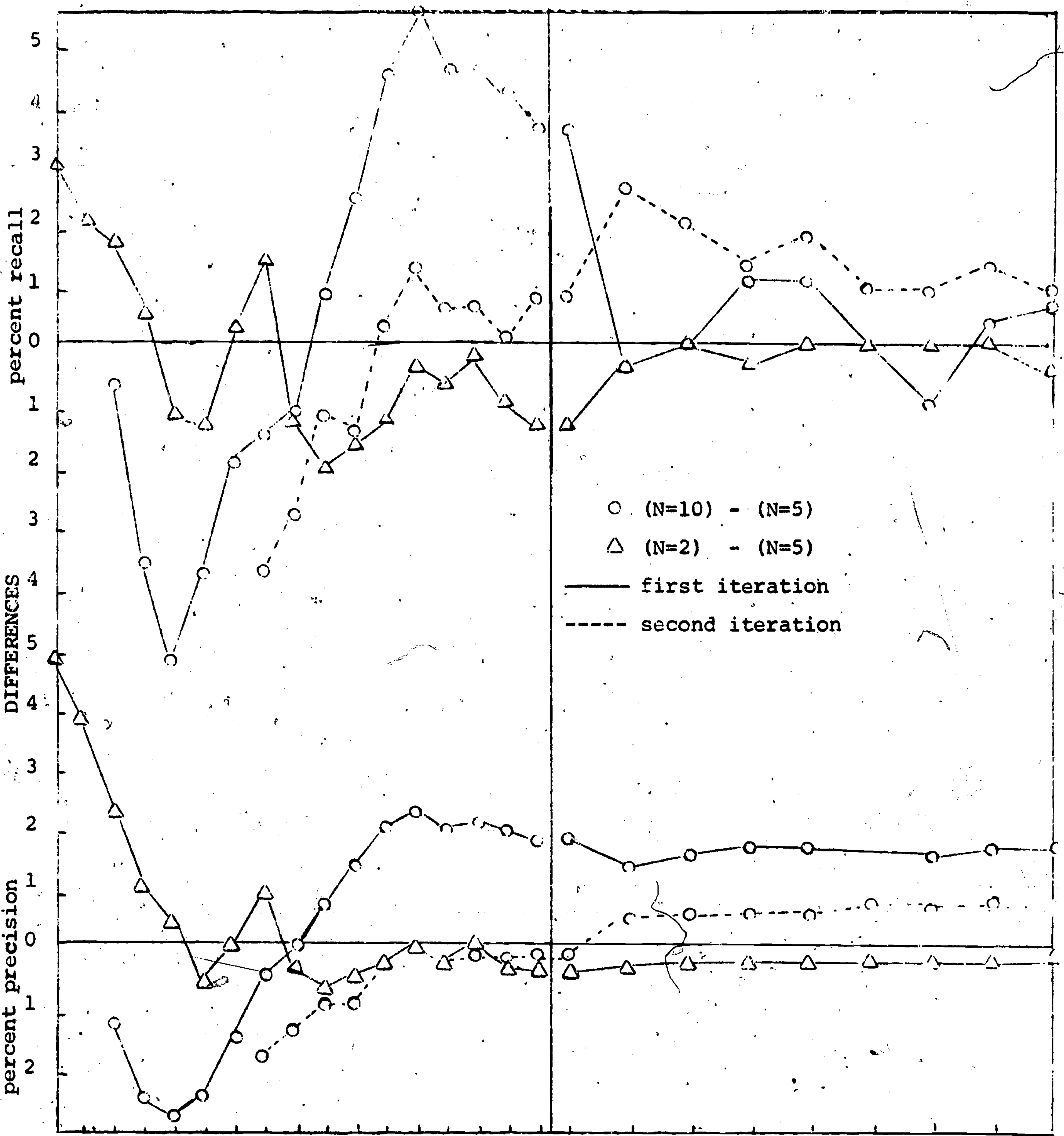


Recall and Precision

Document Curves

Feedback Increment Strategy

Figure 7



Feedback Effect Document Curves

Recall and Precision Differences

Varying the Number of Documents Retrieved
(N=5 normal, N=2 and N=10 compared)

Figure 8

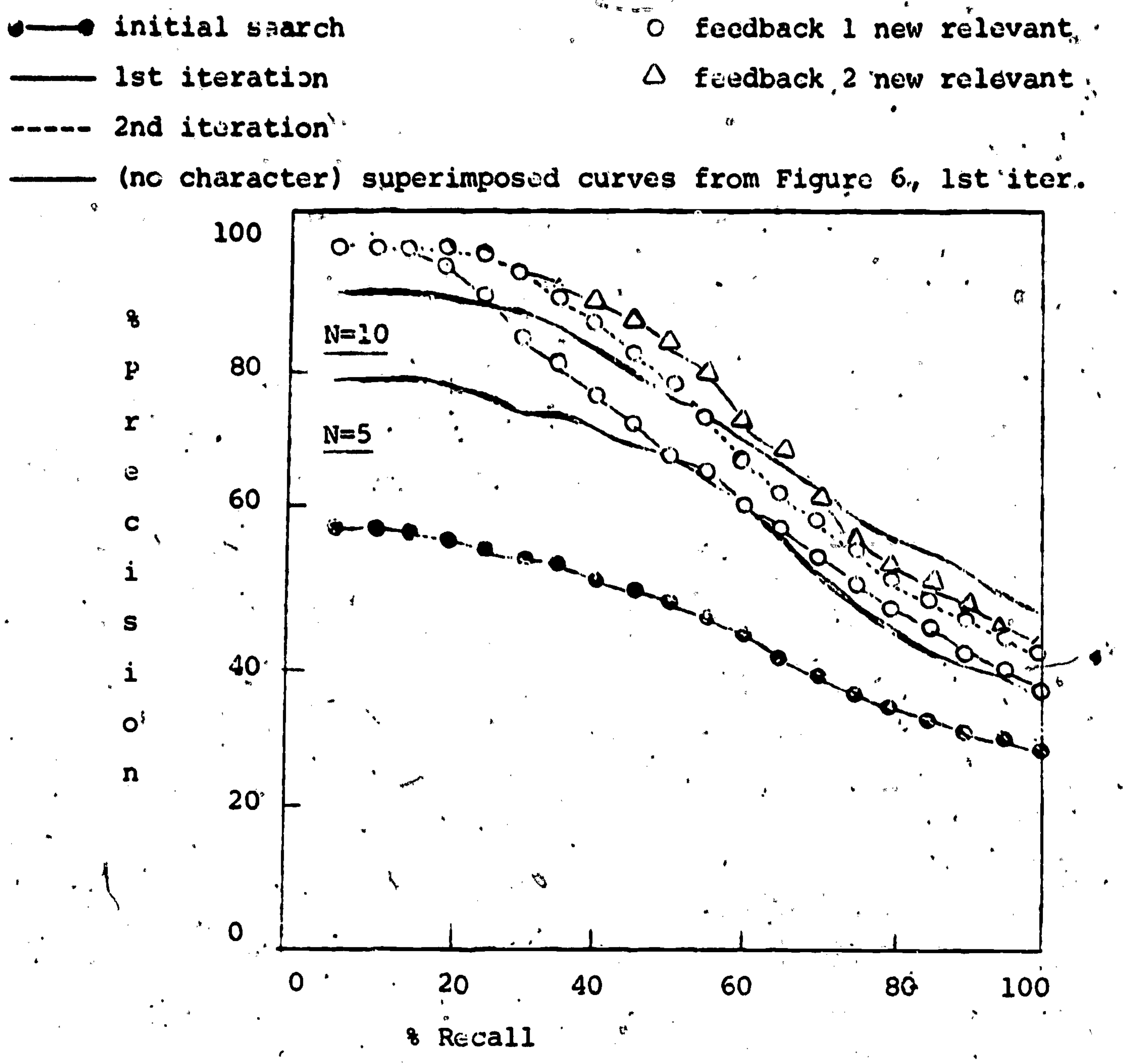
feedback effect observed in Figure 8 are quite low. However, the marked improvement in early retrieval caused by additional feedback might justify the additional user effort and system output required, especially if further feedback iterations are desired. Moreover, it is important to note that certain users get no benefit from any feedback strategy using only relevant documents. These are the users who find no relevant in the first N documents retrieved. For N=5, 10, and 15, the number of queries retrieving no relevant on the initial search is given in Figure 6, in the table below the graph. This table probably explains much of the performance difference among the three strategies. Eleven queries in the N=5 case produce the same low performance on the initial search, first iteration, and second iteration. These low results are averaged into all the N=5 curves. The N=10 curve is pulled down by only five such queries, and the N=15 curve by only two. In the N=5 case, one quarter of the users are not assisted by the chosen feedback strategy, a large proportion for a practical retrieval system. For these unlucky users, feedback of more documents is worth the effort.

A variable feedback strategy is here proposed which might save effort to the average user and give better service for more effort to the user who does not find a relevant document early in the initial search. Each user is fed retrieved documents until he finds one relevant document that he hasn't seen on any previous iteration. The relevant document found is immediately used to produce a new query. The success of this strategy depends on the ability of a single relevant document to improve the retrieval performance.

Figure 9 shows the total performance results of two iterations where the "user" is instructed to search the retrieved documents until he finds one new relevant document or until he has seen 15 documents. The "△" curve in Figure 9 shows what happens when the user is instructed to find two new relevant documents. Only one iteration of the latter scheme was run because several queries do not have four or more relevant documents.

The first iteration feeding back one relevant document begins near the N=15 curve of Figure 6 but by 50% recall has dropped near the first iteration N=5 curve, which has been superimposed on Figure 9. The table below the graph shows that the "average" user had to scan only four documents for feedback in order to achieve the performance displayed in the first iteration "o" curve. By contrast, each user looked at exactly 5 documents to produce the first iteration N=5 curve. The first iteration strategy of feeding back only one relevant document gives equal or better performance for less average effort.

The second iteration "o" curve requires the average user to search 5.9 more documents, or a total of 9.9 documents. This curve drops below the first iteration N=10 curve (10 documents scanned) at roughly 55% recall (the first iteration N=10 curve from Figure 6 is superimposed on Figure 9). The user desiring high precision and who may be less interested in high recall might be wise to feed back one relevant document for each of two iterations. However, the user needing higher recall should instead look at ten documents retrieved on the initial search. (These statements



	1 relevant		2 relevant	combined strategy
	Initial output	1st iter output	initial output	Initial output
Avg. no. of documents searched	4.0	5.9	7.0	6.4
No. of users (of 42) not finding n new relevant in the first 15 documents retrieved	2	11	9	2

Variable Feedback

Feeding Back Only the First or First Two New Relevant Documents Found Within the First Fifteen Documents Retrieved, Total Performance, Q_o Strategy.

Figure 9

apply to the "average" user). It is also seen from the table below the graph that for the second iteration "o" curve one quarter of the users cannot find a new relevant document, and thus after the first iteration these users search 15 documents to no avail. Such performance would be quite annoying in practice.

The average user who searches for two relevant documents in the initial output looks at 7 documents. His recall-precision ("△") drops below the first iteration N=10 curve at 65% recall. Although 9 out of 42 users do not find two relevant documents in the first 15, all but two of them find one relevant to feed back to the system.

The user who feeds back two relevant documents on one iteration ("△") achieves better performance than does the user who feeds back one relevant document on each of two iterations (second iteration "o"). This result shows that the second relevant document retrieved on the initial search is more valuable for feedback than the first new relevant retrieved on the first iteration by the total performance retrieval method. Feeding back one relevant document on the first iteration evidently pushes down some relevant documents that are valuable for retrieval. This finding provides a strong argument against the proposed variable feedback strategy, at least where high recall is desired. Perhaps some sort of combination strategy might be optimal; for instance, the user could be instructed to feed back all relevant documents in the first five retrieved, but if none are found in the first five to keep looking and feed back the first relevant document found.

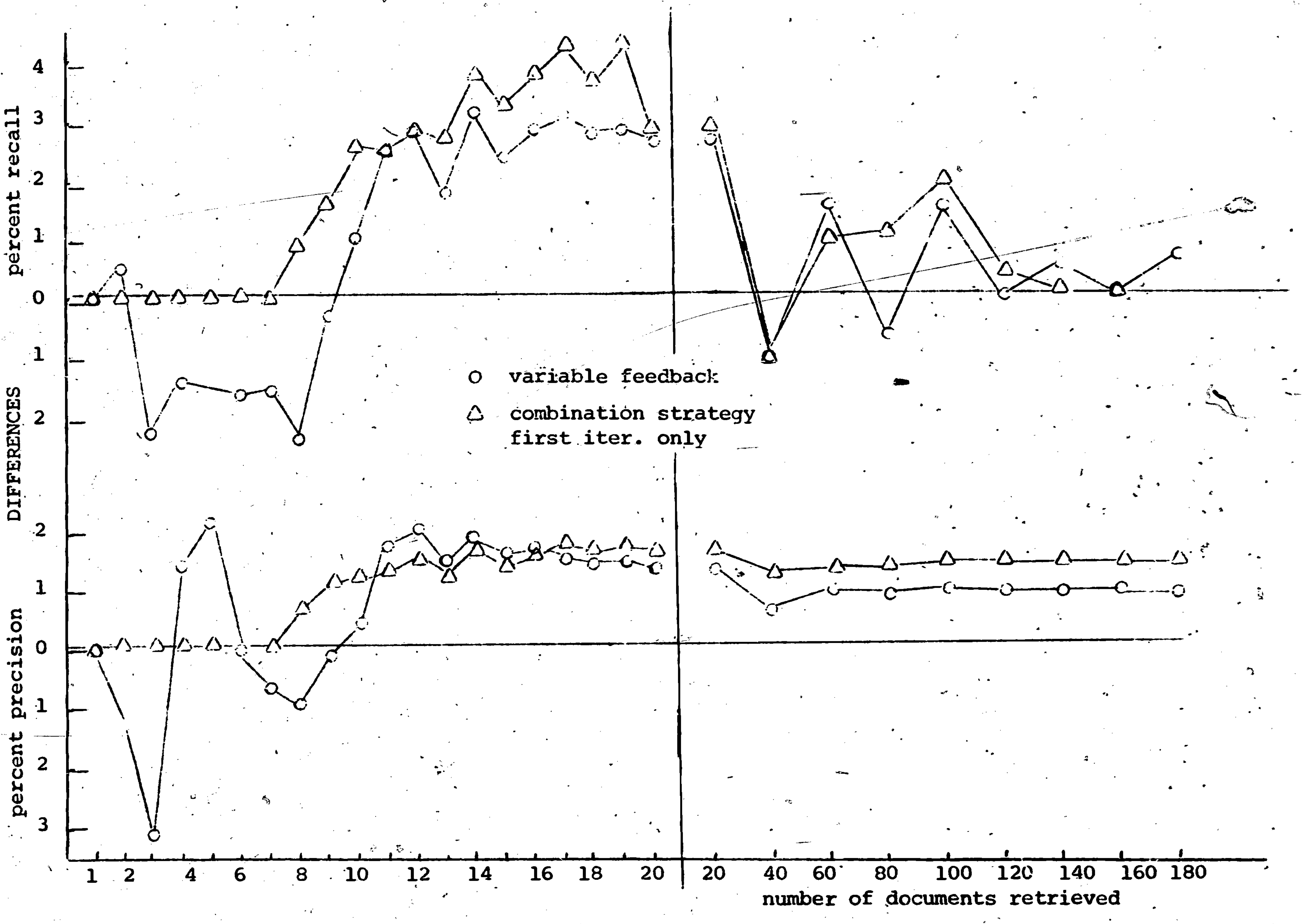
One iteration of feedback effect performance of variable feedback and of the combined strategy described above is presented in Figure 10. The differences between variable feedback (feeding back one relevant) and constant N=5 feedback are graphed (o). After nine documents have been retrieved the feedback advantage using the first relevant document for feedback is evident. The combination strategy (graphed △) that retrieves at least five documents shows a greater improvement over N+5 than does variable feedback. This result shows that this variable feedback algorithm is advantageous only for those queries that retrieve no relevant among the first 5 documents. For those that do retrieve relevant within the first five documents, the constant N=5 strategy gives better feedback effect results. Figure 9 shows that the combination strategy requires the average user to look at 6.4 documents, as compared to the 4.0 documents retrieved by variable feedback.

Total performance and feedback effect results support three conclusions about feedback strategies that use only relevant documents:

1) Retrieving more documents does improve both types of performance (except where the rank-freezing of feedback effect evaluation prevents any improvement). Further, notable improvement can be obtained by searching further if no relevant documents are retrieved in the first N.

2) The total performance for 5, 10, and 15 documents indicates that when N is constant for all queries, the average increment of improvement obtained tends to become smaller as more documents are used for feedback.

3) However, a comparison of the variable feedback and com-



Variable Feedback and Combination Strategy Compared to N=5 Norm Recall and Precision Differences, Feedback Effect Document Curves

Figure 10

bination strategies show that the first relevant document re-retrieved generally does not contain enough information for retrieval, and that documents retrieved soon after the first (N=5) can add useful information. In their study cited in Section III, Crawford and Melzer used only one 'very relevant' document for retrieval. The finding of this study would indicate that if several documents are almost equally relevant, all of them should be used.

These three conclusions lead to a recommendation that N be set to some value that most users would consider reasonable, but that for some queries N should be raised until at least one relevant document is retrieved. This recommendation endorses the "combination strategy" for a retrieval system using only relevant documents.

D. Strategies Using Non-Relevant Documents

Rocchio's update formula (equation A) considers the information contained in the set of non-relevant documents retrieved (S) to be as important as that contained in the set of relevant documents retrieved (R). If this is the case, the strategies so far examined disregard half the available feedback information. Further, information from non-relevant documents retrieved on the initial search might help those users who retrieve no relevant documents on the initial search (see Section III and reference 6). Figure 6 shows that there are eleven such users out of 42 when N equals 5.

However, problems arise in using the non-relevant documents in the SMART experimental system. There is no provision for negative weights in the query vector. Also, queries and documents cannot be normalized to the same length for query updating. There is some danger, therefore, that the query will be reduced to nearly the zero vector when the documents of S are subtracted from it. Riddle, Horwitz, and Dietz [11] try to avoid this danger in their "negative heuristic strategy" by feeding back only the first two non-relevant documents retrieved.

The Rocchio strategy adjusts the multipliers for each query so as to weight the original query, the sum of the relevant, and the sum of the non-relevant equally, and uses all retrieved documents.

This study compares the Rocchio and 'negative heuristic' strategies using total performance and feedback effect measure. All comparisons are made with N equal to 5. Figure 11 compares the Q_o strategy (see Section VI-B) with a strategy called 'Dec Hi', that decrements each query by subtracting from it the first retrieved non-relevant document. In the query update formula (equation D), the parameter values and effective update formulas for these strategies are:

Q_o: $\pi=0,\ \omega=1,\ \alpha=1,\ \mu=0,\ n_a=N,\ n_b=0$

$$Q_{i+1} = Q_i + \sum_{1}^{N} r_i$$

Dec Hi: $\pi=0,\ \omega=1,\ \alpha=1,\ \mu=-1,\ n_a=N,\ n_b=1$

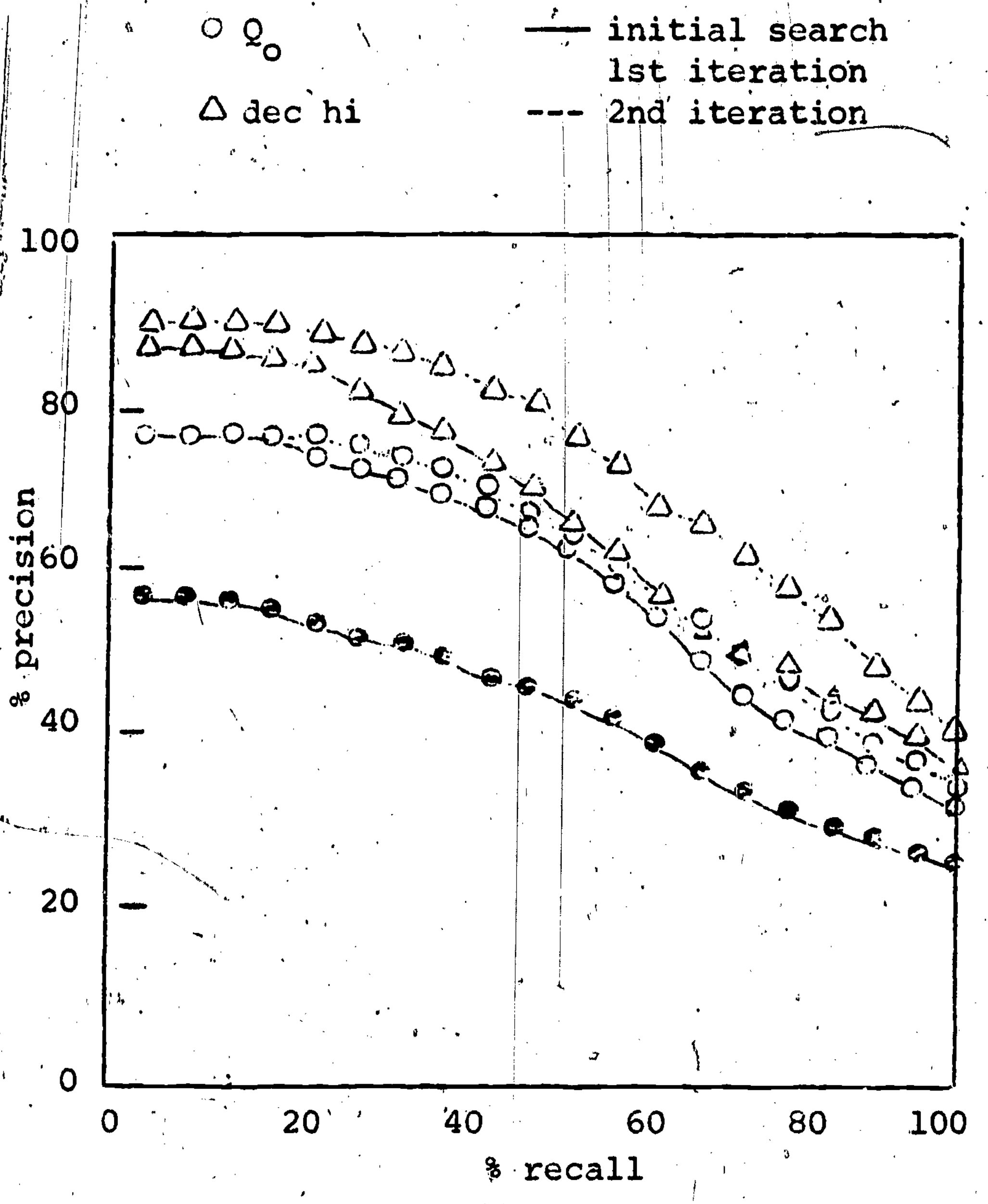
$$Q_{i+1} = Q_i + \sum_{1}^{N} r_i - s_1$$

Figure 11 shows that the average results are consistently better for the "dec hi" strategy, especially on the second iteration.

For this experiment, the implications of the total performance normalized precision and recall results given in Figure 12 seem inconsistent with those of the recall-precision curves of Figure 11. On the first iteration, the recall-precision curve for the dec hi strategy is above that for Q_o at all recall levels. However, the normalized recall for the first iteration is lower for dec hi, although precision is one percent higher. (On the second iteration, the normalized recalls are the same, and the normalized precision for dec hi is five percent higher). This apparent paradox can be understood by considering the normalized recall measure.

Each document retrieved is assigned a "rank" in order of retrieval (rank 1 is the document retrieved first). The normalized recall measure is based on the sum of the ranks of all relevant documents in the search. A change in rank affects this measure equally regardless of the magnitude of the rank. That is, a change from rank 195 to rank 191 is equivalent to a change from 5 to 1 in its effect on normalized recall. The same is not true for normalized precision. It seems evident that while the dec hi strategy increased the rank (1 is considered highest) of some of the relevant documents, it decreases the ranks of others that are, on the average, of lower rank already. This explains the phenomenon of higher precision at all levels of recall but lower overall normalized recall.

Figure 13 shows how much the dec hi strategy helps the 11 users who receive no relevant documents in the first 5 on the initial search. For the "inc only" strategy, the initial search and all subsequent iter-



Decrementing the Highest Non-relevant Document

Total Performance Recall-Precision Curves

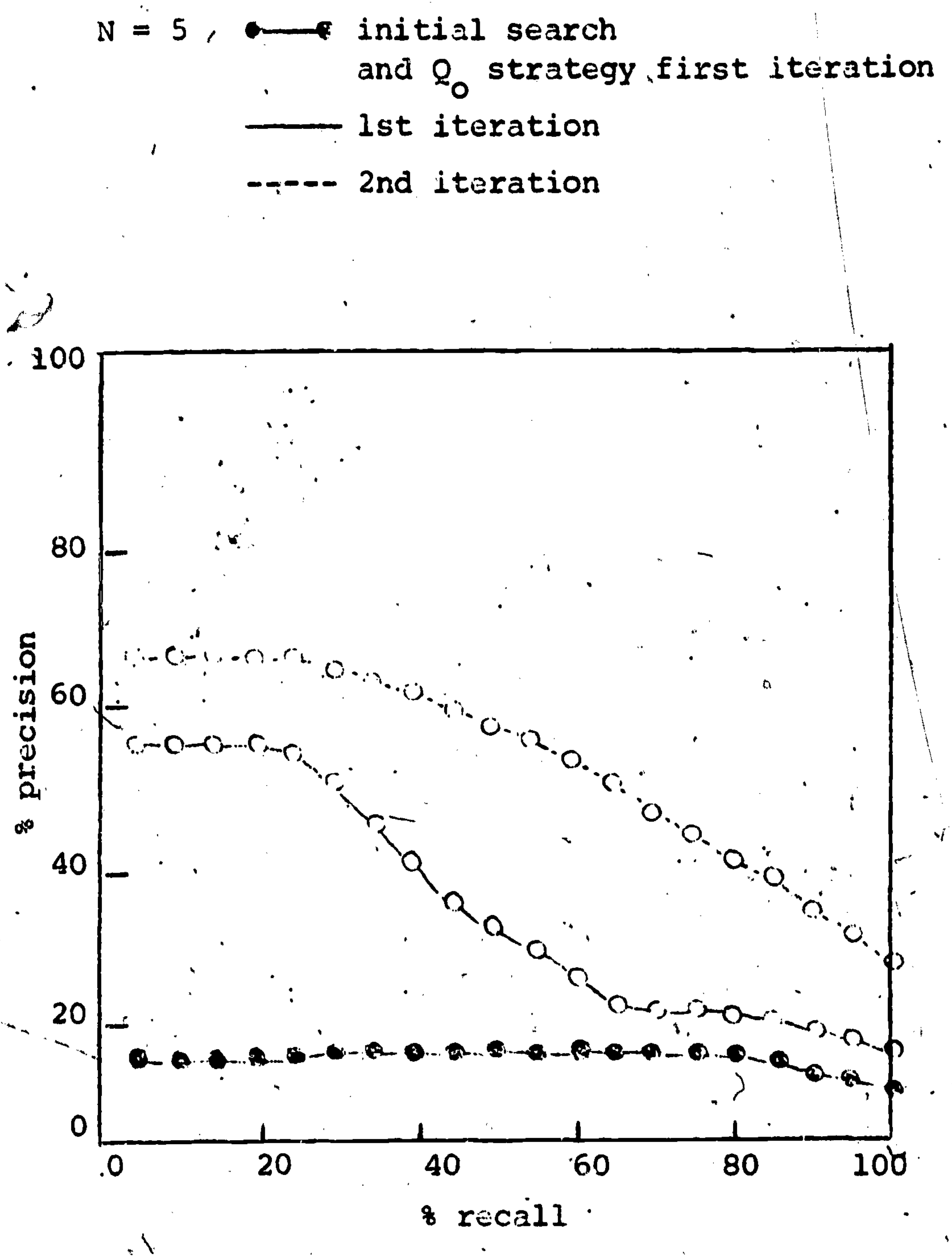
Figure 11

		Init	1	2
Normalized Recall	Q_o	88%	90	90
	Dec High	88	87	90
	Dec 2 Hi	88	85	89
Normalized Precision	Q_o	68	75	76
	Dec High	68	76	81
	Dec 2 Hi	68	75	80

Normalized Results for Non-relevant Document Strategies

Total Performance

Figure 12



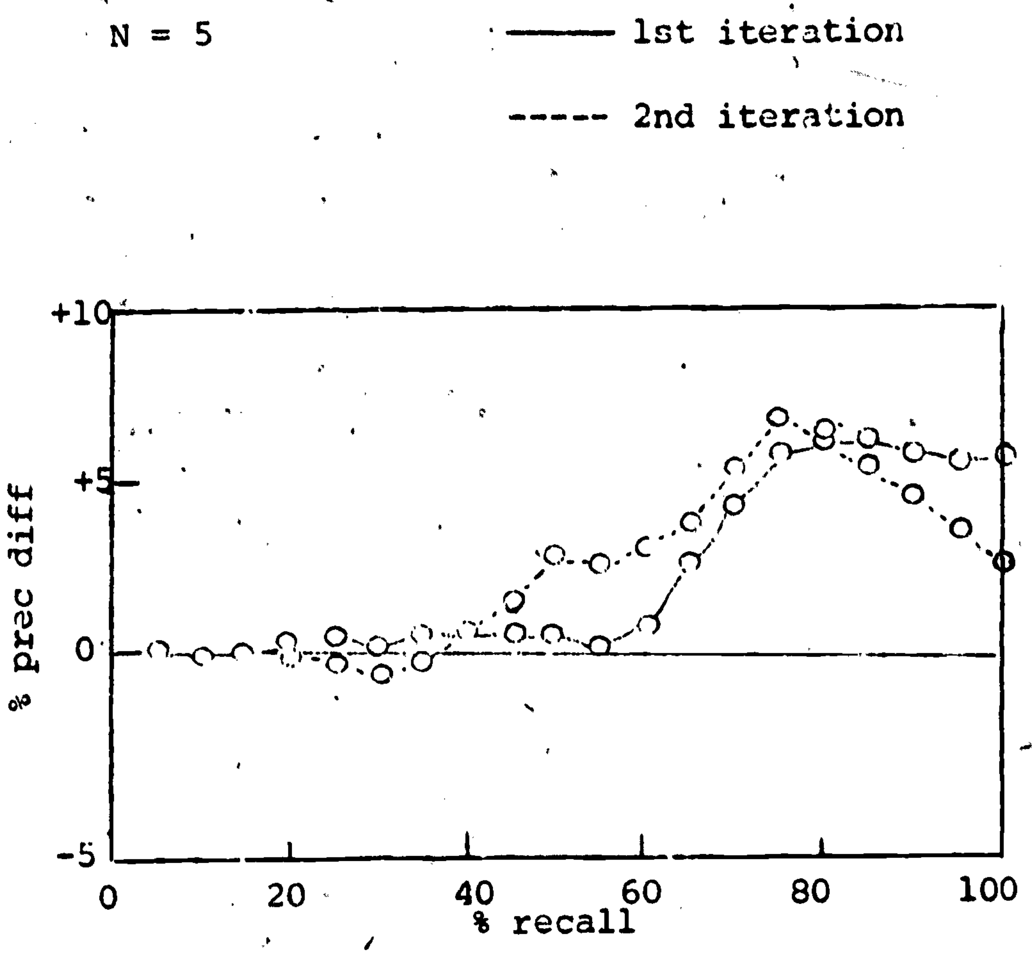
Decrementing the Highest Non-Relevant Document on Eleven Bad Queries Total Performance Recall-Precision Curves

Figure 13

ations are the same for these 11 users; the precision being about 10 percent. Feeding back one non-relevant document fetches at least one relevant document on the first iteration for 7 of these 11 users. For some of these queries, some low ranking relevant documents are pushed still lower at first. The relevant documents which are raised to the first 5, however, provide a second iteration query which often raises these same low documents again. The first iteration curve thus shows the most improvement at low recall, while the second iteration shows great improvement all along the recall-precision curve.

Since the improvement in total performance for the 11 "bad" queries is so striking, it is natural to wonder whether this strategy is helping or hurting the other 31 users. Figure 14 shows a different curve for the dec hi and Q_o strategies run only on the 31 queries that retrieve at least one relevant in the first 5 documents. A point above the zero line indicates that dec hi is better than Q_o at that recall. Both iterations are better for dec hi, especially at the high recall end of the curve, where they differ by as much as six percent. Since the dec hi strategy improves the results even for the "good" queries, a heuristic strategy that selects only some of the queries (as does the "negative heuristic strategy" of Riddle, Horwitz and Dietz) for the dec hi algorithm appears unnecessary in this environment.

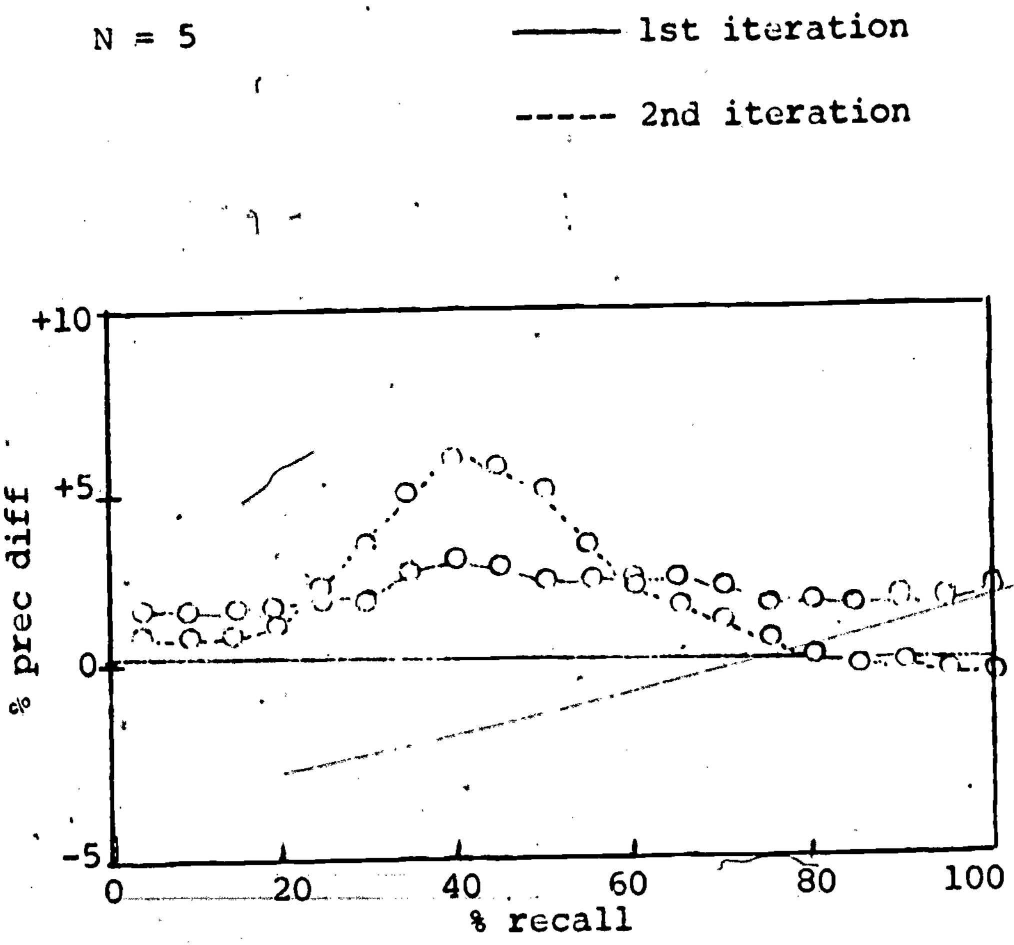
Figure 15 represents a total performance difference curve comparing the "dec hi" strategy with the alternative of decrementing the query by subtracting the two highest non-relevant documents retrieved on each iter-



(Dec Hi) - (Q_o)

For the 31 Good Queries, a Comparison Between Decrementing the Highest Non-Relevant and Incrementing the Relevant Only Total Performance Difference

Figure 14



(Dec Hi) - (Dec 2 Hi)

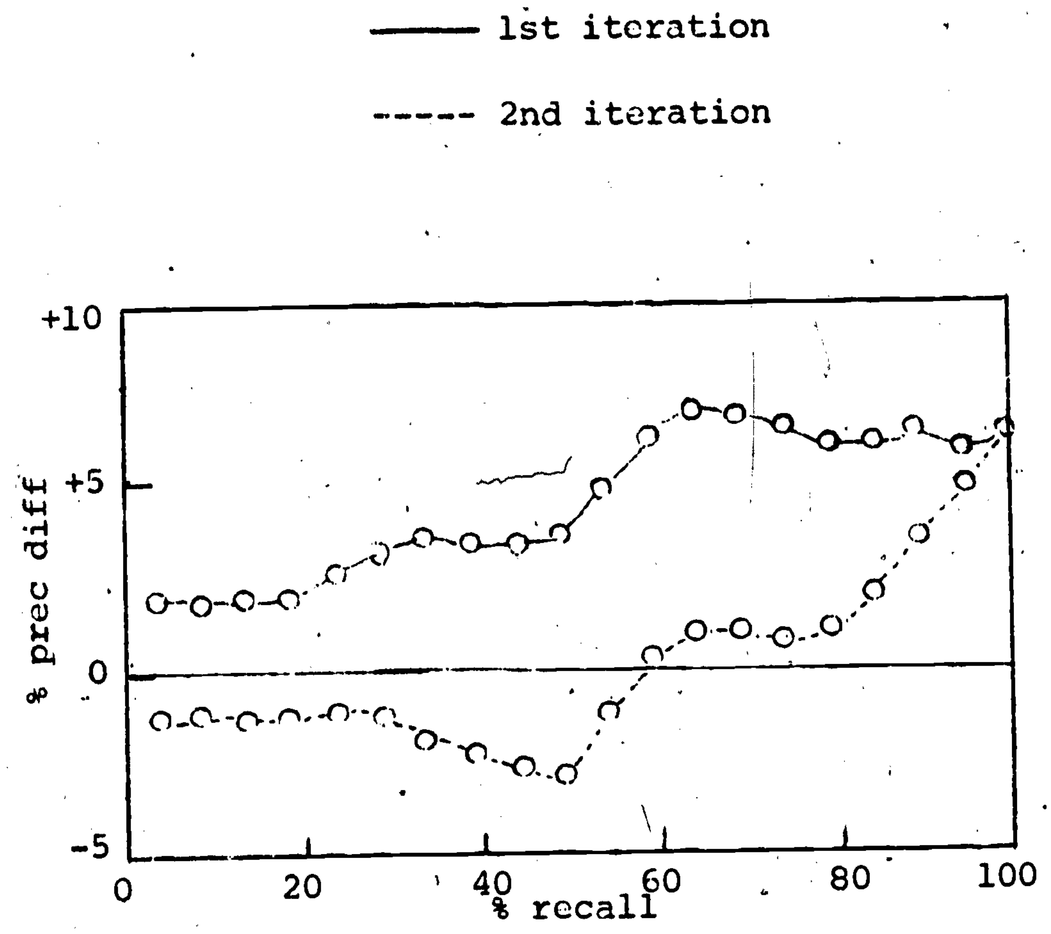
A Comparison of Decrementing the Highest or the Two Highest Non-Relevant Documents on Each Iteration

Total Performance Difference

Figure 15

ation (called "dec 2 hi"). It shows that decrementing only one non-relevant gives generally better results; the largest difference being a five percent hump at 40% recall in the second iteration. In Figure 12 the normalized measures show the same relationship; the dec 2 hi strategy is one or two percent lower on each iteration than is dec hi. This result may be due to the danger mentioned earlier, that the non-relevant documents may be subtracting out most of the query. (Only one query completely disappears using this strategy, and it is erased also by the dec hi and Rocchio's algorithms). It might be possible to overcome this "disappearing query" phenomenon by juggling the parameters π, ω, α, and μ, without introducing the complications of Rocchio's normalizing method.

It was mentioned in Section VI-C that much of the improvement between the N=10 and the N=5 curves of Figure 6 might be caused by the improvement on the six queries that fetch a relevant document within the first 10 but not the first 5 on the initial search. Seven users of the unlucky 11 are helped by the dec hi strategy; that is, the dec hi strategy provides useful feedback for one more user than does the relevant document strategy with N=10. It is pertinent to ask if the dec hi algorithm has, in fact, attained the total performance of the N=10 curve. Figure 16 shows a difference curve between the N=10 curve of Figure 6, and the dec hi curve of Figure 11. The N=10 curve is higher for the first iteration, over five percent higher at the high recall end of the curve. This is understandable in view of the lowering of low-ranking relevant documents on the first iteration, discussed earlier in this section. For the second iteration,



(Feedback 10) - (Dec Hi)

A Comparison Between Feeding Back 10 Documents

but Increasing Relevant Only,

and Feeding Back 5 Documents

but Decrementing the Highest Non-Relevant

Total Performance Difference

Figure 16

the dec hi curve is slightly better at the low recall end and only slightly worse at recalls between sixty and ninety percent. Considering that the dec hi curve only requires half as much user effort (5 documents scanned instead of 10), the total performance results strongly favor this non-relevant document retrieval strategy.

The feedback effect results are not as encouraging. Using the two overall normalized measures and the recall-precision curves, three strategies are compared and their differences tested with the two significance tests described in Section V-A, the T-Test and the WSR Test. The three strategies that are compared in feedback effect performance are:

1) Feedback Dec Hi: The Dec Hi strategy with the feedback effect retrieval method, using the first retrieved non-relevant document.

2) Rocchio: Rocchio's recommended strategy (without normalized vectors) using all retrieved documents.

3) Q_o+: The strategy described in Section VII-B, that gives added weight to the original query and uses only relevant documents.

The differences in feedback effect recall-precision curves among these strategies are not significant. The largest differences found were 1.3% in the first iteration, significant at the 30% level, and 2.0% in the second iteration, significant at the 11% level. The largest difference between Q_o+ and any other strategy was 1.2%, significant at the 64% level. These significance figures were obtained using the less conservative T-test. Figure 17 shows the differences among the three strategies in normalized recall and normalized precision. The feedback effect results agree with the total performance results in showing a drop in normalized precision for the two non-relevant document strategies on the first iteration. The five percent difference between Q_o+ and feedback Dec Hi is significant at the 6% level, and the six percent difference between Q_o+ and Rocchio is significant at the 3% level, according to the T-test. However, the Wilcoxon Signed-Rank Test (WSR) indicates that the two algorithms do not give significantly different results. The significance level comparing Q_o+ and feedback Dec Hi is 46%, and that comparing Q_o+ and Rocchio is 95%.

These different significance levels must be considered in the light of the characteristics of the two significance tests. The T-test takes account of magnitude, the WSR test considers only rank. Evidently, differences favoring Q_o+ and differences favoring the non-relevant document strategy are mixed in rank, producing insignificant results on the WSR test. Yet, some of the results favoring Q_o+ (not <u>all</u>, because the ranks are mixed) must be very large in magnitude, to give significant indications on the T-test. Thus, for some queries, the Rocchio and Feedback Dec Hi algorithms must be <u>much</u> less effective than Q_o+ as measured by normalized recall, while remaining effective as measured by normalized precision.

The normalized recall obtained by feedback effect evaluation shows the same behavior as the total performance normalized recall on the first iteration. Both evaluation methods lead to the conclusion that the use of non-relevant documents for feedback apparently raises the ranks of fairly high-ranking relevant documents, and at the same time lowers the ranks of some low-ranking relevant documents on the first iteration.

The significance levels obtained by comparing the first and second iteration results to the initial search result within the strategies are very informative. Figures 18 and 19 show the performance of algorithms Q_o+ and Rocchio respectively. The significance of the gap between the initial search and each iteration is tested, using the more conservative WSR test.

Looking at the three recall-precision graphs, the average performance of the three algorithms seem quite similar. In fact, the differences in average performance are not significant. Yet, the significance levels displayed in Figure 18 differ greatly from those displayed in Figure 19.

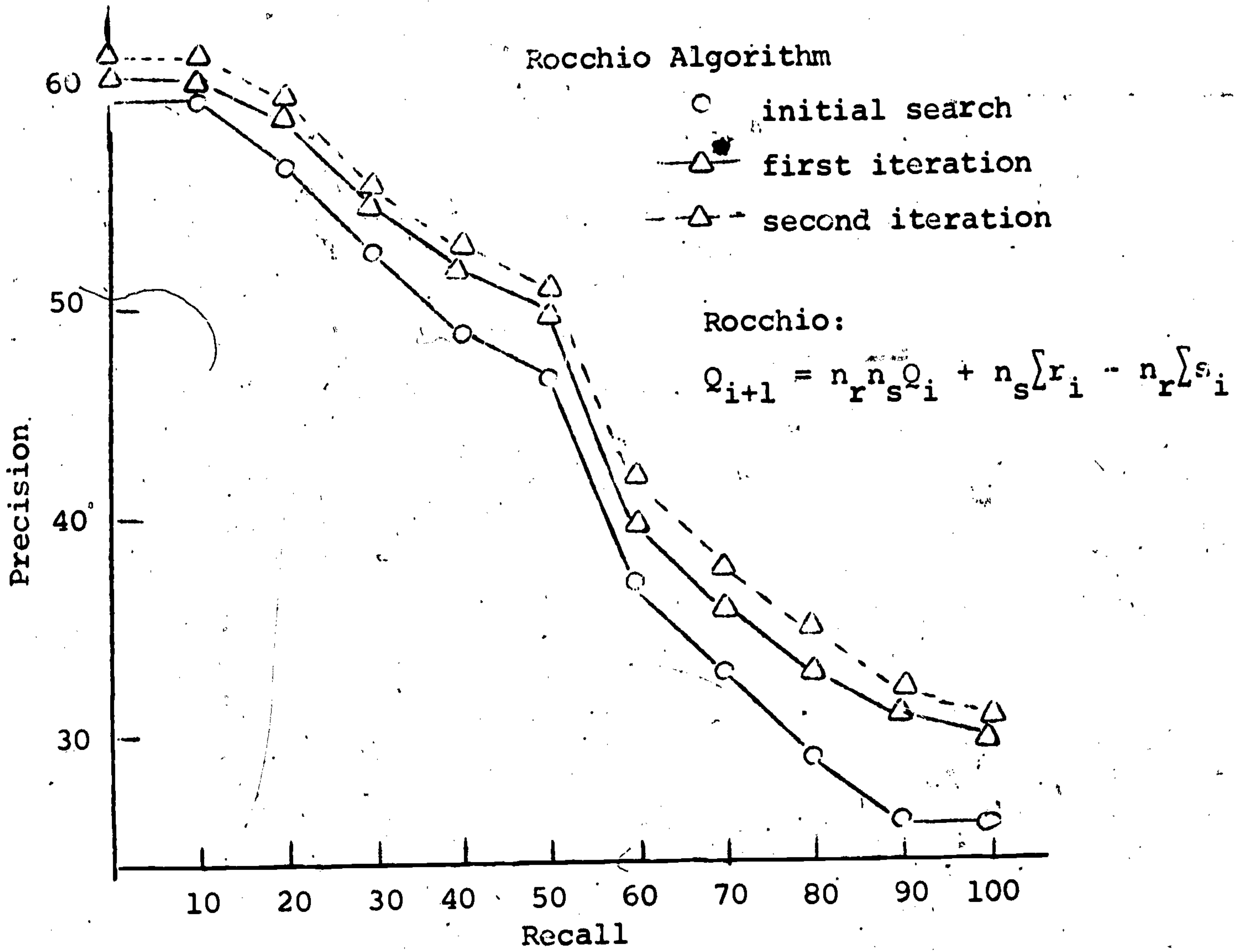
For the Q_o+ strategy, the differences between the initial search and each feedback iteration are significant. On the first iteration, the two overall measures and the precision differences from 20% through 50% recall are significant at the 5% level or less, and only at 70 and 80% recall are the precision differences not significant at the 10% level.* On the second iteration, the performance difference is significant at the 5% level for all points except 70% recall. For the Rocchio strategy, however, only one measure (precision at 50% recall) shows a significant difference between the first iteration and initial search at the 10% level or less.*

* For these comparisons, a one-tailed significance level is appropriate, since performance is expected to improve. To obtain one-tailed values, the reported two-tailed values must be divided by two. That is, the probability that the first iteration is no better than the initial search is 5% or less except at 70 and 80% recall.

		Normalized Recall			Normalized Precision	
		% Difference	T Test	WSR Test	% Difference	T Test
Q_o+ strategy minus Rocchio strategy	Iter 1	6.1	3.4	24.1	2.9	15.4
	Iter 2	4.4	17.7	48.9	2.0	43.0
Q_o+ strategy minus Dec Hi strategy	Iter 1	5.4	5.7	98.5	2.1	31.4
	Iter 2	7.0	9.4	45.6	3.0	29.6

Statistical Comparison of Feedback Effect
Relevant and Non-Relevant Document Strategies
Normalized Recall and Precision

Figure 17



Rocchio:

$$Q_{i+1} = n_r n_s Q_i + n_s \sum r_i - n_r \sum s_i$$

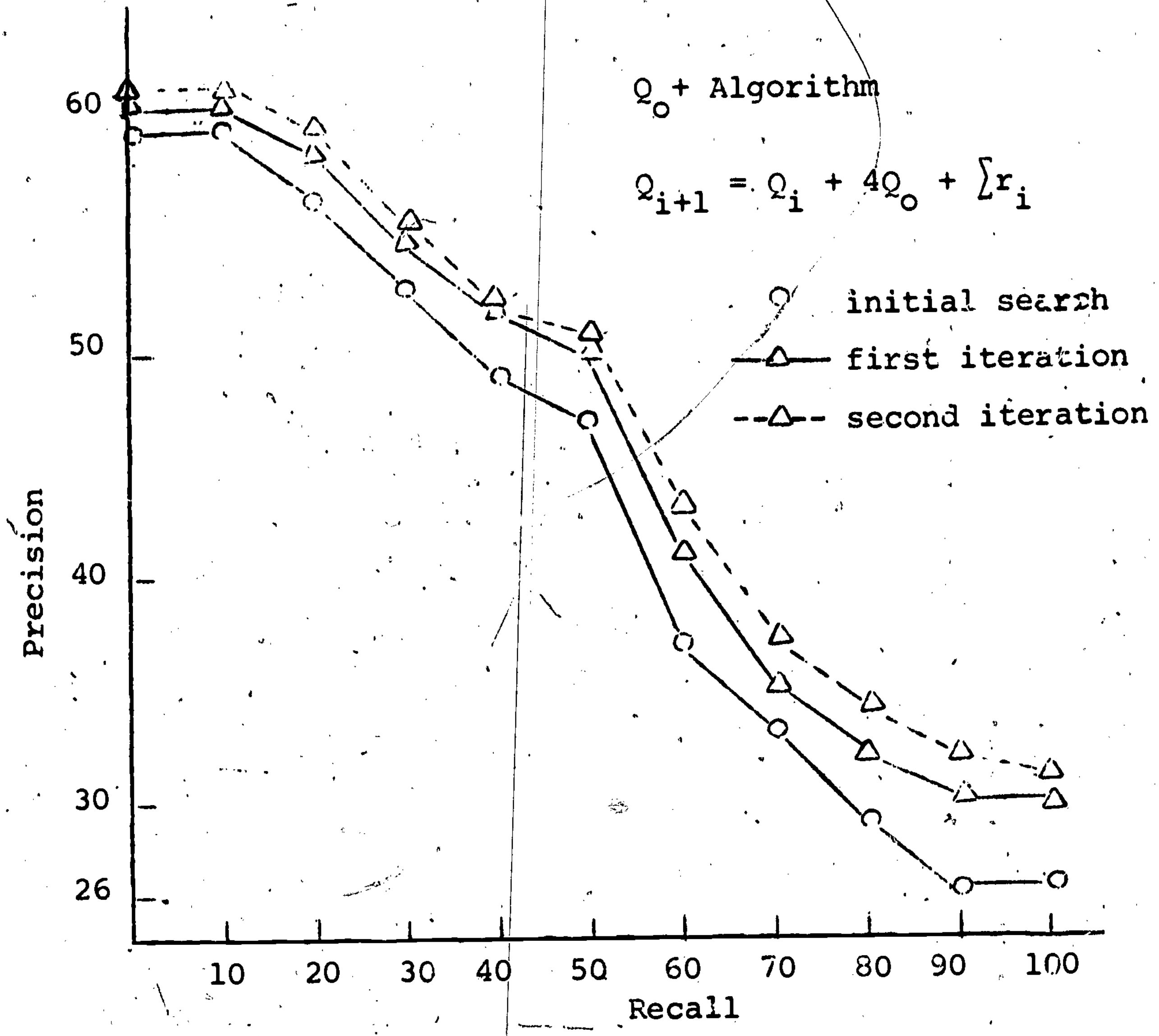
	First Iter Minus Initial Search		Second Iter Minus Initial Search		Second Iter Minus First Iter	
	% Difference	% Significance	% Difference	% Significance	% Difference	% Significanc
Normalized Recall	-3.0	60.0	-0.8	23.5	2.2	17.4
Normalized Precision	-0.1	55.5	1.6	20.8	1.8	8.0
Recall Level 10%	1.0	51.8	1.6	30.7	0.6	3.6
20%	2.1	25.8	2.8	12.4	0.7	3.1
30%	2.3	14.9	3.2	5.3	0.9	1.9
40%	2.9	10.4	4.1	2.2	1.2	5.1
50%	3.2	3.8	4.3	3.0	1.1	10.9
60%	3.1	47.7	4.5	28.7	1.4	5.7
70%	3.1	43.2	5.4	12.0	2.3	3.1
80%	4.3	22.1	5.8	8.5	1.5	11.8
90%	4.4	19.1	5.1	5.7	1.3	8.5
100%	4.2	21.1	5.3	7.4	1.1	17.1

Comparison of First and Second Iterations to Initial Search

Differences and Significance Levels - Rocchio Algorithm

Feedback Effect Recall-Precision Curves

Figure 18



	First Iter Minus Initial Search		Second Iter Minus Initial Search		Second Iter Minus First Iter	
	% Difference	% Significance	% Difference	% Significance	% Difference	% Significance
Normalized Recall	3.1	0.6	3.7	0.1	0.6	24.9
Normalized Precision	2.7	1.8	3.6	0.5	0.9	16.5
Recall Level 10%	1.0	6.8	1.8	1.3	0.8	11.2
20%	1.8	4.3	2.5	0.8	0.7	17.7
30%	2.2	1.5	3.0	0.9	0.8	25.0
40%	3.0	0.5	3.7	0.4	0.7	26.4
50%	3.1	0.8	3.7	1.1	0.6	40.7
60%	3.5	5.7	5.7	2.8	2.1	7.0
70%	2.4	29.2	4.4	19.5	2.0	7.6
80%	3.2	18.3	4.9	4.5	1.7	4.1
90%	3.6	6.1	5.6	2.4	2.0	3.5
100%	3.6	6.1	5.2	2.6	1.7	9.8

Comparison of First and Second Iterations to Initial Search

Differences and Significance Levels - Q_o+ Algorithm

Feedback Effect Recall-Precision Curves

Figure 19

Even on the second iteration, only six of the twelve differences are significant at the 10% level or less, two at the 5% level or less. Significance results for the feedback Dec Hi strategy are similar.

When comparing first and second iterations, the Q_o+ results are no longer more significant than the Rocchio results. In fact, the Rocchio results are significant (10% level or less*) for eight of the twelve measures; the Q_o+ results for only five. The significant improvement between first and second iterations occurs at the high recall end of the Q_o+ curve, while the improvement for the Rocchio strategy is more evenly distributed.

This difference between strategies in the significance of the improvement over the initial search leads to a general conclusion: Performance on all measures is less consistent for the non-relevant document strategies than for the Q_o+ strategy. However, since the average magnitude of this improvement is equal for the three algorithms (from the significance results presented in Figure 5), it must be true that the Rocchio and Dec Hi strategies are better for some queries and worse for others than is the more consistent Q_o+ strategy.

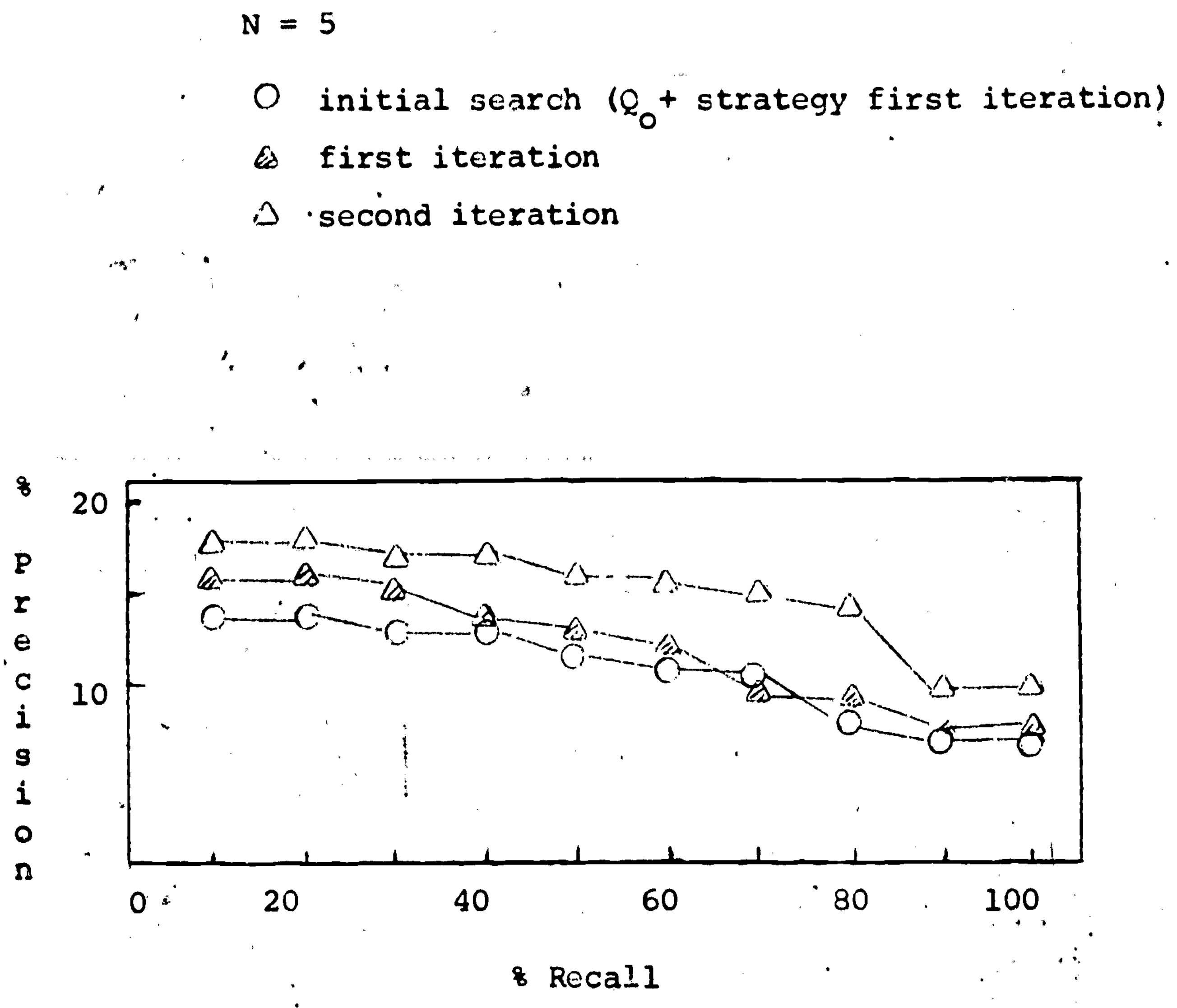
The total performance results of Figure 13 indicate that the queries that retrieve no relevant documents on the initial search are helped by the non-relevant feedback strategies. Figure 20 supports this conclusion with evidence that even using feedback effect evaluation, the Rocchio strategy provides better performance on these

IBID., p. VI-33.

eleven queries. Figure 14 adds the information that on the average, the total performance on the remaining 31 queries is not hurt by negative feedback. The preceding paragraph leads to the conclusion that in feedback effect the Rocchio strategy gives worse performance on some of these queries. This conflict between total performance and feedback effect results requires further investigation of subgroups of queries.

The document curve differences presented in Figure 21 provide new information about the performance of the negative feedback strategies. The Q_o+ strategy is taken as the norm, and the Rocchio and Feedback Dec Hi differences from Q_o+ are graphed. For the first fifteen or so documents retrieved, both Rocchio and Feedback Dec Hi are superior in feedback effect performance to Q_o+. After 40 documents have been retrieved, both are much worse than Q_o+ in recall, though about the same in precision. The recall-precision curves of Figures 18 and 19 average out these two extremes, and lose the significant information.

Figure 20 strongly supports the conclusion implied by normalized recall, that on the first iteration non-relevant document strategies tend to raise some relevant doucments, but to lower others that are already low in rank. The average advantage of the non-relevant document strategies appears early in the retrieval process. After 20% of the collection has been scanned, the Q_o+ strategy is clearly superior in recall. The rank-freezing of the feedback effect

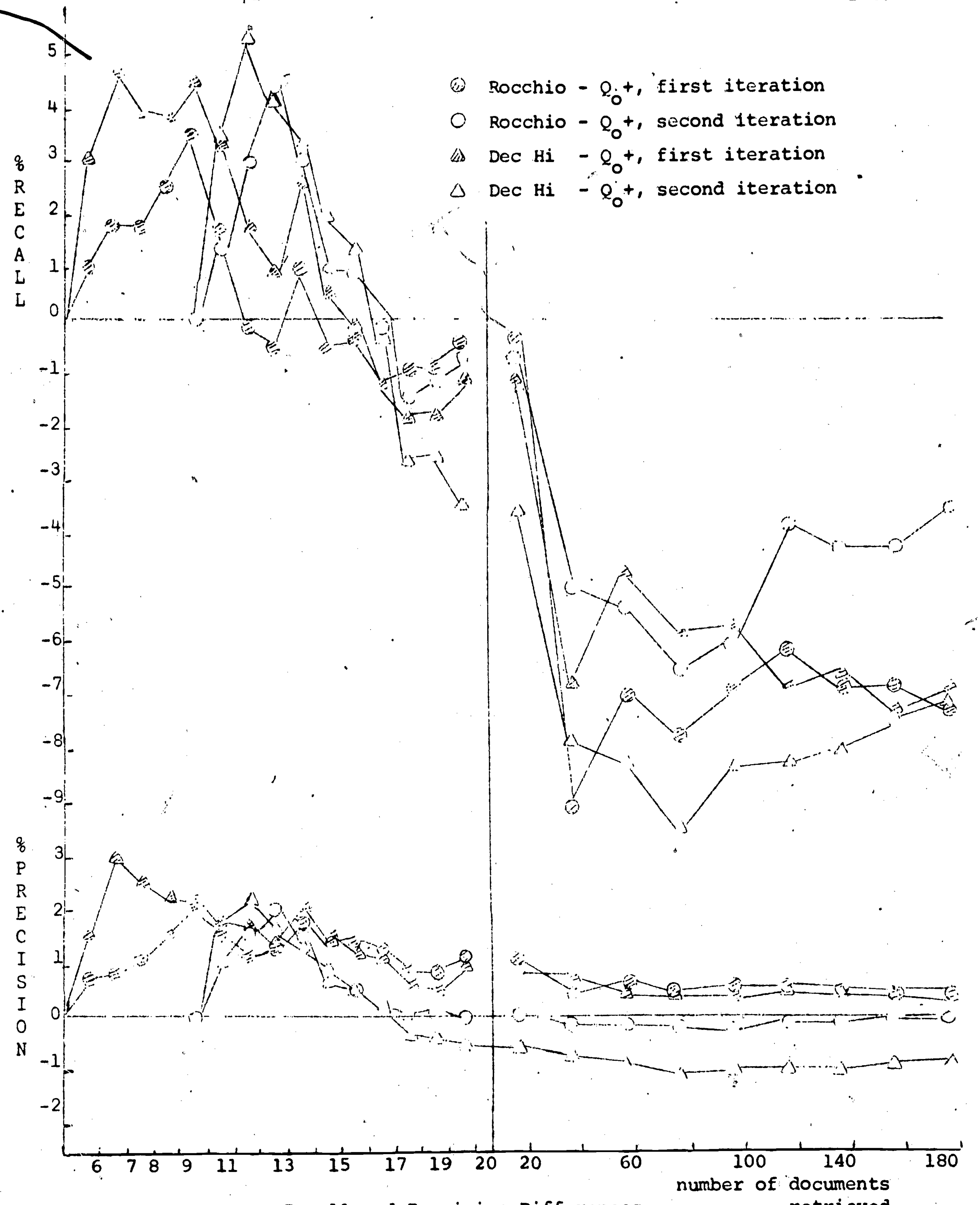


Rocchio Strategy

On Eleven Bad Queries

Feedback Effect Recall-Precision Curves

Figure 20



Recall and Precision Differences
Comparing Rocchio and Dec Hi Strategies
to Q_o+ norm
Feedback Effect Document Curves

Figure 21

evaluation affects the ranks of the earliest documents retrieved, so the recall-precision curves of the non-relevant document strategies appear superior in total performance but only equal in feedback effect. Normalized recall expresses the later large drop in recall which overwhelms the earlier advantage of negative feedback. Thus, the document curves support and clarify the tentative deductions made from the less detailed measures presented earlier.

Total performance comparisons encourage the use of algorithms that employ negative feedback of non-relevant documents. However, the feedback effect results indicate that the performance of negative feedback algorithms is highly variable. These findings encourage a search for a means of predicting the appropriate strategy for a given query. For this reason the characteristics of selected subgroups of queries are explored in the following section.

E. Characteristics of Query Subgroups

To investigate the performance of positive and negative feedback in more detail, the available queries are split several ways into pairs of query subgroups. Each subgroup pair represents a contrast based on one or two characteristics. For example, all queries with four or fewer relevant documents might form one subgroup of a pair, and all queries with five or more relevant documents might constitute the contrasting subgroup of that pair. The six queries that retrieve all relevant documents with rank 5 or less on the initial search are omitted from analysis because relevance feedback cannot improve the feedback effect performance on these queries. Figure 22 lists the characteristics used for selection and describes some subgroups for which comparisons are reported.

Each subgroup is statistically compared to the contrasting subgroup using the Wilcoxon Rank Sum test (see Section V). All findings of less than 10% significance are reported in this section. If the word 'null' is used in the WRS column of a figure, the significance level of the indicated comparison is greater than ten percent, and the 'null hypothesis' of no difference between subgroups is supported. Although significance level from five to ten percent are not normally considered meaningful, they are reported here for two reasons. First, the WRS test is conservative when too many ties in rank occur, and the data contains ties. Second, the numbers of queries in each subgroup is low, and statistical significance is difficult to prove for small samples. For these reasons significance levels from five to ten percent may indicate areas for productive investigation using larger query collections and perhaps more sophisticated statistical techniques. Twenty-two variables are used for WRS comparisons within each subgroup pair. Two are not generated by the search process; the number of concepts in the initial query and the number of relevant documents (2 vars.). The three search-related measures used are correlation of the modified query with the original query, feedback effect normalized recall, and feedback effect normalized precision. Normalized recall and precision are calculated for the initial search (2 vars.) and all three measures are calculated for two iterations of two feedback strategies, the positive feedback Q_o+ strategy and the Rocchio algorithm

<table>
<tr><th>Selection Characteristic</th><th>Group Name (Size)</th><th>Group Description</th></tr>
<tr><td rowspan="2">Initial Search Retrieval</td><td>Eleven Bad (11)</td><td>No relevant documents are retrieved with rank 5 or less on the initial search.</td></tr>
<tr><td>Twenty-Five (25)</td><td>Some but not all relevant documents are retrieved with rank 5 or less on the initial search.</td></tr>
<tr><td rowspan="2">Relevance Feedback Performance</td><td>Good Performance (16)</td><td>At least one feedback strategy retrieves all relevant documents with rank 15 or less within three iterations.</td></tr>
<tr><td>Bad Performance (20)</td><td>No feedback strategy retrieves all relevant documents with rank 15 or less within three iterations.</td></tr>
<tr><td>Correlation of Modified Query with Original Query</td><td>
Low
High</td><td>Six subgroups are chosen. The number of queries in each is given in the following table:
<table>
<tr><td>Strategy</td><td colspan="3">Q_o+</td><td colspan="3">Rocchio</td></tr>
<tr><td>Iteration</td><td>1</td><td>2</td><td>1-2</td><td>1</td><td>2</td><td>1-2</td></tr>
<tr><td></td><td>17</td><td>16</td><td>13</td><td>16</td><td>19</td><td>17</td></tr>
<tr><td></td><td>19</td><td>20</td><td>17</td><td>20</td><td>17</td><td>19</td></tr>
</table></td></tr>
<tr><td rowspan="2">Relevance Feedback Strategy</td><td>Q_o+ (13)</td><td>The Q_o+ strategy retrieves more documents with rank 15 or less in three iterations.</td></tr>
<tr><td>Rocchio (15)</td><td>The Rocchio strategy retrieves more documents with rank 15 or less in three iterations.</td></tr>
<tr><td rowspan="2">Number of Concepts in Original Query and Number of Relevant Documents</td><td>High-Low or Low-High (17)</td><td>Queries having relatively many relevant documents and relatively few concepts or vice versa:
From 2-3 relevant and 7+ concepts (2)
From 4-5 relevant and 10+ concepts (5)
From 5-6 relevant and 3-6 concepts (5)
7+ relevant and 3-7 concepts (5)</td></tr>
<tr><td>Similar (19)</td><td>Queries having a number of concepts and a number of relevant documents similar in magnitude:
From 2-4 relevant and 3-6 concepts (8)
From 4-6 relevant and 7-9 concepts (6)
6+ relevant and 8+ concepts (5)</td></tr>
</table>

Some Query Subgroups Investigated

Figure 22

which uses negative feedback (12 vars.). For normalized recall and precision, the improvement caused by feedback over the initial search is used for comparison to remove the effect of initial search differences between subgroups. To provide a direct comparison between positive and negative feedback, the differences between the Q_o+ and Rocchio strategies in normalized recall and precision for two iterations are used (4 vars.). Finally, the difference between the first and second iteration correlation of the modified query with the original query is calculated for each strategy (2 vars.). Obviously significant relationships such as the difference in number of relevant documents between queries with four or fewer and queries with five or more relevant doucments are not reported.

Normalized recall and normalized precision were chosen for the subgroup comparisons because they are overall measures of retrieval. However, the analysis in the previous section indicates that the normalized figures are not representative of overall performance as indicated by the recall-precision curves and the document curves. In particular, normalized recall shows a large drop for the Rocchio strategy, and neither recall nor precision reflects the initial advantage of the Rocchio strategy displayed in Figure 21. Therefore, the normalized measures may not be the best choice for meaningful comparison of positive and negative feedback.

Figures 24, 28, and 29 in this section display recall-precision curves. Unfortunately, significance tests between subgroups for recall-precision curves are not available. However, in three subgroup pairs, selected by strategy, performance, and number of relevant documents, Wilcoxon Signed Rank tests of the difference between the Q_o+ and Rocchio strategies within each subgroup were made. All differences were significant in both strategy subgroups; no differences were significant in any other subgroup.

In the figures of this section, the average values of variables as well as the WRS probabilities are presented. It should be noted that the WRS probabilities do not indicate the significance of differences in average value, but the significance in differences in rank sum when all queries in both subgroups are ranked. The average value is reported because it is a more familiar figure and conveys more intuitive meaning than the rank sum.

The first subgroup pair listed in Figure 22 is familiar from the previous section. Figure 13 and 20 present total performance and feedback effect recall-precision curves for the 'eleven bad' group, both showing an advantage for the Rocchio strategy. Figure 23 presents some of the significant WRS findings for this group. The average number of relevant documents for the eleven bad queries is 4.3, contrasting with 5.6 for the remaining twenty-five queries. The WRS probability that these subgroups represent populations that have the same distribution of number of relevant documents is less than ten percent, so the difference is of doubtful significance. When the Q_o+ improvement is compared to the Rocchio improvement, the normalized recall and precision indicate an advantage for the Q_o+ strategy in both subgroups. This finding contradicts the recall-precision curves presented earlier, and is misleading for the reasons stated early in this section. The meaningful conclusion to be drawn from Figure 23 is that the differences in feedback improvement between subgroups are not significant except for the first iteration of the Q_o+ strategy. That is, the performance of the Rocchio strategy does not depend on whether or not relevant documents are available for feedback. This conclusion agrees with the recall-precision curves for this subgroup. It should be noted that the low average normalized figures for the eleven bad queries are due to a single query, query 34, that is destroyed by all negative feedback strategies. Since magnitude is reflected in the averages but not in the WRS test, query 34 has a disproportionate effect on the average figures but does not similarly bias the probabilities.

The comparisons using correlation of the modified query with the original query show stronger differences between subgroups than the performance comparisons. The Rocchio strategy changes the query more in both subgroups, as expected. The Q_o+ strategy changes the eleven queries not at all on the first iteration and very little on the second. The figure for the first iteration correlation minus the second iteration correlation indicates that the eleven bad queries tend to move further away from the original query on the second it ration, but the twenty-five queries tend to stay about the same difference from the original query. The direction of the Rocchio strategy comparisons is the same as that of the Q_o+ comparisons, but all correlations are much weaker. The eleven bad queries change significantly less than the twenty-five on the first iteration, but on the second the amount of change no longer differs. The ten-

'Eleven Bad' Group: Eleven queries that retrieve no relevant documents with rank 5 on the initial search.

'Twenty-Five' Group: Twenty-Five queries that retrieve some but not all relevant documents within rank 5 on the initial search.

			Eleven Bad	Twenty-Five	WRS Probability
Number of Relevant Documents			4.3	5.6	<10%
Initial Search		NR	76.4	88.0	<02
		NP	42.8	72.0	<01
First Iter.	Improvement Q_o+ Strategy	NR	0.0	5.2	<05
		NP	-0.2	4.6	<02
	Improvement Rocchio	NR	-18.7	3.3	<10
		NP	-0.5	3.2	null
Second Iter.	Improvement Q_o+	NR	1.7	5.3	null
		NP	-0.3	5.0	null
	Improvement Rocchio	NR	-10.9	3.5	null
		NP	-2.1	3.7	null
Correlation of modified query with original query	Q_o+ Strategy	Iter 1	100.0	78.5	<01
		Iter 2	95.9	78.8	<01
		Iter 1-2	4.1	-0.4	<02
	Rocchio Strategy	Iter 1	59.7	43.8	<01
		Iter 2	50.2	45.8	null
		Iter 1-2	9.4	-0.1	<01

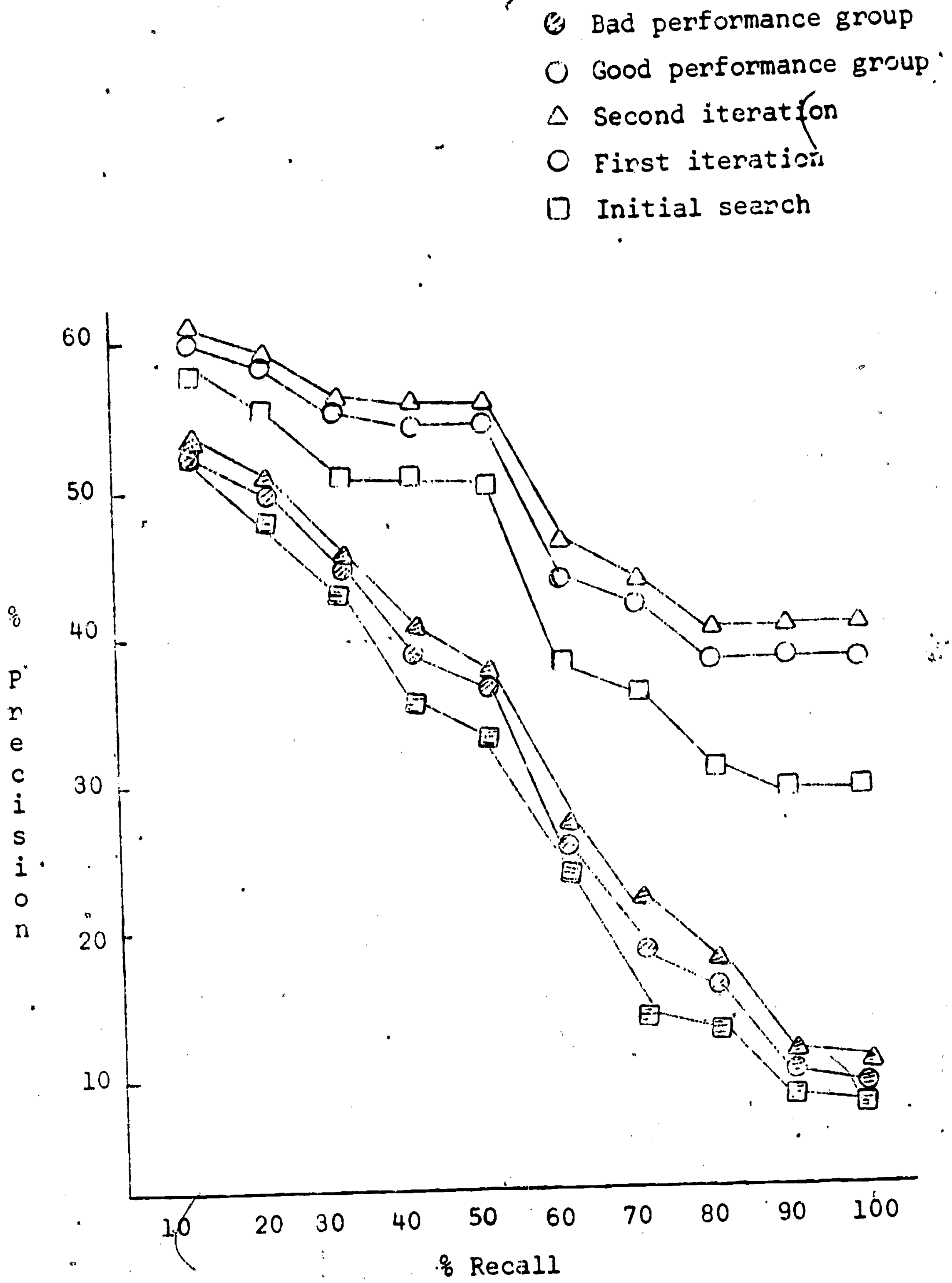
Characteristics of
Subgroups Selected by Initial Search Retrieval

Figure 23

dency for the eleven bad queries to move further away from the original query on the second iteration is stronger for the Rocchio strategy, so that the second iteration change compensates for the lack of change in the first iteration.

Figure 24 presents feedback effect recall-precision curves for the subgroups selected by performance. These curves seem to indicate not only better initial search performance, but also greater first iteration improvement for the good performance group. The precision level of the good performance group drops less as recall increases than that of the bad performance group. The normalized recall and precision reported in Figure 25 indicate better initial search but slightly worse first iteration feedback for the good performance group. The first and second iteration feedback improvement differences are not significant. It is interesting to note that the eleven bad queries on the initial retrieval had fewer relevant documents than the remaining twenty-five, but the bad performance group tends to have more relevant documents than the good performance group. Also noteworthy is the significant tendency of the second iteration Rocchio query to move further away from the original query in the bad performance group, as though searching farther afield for relevant documents. This tendency is not observed for the Q_o+ strategy.

Figure 26 describes the general behavior of the modified queries in relation to the initial query. For both strategies the first iteration and second iteration queries tend to be similar in correlation with the original query. For the Q_o+ strategy, queries that don't move far



Subgroups Selected By Performance
Feedback Effect Recall-Precision Curves
Rocchio Strategy

Figure 24

Good Performance Group: Sixteen queries that retrieved all relevant documents with rank greater than 16 in three iterations of at least one feedback strategy.

Bad Performance Group: Twenty queries that did not retrieve all relevant documents with rank greater than 16 in three iterations of any feedback strategy.

			Good Performance	Bad Performance	WRS Probability
Number of Relevant Documents			4.5	5.7	<10%
Correlation of Modified query with original, Rocchio Strategy		Iter 1	50.9	46.8	null
		Iter 2	51.1	41.6	null
		Iter 1-2	-.2	5.2	<02
Initial Search		NR	90.6	79.5	<01
		NP	70.5	57.1	<10
First Iter.	Improvement, Q_o+ Strategy	NR	2.2	4.8	null
		NP	2.9	3.4	null
	Improvement, Rocchio	NR	-5.3	-1.9	null
		NP	-0.5	0.1	null
Second Iter.	Improvement, Q_o+ Strategy	NR	3.8	4.7	null
		NP	5.2	3.4	null
	Improvement, Rocchio	NR	-0.9	-0.9	null
		NP	2.5	1.4	null

Characteristics Of

Subgroups Selected By Performance

Figure 25

		WRS Probability
Correlation of Modified Query with Original Q_o+	First Iter./Second Iter	<01%
	First Iter./Iter 1-Iter 2	<01
	Second Iter/Iter 1-Iter 2	null
Correlation of Modified Query with Original Rocchio	First Iter./Second Iter	<01
	First Iter./Iter 1-Iter 2	<10
	Second Iter/Iter 1-Iter 2	null
Correlation of Modified Query with Original Q_o+ / Rocchio	First Iter	<05
	Second Iter	null
	Iter 1 - Iter 2	<01

Cross-probabilities for
Correlation of Modified Query
With Original Query

Figure 26

from the original query on the first iteration tend to move farther on the second; this tendency is much weaker for the Rocchio strategy. The first iteration correlations for the Q_o+ strategy and Rocchio strategy tend to vary similarly, the second iteration queries are no longer related; but the movement between first and second iterations strongly tends to be in the same direction.

Figure 27 reports the characteristics of five of the six subgroups chosen by correlation of the modified query with the original query. For the Q_o+ strategy on both iterations, queries that are more correlated with the original query tend to have fewer relevant documents and inferior performance. These findings can be explained by the behavior of the eleven queries that do not retrieve relevant documents initially. For the Rocchio strategy, there is a slight counter-tendency for queries that remain more correlated with the original on the second iteration to have more rather than fewer relevant documents. The significant findings for the Rocchio strategy concern the direction of query change between first and second iterations. Queries that mcve further from the original query tend to have more relevant documents and poorer performance. This tendency agrees with the earlier finding in Figure 25. The subgroups chosen on the basis of Q_o+ query change between first and second iterations are not shown because for all variables the differences between subgroups support the null hypothesis.

Thus far no relationships explaining the differences in performance between positive and negative feedback have been observed. No subgroup pair has shown significant differences on the four final variables;

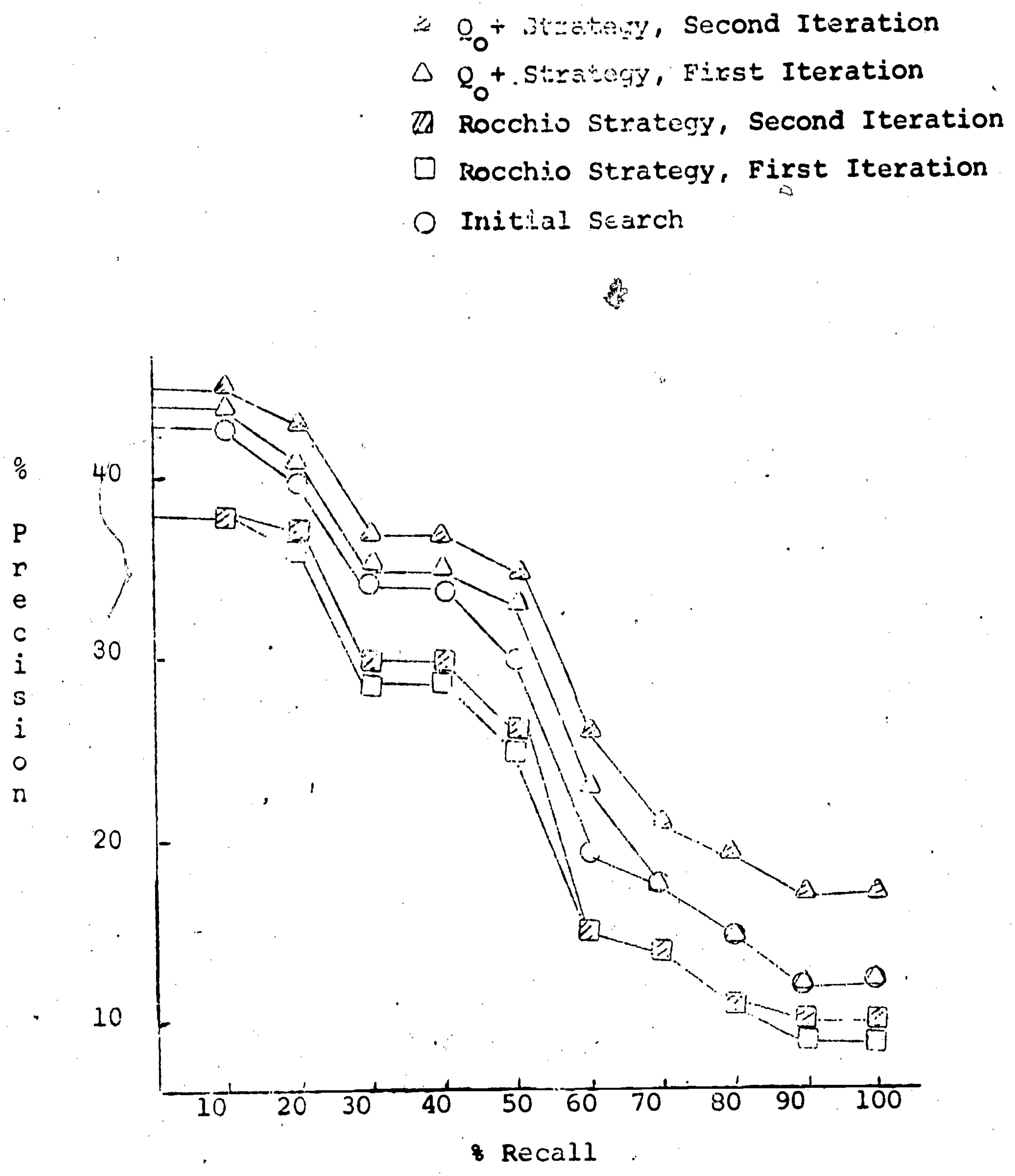
				Initial Search	
			Number Relevant	Normalized Recall	Normalized Precision
Correlation of Modified Query with Original Q_o^+	First Iter	Low	5.9	88.2	73.7
		High	4.5	81.1	53.5
		WRS	<10%	<10%	<01%
	Second Iter	Low	6.3	90.4	76.5
		High	4.3	79.7	52.3
		WRS	<01	<05	<01
Correlation of Modified Query with Original Rocchio	First Iter	Low	5.1	84.6	65.7
		High	5.3	84.3	60.9
		WRS	null	null	null
	Second Iter	Low	4.9	80.3	58.4
		High	5.4	89.1	68.2
		WRS	<10	null	null
	Iter I minus Iter 2	Low	4.3	92.5	74.2
		High	5.9	77.2	53.1
		WRS	<05	<01	<01

Characteristics Of
Subgroups Selected By
Correlation of Modified Query
with Original Query

Figure 27

difference between Q_o+ and Rocchio strategies for first and second iterations. The strategy subgroups were chosen in an attempt to explore these differences from the opposite direction, to select the queries that display differences between positive and negative feedback and see if they also show other differences. Thirteen queries showing superior performqnce with Q_o+ and fifteen showing better performance with Rocchio were selected; the remaining eight queries showed no difference between strategies.

Figures 28 and 29 show the feedback effect recall-precision curves for each strategy in each group. In the Q_o+ group, the Q_o+ strategy causes slight improvement on the first iteration and more improvement on the second over the initial search, but the Rocchio strategy degrades performance. The initial search performance of the Rocchio group is higher than that of the Q_o+ group until 70% recall. Both the Q_o+ and Rocchio strategies improve performance in the Rocchio group, but the Rocchio improvement is greater. In Figure 29 the initial search on the remaining queries is graphed, showing that initial performance is far superior for those queries that have equivalent performance on both strategies. Figure 30 shows a similar pattern in the normalized recall and precision. In all cases, the improvement caused by the Q_o+ strategy is statistically equivalent in the two groups, but the Rocchio strategy degrades performance in the Q_o+ group. Except for normalized recall in the first iteration, the Rocchio strategy improves



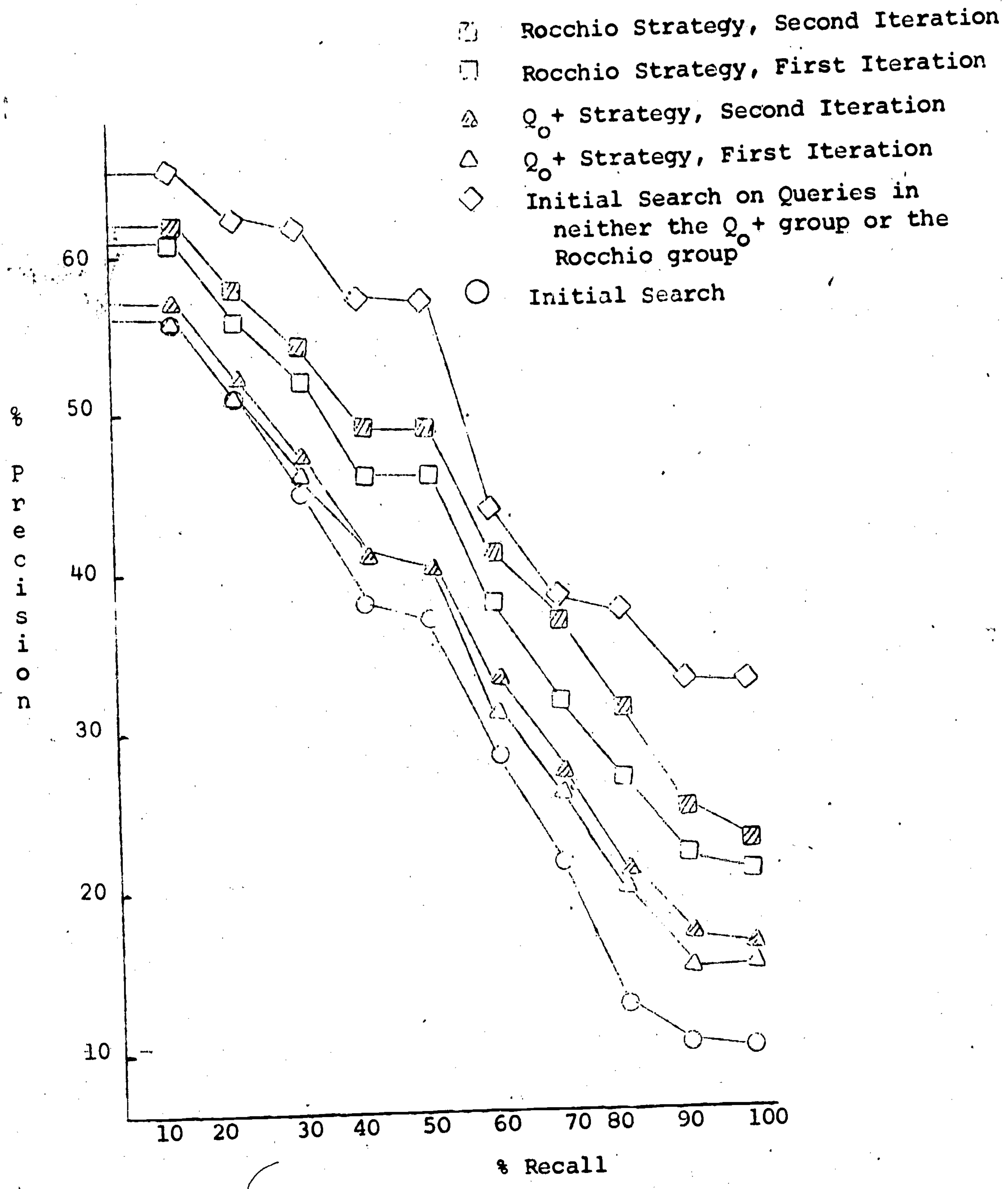
Subgroups Selected By Strategy

Q_o+ Group

Feedback Effect Recall-Precision Curves

Comparing Positive and Negative Feedback

Figure 28



Subgroups Selected By Strategy

Rocchio Group

Feedback Effect Recall-Precision Curves

Comparing Positive and Negative Feedback

Figure 29

Q: Q_o+ Strategy

R: Rocchio Strategy

Q-R: Q_o+ strategy minus Rocchio strategy

Q_o+ Group: Thirteen queries that have better performance with the Q_o+ strategy.

Rocchio Group: Fifteen queries that have better performance with the Rocchio strategy.

			Q_o+ Group	Rocchio Group	WRS Probability
Initial Search	Normalized Recall		83.9	82.1	null
	Normalized Precision		58.6	60.3	null
First Iter	Normalized Recall	Q	2.9	3.3	null
		R	-16.3	2.8	<01%
		Q-R	19.2	0.5	<01
	Normalized Precision	Q	1.9	1.5	null
		R	-12.0	7.2	<01
		Q-R	13.9	-5.7	<01
Second Iter	Normalized Recall	Q	4.5	3.2	null
		R	-14.7	7.5	<01
		Q-R	19.2	-4.3	<01
	Normalized Precision	Q	4.2	1.4	null
		R	-10.6	10.7	<01
		Q-R	14.8	-9.3	<01

Comparison of Positive and Negative Feedback

In Subgroups Selected by Feedback Strategy

Figure 30

performance in the Rocchio group more than the Q_o+ strategy does. The hypothesis of greater variability in performance with the Rocchio strategy is again reinforced.

Unfortunately, the WRS tests show no differences between the Q_o+ and Rocchio groups except in feedback performance. The indication of the recall-precision curves that the Rocchio group is superior on the initial search is not supported by the normalized measures. No differences in number of concepts, in number of relevant documents, or in query correlations are found.

To further investigate strategy differences, three subgroup pairs are chosen based on feedback improvement in the normalized measures. One subgroup includes all queries that show feedback improvement over the initial search for all measures; normalized recall and precision for two iterations of both strategies. There are only thirteen queries in this group, because of the zero change in the 'eleven bad' queries with the Q_o+ strategy and the first iteration normalized recall plunge often encountered with the Rocchio strategy. The contrasting subgroup of the 'All Measures' pair contains the twenty-three queries that have zero or negative improvement on any measure. Two other subgroup pairs are chosen similarly, one by feedback in improvement on all measures of the performance of the Rocchio strategy (17 queries improved on all Rocchio measures, 19 did not), and the other by feedback improvement for the Q_o+ strategy (16 queries improved on all Q_o+ measures, 20 did not).

Figure 31 displays the significant differences for the three subgroup pairs. A tendency for the improved queries to be less correlated with the original query is seen in the All Measures pair. This tendency is even more significant in the pair based on Q_o+ measures, but it disappears in the Rocchio Measures pair. The 'Eleven Bad' queries that retrieve no relevant documents on the initial search account in part for these findings. None of the queries in the 'Eleven Bad' group are in the 'All Q_o+ Measures' group. However, three of the eleven bad queries improve on all Rocchio measures. The eleven bad queries have high correlations with the original query, especially on the first iteration of the Q_o+ strategy when the correlation equals 1 for all eleven queries. The differences in correlation in the Q_o+ Measures subgroup pair cannot be entirely explained by the eleven queries, however. The eleven queries do not improve in both the All Measures and the Q_o+ Measures pairs, yet the Q_o+ differences in correlation are more significant than the All Measures differences.

Again an a priori choice of subgroup pairs forces significant differences in the performance of negative and positive feedback strategies. Figure 31 shows no significant relationship in the Q_o+ Measures pair with the differences between the Q_o+ and Rocchio strategies. However, in the Rocchio Measures pair all strategy difference measures show relationships significant at the one percent level. The relationships in these two subgroup pairs support the conclusion drawn from Figures 28 through 30 that the Rocchio performance variability creates the performance differences between strategies. A tendency for the thirteen queries that improve on all measures to favor the Rocchio strategy is significant only for first iteration precision improvement. This tendency supports the difference in recall-precision curves observed in Figures 28 and 29. In these figures the Rocchio strategy improves the Rocchio group more than the Q_o+ strategy improves the Q_o+ group on the first iteration.

Except for a tendency explained by initial search retrieval, no relationships have been found that can predict feedback improvement for the Q_o+ strategy or the Rocchio strategy. However, the lack of a relationship between feedback improvement and initial search performance is encouraging, since it indicates that relevance feedback causes as much improvement in original queries providing inadequate information as it causes in initially well-phrased queries.

Neither the experimental nor the analytical approach isolates a single variable that predicts performance differences between negative and positive feedback. At this point, although several aspects of retrieval behavior have been detailed, initial search performance seems to be the only effective predictor of final results. However, it seems anomalous that neither of the search-independent variables is related to any performance variable. No subgroup pair shows any difference in number of concepts in the original query, and differences in number of relevant documents are significant at the ten percent level or insignificant. Subgroups based on the number of concepts or on the number of relevant show no relationship with any variable.

Number of concepts is a measure related to the length of the query, and indicates the amount of detail with which the user has specified his needs. The number of relevant documents is an indication of how wide a subject area the user's query is intended to specify within the given document collection. Although these two variables are theoretically important to retrieval, each has no individual relationship to performance, and they are not related to each other. Therefore, it seems probable that the number of concepts in the original query and the number of relevant documents have some joint relationship to retrieval behavior. In fact, Figure 32 shows that these two variables combined are the desired predictor of performance differences between negative and positive feedback.

Two contrasting subgroups are chosen based on the relationship of the number of concepts to the number of relevant. In the 'Similar' group the two numbers are either both low, both high, or both in mid-range. The contrasting 'High-low, Low-high' group contains those queries with few relevant and many concepts or with few concepts and many relevant. The Similar group attains significantly better performance with the Rocchio strategy than does the High-low, Low-high group. The differences between the Q_o+ and Rocchio strategies favor the Q_o+ strategy in the High-low, Low-high group and the Rocchio strategy in the Similar group. In short, every significant relationship in Figure 30 is echoed in Figure 32. The fact that some Figure 32 relationships are weaker can be attributed in part to the eight 'neutral' queries omitted from Figure 30 but included in Figure 32. Three of these fall in the Similar group; the remaining five in the contrasting group. The average differences in Figure 32 compare favorably to those in Figure 31. Except for first iteration normalized recall, the differences between the Similar and High-low, Low-high groups are as great or greater than the differences between queries that improve on all Rocchio measures and those that don't.

The joint relationship of query size and number of relevant documents is of little use for prediction in an operating retrieval system, since the number of relevant documents in the collection is not known at the beginning of the search. However, some estimator of the number of rele-

All Measures Group:	Improved on all normalized measures for two iterations of both Rocchio and Q_o+ strategies. (13 queries)
Not All Group:	Zero or negative improvement on any measure, any strategy, any iteration. (23 queries)
All Q_o+ Measures Group:	Improved on all normalized measures for two iterations of the Q_o+ strategy. (16 queries)
Not All Q_o+ Group:	Zero or negative improvement on any Q_o+ measure. (20 queries)
All Rocchio Measures Group:	Improved on all normalized measures for two iterations of the Rocchio strategy. (17 queries)
Not All Rocchio Group:	Zero or negative improvement on any Rocchio measure. (19 queries)

	All Measures	Not All	WRS	All Q_o+ Measures	Not All Q_o+	WRS	All Rocchio Measures	Not All Rocchio	WRS
Iter									
Q_o+ 1	77.8	89.1	=1%	77.9	90.8	<1%	82.5	87.3	null
2	75.9	88.7	<1	76.0	90.5	<1	80.6	87.1	null
Rocchio 1	42.4	52.2	<5	41.1	54.6	<1	45.7	51.2	null
2	42.2	47.8	null	40.5	50.0	<10	42.8	48.5	null

Correlation of Modified Query With Original Query

Iter 1 NR	-0.4	11.3	null	2.6	12.5	null	-2.0	15.2	<1%
NP	-1.2	4.8	<5	0.1	4.7	null	-5.6	10.0	<1
Iter 2 NR	-0.8	8.4	null	-0.2	9.3	null	-4.0	13.1	<1
NP	-1.6	3.2	null	-0.3	2.9	null	-6.7	8.8	<1

Q_o+ Strategy Minus Rocchio Strategy

Comparison of Negative and Positive Feedback
In Subgroups Chosen by Feedback Improvement

Figure 31

'Similar' Group:	Number of concepts in original query and number of relevant documents are similar in magnitude; From 2-4 relevant and from 3-6 concepts, from 4-6 relevant and from 7-9 concepts, 6 or more relevant and 8 or more concepts.
'High-low, Low-high' Group:	Few concepts and many relevant or few relevant and many concepts. Not meeting the criteria of the 'similar' group.

			High-low Low-high	Similar	WRS Probability
Initial Search	Normalized Recall		82.2	88.4	null
	Normalized Precision		60.0	65.8	null
First Iter	Normalized Recall	Q	4.8	2.5	null
		R	-10.7	3.1	<10%
		Q-R	15.6	-0.5	<01
	Normalized Precision	Q	3.5	1.6	null
		R	-6.8	5.8	<05
		Q-R	10.3	-4.2	<01
Second Iter	Normalized Recall	Q	6.1	2.5	null
		R	-10.4	7.7	<05
		Q-R	16.5	-5.2	<01
	Normalized Precision	Q	5.4	1.6	null
		R	-6.1	9.0	<02
		Q-R	11.5	-7.5	<01

Comparing Positive and Negative Feedback
In Subgroups Selected by
Number of Concepts in Original Query
and Number of Relevant Documents

Figure 32

vant documents might be available to the system before feedback. The user could be asked to state whether he intends his query to be specific or general, and some users might even be able to estimate the number of relevant documents available. In a larger collection the number of relevant documents retrieved on the initial search might be useful for prediciton as the number of relevant documents available. In this collection the number of relevant documents retrieved by the original query when N equals 5 correlates highly with the number of relevant documents in the collection. Spearman's coefficient of rank correlation is significant at the one percent level. [21] However, the number of relevant documents retrieved can range from 0 to 5 only, and this range does not provide sufficient information for prediction of differences in performance of negative and positive feedback strategies.

When the number of relevant retrieved and the number of concepts are used to predict strategy differences, the WRS test results support the null hypothesis. Nevertheless, a search for a predictive relationship between query size and some estimator of the number of relevant documents might well be profitable in a larger collection.

The results in Figure 32 indicate the possibility of taking advantage of the performance differences between negative and positive feedback by choosing in advance the appropriate strategy for each query. Another approach is to develop a single algorithm that causes feedback improvement on all queries. With this possibility in mind, the factors causing the failure of the Rocchio algorithm on some queries in the High-low, Low-high group should be investigated. It is evident from earlier results that the inferior Rocchio performance on some queries is not caused by a failure to retrieve relevant documents on the initial search. In fact, the possible obliteration of the initial query by subtraction of non-relevant documents does not appear to be a general problem. Only query 34 is reduced to zero by the Rocchio strategy. All other queries gain in length on the first iteration. Of the ten queries that lose some concepts, seven gain in performance from the change.

The data presented in this section does not directly indicate the causes of the variability of the Rocchio strategy. In Section VII-C a hypothesis consistent with all experimental results is advanced to explain the contrasting behavior of positive and negative feedback.

Chapter VII

Recommendations Based on Present and Prior Experiments

In this section, recommendations for practical interactive retrieval systems and for further research in relevance feedback are made. First, specific recommendations for operational interactive retrieval are drawn from the experimental results presented in the previous section. Then five general areas of concern are discussed and research problems are suggested using present and prior experiments as foundations for conjecture. These general areas are evaluation of relevance feedback performance, feedback of non-relevant documents, partial search strategies, and multiple query feedback strategies.

A. Relevance Feedback Recommendations for Concept Vector Document Classifications Systems

The findings of this study apply to retrieval systems that use positively weighted concept vectors to describe both documents and retrieval requests. Caution is necessary in generalizing these results to systems that differ from the SMART system in such aspects as the vector distance function used for retrieval, or the significance of vector position and weight magnitude. If the cosine correlation is used as the distance function, and if each vector position signifies a subject classification, and if the magnitude of a weight is in some way related to the importance of the corresponding subject in the document being classified, the means of construction of the concept vectors should not affect the applicability of these results.

Because of the characteristics of the Cranfield 200 collection (Section IV), these results are most relevant to collections of articlelength documents from limited technical subject areas, classified by information from the document abstracts. The following considerations are important when generalizing to document collections of realistic size.

a) The generality (ratio of number of relevant documents available to collection size) of a larger document collection would be lower. Lower generality would result in lower precision and less striking precision improvement. [18]

b) The relationship of the number of documents provided for feedback to retrieval results would change as the relationship of this number to the collection size changes. Five documents, the number used most often for feedback in this study, constitute 2.5% of the experimental collection, equivalent to fifty documents in a collection of as few as two thousand documents. Therefore the results presented in Section VI-C must be interpreted with regard to the relative as well as the absolute magnitude of N.

c) The proportion of retrieved relevant to retrieved non-relevant documents might become smaller in a larger document collection, although this proportion probably would not maintain a constant relationship to the ratio of retrieved documents to collection size. Comparison of feedback strategies using only relevant documents to those using non-relevant documents would be particularly affected by this proportion.

d) The number of queries available for the experimental collection is dangerously low from a statistical viewpoint. The subgroups results in Section VI-E divide a barely adequate query sample into even smaller groups. Although care has been taken to choose contrasting subgroups of near equal size, results of these experiments cannot be used for practical recommendations without verification in larger collections.

Because of the importance of these size considerations, experimentation with larger document collections is strongly recommended. The 1000 to 5000 document size is convenient for many reasons. First, the time and money needed for experiments would not be prohibitive. Second, a collection of that size could be found that would be useful to some professional or student group, so that actual users might be made available. Third, subject area clusters of this size would probably constitute a lower search level in a multi-level algorithm for large libraries, so the techniques found useful by experiment could be directly applied as subunits of such an algorithm.

Despite the limiting considerations listed above, some recommendations can be drawn from the data presented. First, the general usefulness of the relevance feedback technique is supported. Comparison of a larger and more carefully chosen experimental document collection (Cranfield 200) to a smaller and less realistic one (ADI) encourages the generalization of these results to even larger collections by demonstrating that feedback improvement is maintained in spite of a lower ratio of relevant to nonrelevant documents available. For a more definite confirmation of this finding performance in the Cranfield 200 collection should be compared to that in the full Cranfield collection, because the ADI and Cranfield 200 collections are not directly comparable in subject area, query construction, or document characteristics.

The demonstrated stability in the performance of algorithms using only relevant documents for various relative weightings of the original query and the retrieved documents also supports the general usefulness of of the technique. None of the formulas used in this study for 'relevant only' strategies can be chosen as superior. This conclusion agrees with the results reported by Crawford and Melzer [12], who find no indication that the original query must be retained after the initial search.

Firm conclusions can be reached concerning the number of documents used for feedback with strategies using only relevant documents. The performance improvement caused by feeding back more documents is impressive up to five percent of the collection and still noticeable at seven and one-half percent of the collection. In a document collection of useful size, input-output time and user effort would limit feedback to far less than five percent of the collection. Therefore the following algorithm for determining the number of documents to be used for feedback is recommended for larger collections on the basis of the results in Section VI-C.

At least n documents are initially retrieved for each user. If none of these n are judged relevant, more documents are retrieved until one relevant document is found or N documents have been retrieved. The numbers n and N are chosen considering cost, input-output time, and user effort in the particular retrieval system. From the results of this study a value of 5 or more is suggested for n, and a value less than or equal to five percent of the collection is recommended for N. This combination feedback algorithm should be tested with strategies that use nonrelevant documents for feedback.

For queries retrieving no relevant documents within N documents, the Rocchio strategy (Section VI-D, reference 9) using nonrelevant documents is recommended. In fact, the Rocchio strategy is often superior to strategies using relevant documents only even when relevant documents are available for feedback. In the experimental collection the Rocchio strategy is superior on 36% and equal on 32% of the queries that retrieve some but not all relevant documents on the first iteration. Nevertheless, because of the variability in negative feedback performance reported in Section VI-D, feedback of nonrelevant documents cannot be recommended as a general strategy. Possible causes of negative feedback variability are discussed in Section VII-C. The recommendation of the Rocchio strategy for queries retrieving no relevant documents is supported by Steinbuhler and Aleta. [13]

B. Evaluation of Relevance Feedback Experiments

The evaluation problems encountered in this study give rise to several suggestions for future experiments. Some of the recommendations made in this section are applicable only to the evaluation of interactive feedback techniques, but others are generally valid for information retrieval experiments.

The variability of the results reported in Section VI-D casts doubt on all comparisons of average values of retrieval performance measures, and demands tests of statistical significance for meaningful comparison of retrieval parameters. In Figure 17, a difference of 7% in normalized recall is not statistically significant, yet in Figure 19 a difference of 3.1% is found significant at the 0.6% level. Obviously it is dangerous to use the magnitude of performance differences as the only indicators of significance in the experimental environment of this study. The same evidence supports the recommendation that larger query samples be obtained.

The apparent conflict between the normalized recall measure and the recall-precision curves for negative feedback is resolved by the document curves of Figure 20. This suggests that valuable information is lost by attempts to condense complex retrieval information into overall performance measures. Even the ten-point recall-precision curves do not preserve the information contained in the document curves. The two-valued measure, normalized recall and precision, loses all indication of the superiority of the Rocchio strategy when less than 40% of the collection has been retrieved.

This situation presents a problem in evaluation, because available tests of statistical significance deal with single valued measures of performance. Determining the joint significance of more than one measure requires that the statistical dependence of each measure on the other be known. In information retrieval, all measures of performance are based on a single ranked list and thus cannot be assumed independent, yet the dependence of one measure on another is difficult to determine, and may vary in different experimental situations. For this reason no attempt is made in this study to estimate the joint significance of more than one performance measure. Since no single valued measure preserves the information most meaningful to these experiments, there is no way to determine the overall statistical significance of the differences between positive and negative feedback strategies.

In this complex experimental environment it is imperative that the experimenter have a clear conception of the questions he is asking, and that he choose performance measures that can answer his questions. A convincing example of this necessity occurs in Section VI-C of this study, where the tried-and-true recall-precision curves are found inappropriate as a measure of the effect of amount of feedback on performance. Both in Section VI-C and Section VI-D the document curves are used to prevent misinterpretation of the more common measures. The evaluation problems mentioned stimulate thought in three areas: Summary measures of performance, interpolation methods for recall-precision curves, and evaluation methods for interactive strategies. The suggestions made in these areas arise directly from consideration of the questions being asked by the experimenter and the questions being answered by the performance measure.

Although summary measures of performance such as normalized recall and precision lose information, they are nevertheless valuable for statistical evaluation. Since all information cannot be retained in a summary measure, a measure of the aspect of performance most relevant to the experiment should be chosen. The failure of normalized recall and precision to reflect the early retrieval advantage of the Rocchio strategy suggests that these measures are answering the wrong question. They sum the recall and precision at each possible cut-off point over the entire document collection, and weight each possible recall or precision value equally. From a practical standpoint however, early retrieval performance is more important than performance after most of the collection has been retrieved, especially when interactive iterative search algorithms are being tested. Normalized recall in particular seems intuitively inappropriate for this study in that a change in rank from 195 to 191 has the same effect on normalized recall as a change from five to one. Yet the idea of summing recall or precision at all cut-off values is a sound basis for a summary measure of performance. Two alternate measures, called weighted recall and weighted precision, are suggested that preserve the summation idea but attach greater importance to earlier retrieval. Rocchio's normalized recall and precision are stated most simply by the following formulas:

$$NR = \frac{1}{N} \sum_{j=1}^{N} R_j$$

$$NP = \frac{1}{N} \sum_{j=1}^{N} P_j$$

where R_j and P_j are recall and precision at a cut-off of rank j. Weighted recall and precision give a weight of N to the recall and precision values at rank 1 and progressively smaller weights to later values, as indicated by the following formulas:

$$WR = \frac{2}{N(N+1)} \sum_{j=1}^{N} (N-j+1)\ R_j$$

$$WP = \frac{2}{N(N+1)} \sum_{j=1}^{N} (N-j+1)\ P_j$$

The multiplier 2/N(N+1) gives weighted recall and precision the same range as normalized recall and precision. Similar formulas can be constructed giving more or less relative weight t. earlier retrieval performance. In this way a range of two-valued summary performance measures can be provided from which an experimenter can select the measure that reflects his concerns.

The interpolation methods used for the recall-precision curves in this study have been supported or criticized in the past with regard to the 'meaning' of the average curve obtained. Statistical tests of the significance of precision differences at each level of recall treat each interpolated value as a measure of the performance of a single query, and may compare interpolated precision values to actual values achieved by other queries. Thus the meaning of the single interpolated value is the important factor in a choice of interpolation method, because each interpolation method defines a performance equivalence relation among queries with different numbers of relevant documents.

To make this point clearer, the example query of Figures 1 and 2 is used. This query, now called query A, has four relevant documents and retrieves them with ranks of 4, 6, 12, and 20. Suppose query B has eight relevant documents. What ranks are assigned to these eight documents by query B if it achieves performance equivalent to that of query A?

The rank of every other relevant document retrieved by query B is determined by the precision after each relevant document of query A is retrieved. That is, the second relevant document is retrieved by query B with rank 8, giving precision of 2/8 (1/4). The fourth relevant document of query B has rank 12, the sixth rank 24, and the eighth rank 40. The ranks of the first, third, fifth, and seventh documents relevant to query B are determined by the interpolation method used, because for statistical comparison the precision after the first relevant document of query B is retrieved must be equivalent to the interpolated value for query A at 12.5%, the precision after the third relevant must be equivalent to the interpolated value for query A at 37.5%, and so forth.

Figure 33 gives the ranks of the eight relevant documents of query B that are defined as 'equivalent' to the ranks of the four relevant documents of query A: 4, 6, 12, and 20, by several interpolation methods including Quasi-Cleverdon and Neo-Cleverdon. Only exact integer ranks are assigned in the SMART system, but integer ranks equivalent to the ranks listed for Quasi-Cleverdon could occur if a query had enough relevant documents. Note the underlined rank of 6 given to the second relevant document by the Neo-Cleverdon interpolation. At this point the Neo-Cleverdon interpolation ignores the actual Query A precision at 25% recall and assigns a new precision value. This discarding of achieved recall levels is done by Neo-Cleverdon whenever the precision at a subsequent recall level is higher. The 'Lower Limit' interpolation represents the worst performance any query could achieve and still maintain the Query A precision values at 25%, 50%, 75%, and 100% recall. The 'Upper Limit' interpolation represents the best possible performance. The 'Equal Proportion' interpolation expresses the intuitively appealing idea that the relevant documents retrieved between the recall levels achieved by Query A should be ranked half-way between the adjacent well-defined ranks. Figure 33 shows that the ranks defined by Quasi-Cleverdon Interpolation are only slightly different than the Equal Proportion ranks. However, the Neo-Cleverdon interpolation is closer to the bottom limit after the highest precision point has been achieved and near the top limit before the high point of precision. The underlined rank of 6 is in fact above the top limit.

Query A: An example query with four relevant documents. For query A precision values at points other than 25%, 50%, 75%, and 100% recall must be interpolated.

Query B: A hypothetical query with eight relevant documents that achieves performance 'equivalent' to query A. In each column of the table, the ranks of the eight relevant documents of query B are set to give the same precision as the interpolated precision defined for query A by a given interpolation method.

Quasi-Cleverdon
Neo-Cleverdon: Interpolation methods described by Figures 1 and 2.

Bottom Limit: An interpolation method based on the bottom limit of performance that a query could achieve and still have precision values equivalent to those at the uninterpolated points of the given query.

Top Limit: An interpolation method based on the top limit of performance a query could achieve and still have equivalent precision values at the uninterpolated points.

Equal Proportion: An interpolation method based on assigning an interpolated rank at each recall point such that the assigned rank and the adjacent uninterpolated ranks are related in the same proportion as are the recall points of interpolation and the adjacent achieved recall levels.

Recall Level	Query A Ranks	Equivalent Query B Ranks as Defined by:				
		Quasi-Cleverdon	Neo-Cleverdon	Lower Limit	Equal Proportion	Upper Limit
12.5%		4	3	8	4	1
25	4	8	6	8	8	8
37.5		10.3	9	12	10	8
50	6	12	12	12	12	12
62.5		17.3	20	24	18	12
75	12	24	24	24	24	24
87.5		31.5	35	40	32	24
100	20	40	40	40	40	40

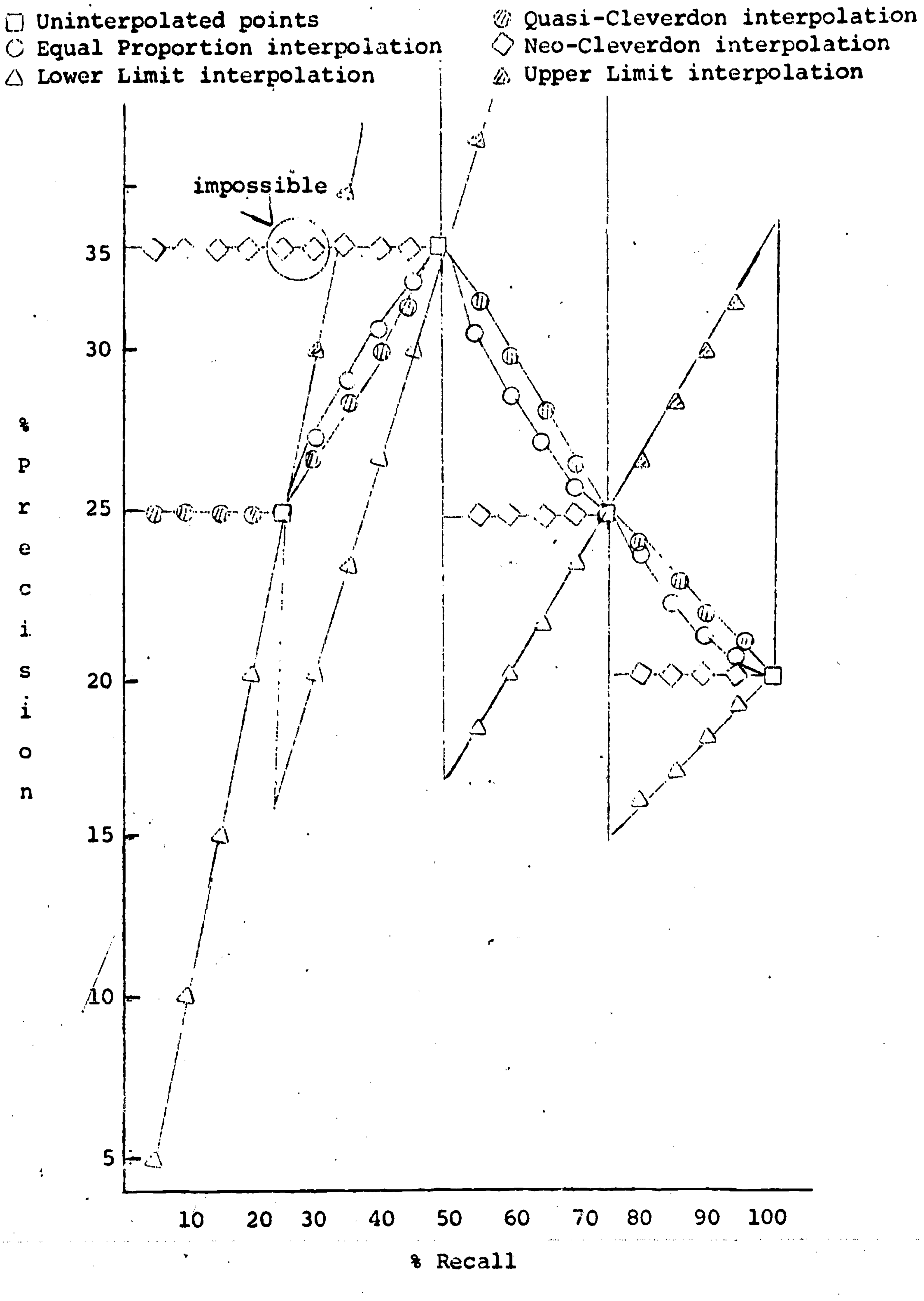
Examples of Performance Equivalence Between Queries
As Defined By Different Interpolation Methods

Figure 33

Figure 34 demonstrates all five interpolation methods in graphical form. The small squares on the graph represent the uninterpolated precision values achieved by Query A. All other figures represent the interpolation points at each five percent of recall defined by the five interpolation strategies described. It is evident that the Quasi-Cleverdon and Equal Proportion interpolations are almost identical. The Upper Limit interpolation is not graphed until 25% recall since it assigns 100% precision to all earlier points. Beyond 25% recall the Upper Limit and Lower Limit interpolations define quadrilaterals within which any precision value is possible to a query with equivalent precision at the uninterpolated points. The two circled points of the Neo-Cleverdon interpolation are outside the defined quadrilateral and thus are impossible.

Note that none of the interpolated curves bear any resemblance to the sawtooth curve of Figures 1 and 2. The sawtooth curve represents the behavior of precision values in a single query between achieved recall levels. It is completely irrelevant to interpolation, because the interpolated values are statistically compared to the precision at achieved recall levels of other queries, not to precision between achieved levels. A comparison of Figure 1 and Figure 34 shows that many points on the sawtooth curve fall outside the range of possible interpolation points defined by the Lower Limit and Upper Limit interpolations.

A basic question arises from this discussion: 'What interpolation method provides the most appropriate definition of equivalent performance for queries having different numbers of relevant documents?' Figure 34 shows that the range of possibly equivalent interpolation points is great. One way of defining equivalence would be to pick the performance within the possible range that has the same probability of occurrence as the per-



Twenty-Point Interpolation From Example Query A

Using Five Different Interpolation Methods

Figure 34

formance of the query for which interpolated values are sought. However, if it is assumed that integer ranks are assigned randomly without replacement, the Lower Limit interpolation curve in Figure 34 describes a performance level more probable than the performance of example Query A. The assumption of random assignment of ranks is inappropriate for information retrieval, because both the query vectors and the document vectors would have to be random. The controlling probability determinant for this study is the set of document vectors, because it is unchanged from one experiment to the next. Thus an appropriate definition of equivalent performance would be performance equally probable in the given set of document vectors. This probability could be estimated from experimental results. For experiments that evaluate changes in the document space, the determinants of probable performance would be the constant factors in the experimental environment.

A rough estimate of an appropriate equivalence relation can be derived from the fact that normalized recall, normalized precision, and the document curves each provide a definition of equivalent performance. Equal performance for both recall and precision is defined by the document curves as performance equal at each cut-off rank, and by the normalized measures as performance giving an equal sum over all cut-off ranks. Since precision is defined as relevant retrieved divided by total retrieved, normalized precision equivalent to query A is provided by a query that retrieves a relevant document at ranks 4, 6, 12, and 20 and no other relevant documents, or by any query that has the same sum of precision at each cut-off value. Lower Limit interpolation provides a lower overall sum of precision values because the same precision levels are achieved at lower ranks. Therefore, the normalized precision definition of equivalence would give slightly higher interpolated values than does Lower Limit interpolation.

Recall, however, is defined as relevant retrieved divided by total relevant, so query B would have recall equivalent to query A if it could somehow retrieve the first two relevant documents with rank 4, the second two with rank 6, the third two with rank 12, and the last two with rank 20. The normalized recall definition of equivalent performance demands n times the precision at each recall level for a query with n times the number of relevant documents. The Upper Limit interpolation gives lower values than this at all points.

To provide some estimate of a reasonable equivalence relation for the experimental environment of this study, the relationship of number of relevant documents to initial normalized recall and precision are presented. Spearman's coefficient of rank correlation is positive for precision (.25) and slightly negative for recall (-.017). [21] If either normalized precision or normalized recall provided a valid definition of equivalent performance for this query and document collection, the number of relevant documents would show no correlation with that measure. Therefore a definition of equivalence that coincides with the performance observed in this environment would be somewhere between the definition implied by normalized precision and that implied by normalized recall, but closer to that of normalized recall. That is, an appropriate interpolation method would be closer to Upper Limit interpolation than to Lower Limit interpolation. This conclusion contradicts the opinion expressed by the proponents of Neo-Cleverdon interpolation that the Quasi-Cleverdon method gives artificially high results. The rank correlations of initial normalized recall and precision to the number of relevant documents indicate that the Quasi-Cleverdon interpolation may be conservative in the environment of these experiments.

Three considerations mentioned in the foregoing discussion support the recommendation that Quasi-Cleverdon interpolation rather than Neo-Cleverdon interpolation be used for investigations in query and document collections similar to the Cranfield data. First, the Neo-Cleverdon interpolation supplies data points that could not occur in a query with precision at uninterpolated recall levels equal to that of the query being represented. Since interpolated data points are statistically compared to achieved data points of other queries, ignoring some of the achieved data points of a query is inappropriate. Second, Quasi-Cleverdon interpolation gives results similar to an intuitively pleasing method (Equal Proportion) that assigns an interpolated rank half-way between the ranks a query with comparable precision at uninterpolated data points could achieve. Third, data supports the conclusion that the Quasi-Cleverdon interpolation does not give interpolated points that are artificially high in this experimental environment. Further investigation of the relationship of retrieval performance to the number of relevant documents should be conducted to support the choice of an interpolation method that provides a meaningful definition of equivalent performance for different queries. Such an interpolation method could lead to more general and more meaningful use of recall-precision curves as measures of retrieval performance.

When Hall and Weiderman [17] propose feedback effect evaluation, they are saying that total performance measures do not answer the question that is most relevant to relevance feedback experiments. Considerations of the questions the experimenter wishes to ask leads to the construction of several evaluation methods appropriate for relevance feedback, one of which is also useful in evaluating other strategies that require partial searches of the document collection.

Total performance, when evaluating an iterative strategy, answers the question 'How much closer is the modified query vector to the optimum query vector (Section III, Reference 9)?' Hall and Weiderman state that "For a relevance feedback system the measure of its effectiveness should be a measure of how many new relevant documents are retrieved as a result of feedback." [17] However, feedback effect evaluation does not measure the variable that Hall and Weiderman propose. Instead, it answers the question 'What is the overall retrieval performance of the system after each iteration from the viewpoint of the user who is interacting with the system?'

The distinction being made above is based on the components of performance that are isolated for measurement. To measure how many new relevant documents are retrieved by feedback, the change in performance caused by the feedback on each iteration must be isolated from all other factors. In feedback effect evaluation, the early retrieval of previous iterations becomes an albatross which is hung on each new iteration, so that the possibility of change in reported performance becomes less for each iteration. The description of feedback effect in Section V-C makes this point clear.

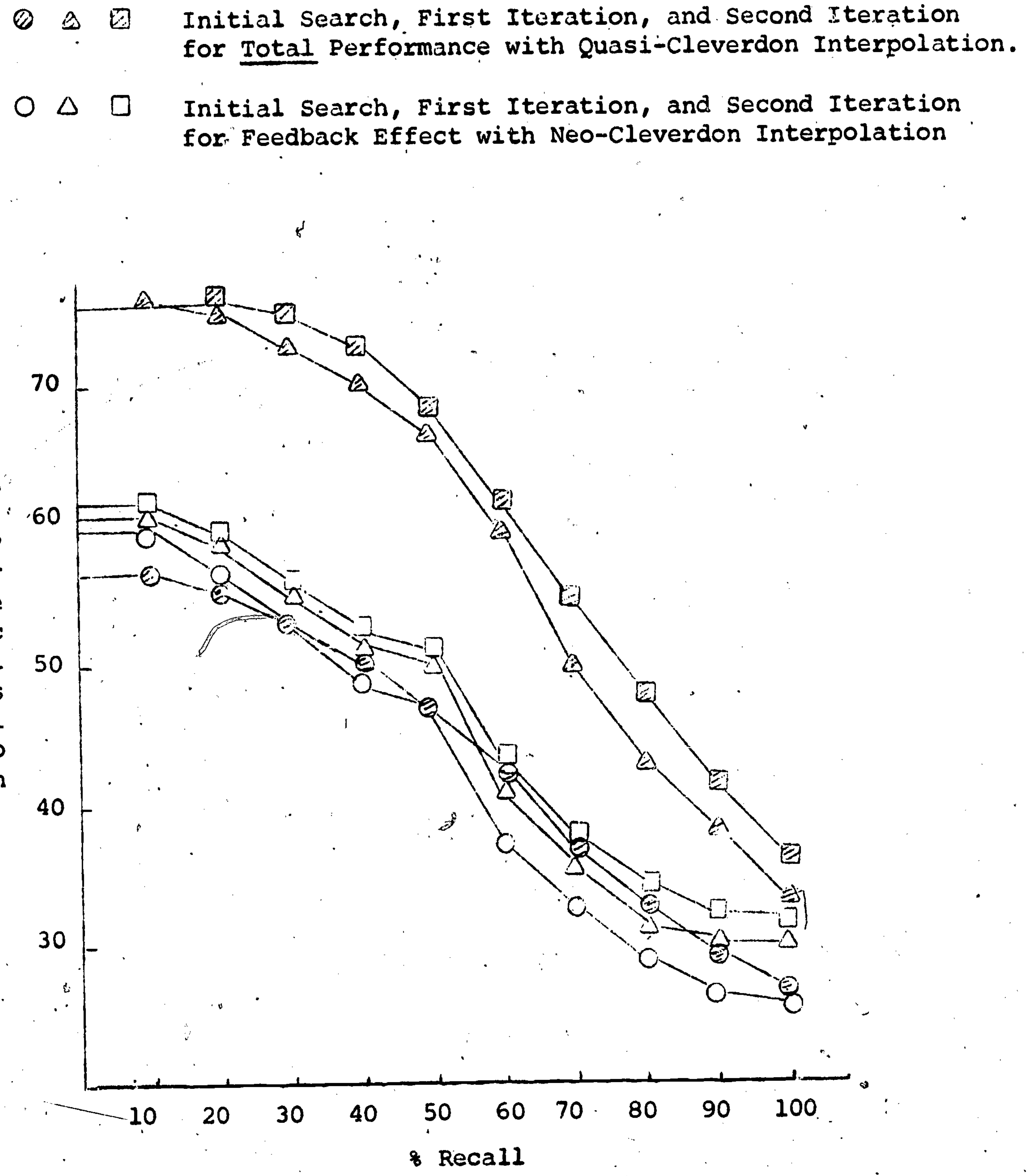
Before further discussion of the question that Hall and Weiderman ask, the question that Feedback Effect evaluation answers is explored. Figure 35 shows total performance evaluation with Quesi-Cleverdon interpolation and Feedback Effect evaluation with Neo-Cleverdon interpolation for two comparable strategies (Total Performance Q_o and Feedback Effect Q_o+) with N equal to 5. The difference between the initial search curves is entirely due to the difference in interpolation methods.

The first iteration total performance curve shows that for the average information request the modified query vector is much closer to the optimum query vector than is the original query vector. The change in total performance from first iteration to second iteration indicates much less change in the query vector was caused by relevant documents retrieved on the first iteration than was caused by relevant documents retrieved on the initial search. This smaller change is due in part to the fact that total performance evaluation does not retrieve five new documents for the second iteration modification. The five documents retrieved on the initial search are new, but the five retrieved on the first iteration probably include all relevant documents retrieved on the initial search.

The feedback effect curves show much less change than the total performance curves after the initial search, demonstrating that the user interacting with the feedback system observes little of the effect of the change in the position of the query vector. The largest changes in the feedback effect curves are observed at high recall levels, because the freezing of the early retrieval limits possible gains in precision at low recall levels. The slight improvement observed at low recall levels is probably due to the leftward extension of later precision improvements by Neo-Cleverdon interpolation.

The comparison of total performance curves to Feedback Effect curves shows that great improvement in the position of the query vector is needed before the user at the teletype notices an overall improvement in interactive retrieval. This conclusion is significant in predicting the psychological impact of automatic interactive retrieval on its users.

Two methods of evaluation answer two valid questions about interactive retrieval systems. Yet other questions can be asked, and other evaluation methods can be constructed to answer them. Four examples of possible evaluation methods are presented below, one of which answers the Hall-Weiderman question 'How many new relevant documents are retrieved as a result



Comparison of Two Evaluation Methods

Total Performance Evaluation with Quasi-Cleverdon Interpolation

and

Feedback Effect Evaluation with Neo-Cleverdon Interpolation

Comparable Strategies, N = 5

Figure 35

of feedback?'

The total performance evaluation method is inappropriate for relevance feedback because it does not ensure that the documents used for feedback have not been encountered previously. This fault in the evaluation method does not invalidate the question it is intended to answer: 'How much closer is the modified query to the optimum query?' The total performance algorithm is here modified to answer this question for more than one iteration of relevance feedback. The modified algorithm flags all documents presented to the user for feedback, and presents N new documents on each iteration regardless of the rank of the N^{th} new document in the list ranked by total performance. The recall-precision curves resulting from this algorithm could be directly compared to those generated by the total performance algorithm in experiments not involving document feedback. However, the document curves are changed in meaning because different queries would assign different ranks to the N^{th} new relevant document. Nevertheless, the modified total performance algorithm is needed to determine the performance increment caused by feedback iterations after the first. Both total performance and feedback effect evaluation limit the attainable performance of later iterations. Both evaluation methods therefore indicate a sharp drop in performance improvement after the first iteration, and both indicate so little improvement between the second and third iterations that third iteration results are not reported in this study. Modified total performance evaluation might show subsequent feedback iterations to be nearly as valuable as the first in moving the modified query toward the optimum query, and thus might stimulate further study of later iterations of relevance feedback.

Two evaluation methods similar to feedback effect evaluation have been discussed, both of which indicate better preformance after feedback than feedback effect evaluation does. One of these methods assigns all previously retrieved relevant documents the highest possible ranks and all previously retrieved non-relevant documents the lowest possible ranks. This algorithm is here called 'best list' because it answers the question 'Using all information available to the system, what is the best ranked list of documents that can be presented to the user after the iteration being evaluated has made a search?' This algorithm would report better performance for iterations after the first than any evaluation method discussed earlier in this report, because it maximizes ranking effect for each query. However, the impressive performance changes would not be informative, because most of the improvement reported by best list evaluation would not be caused by the changes made to the query vector as a result of feedback.

A better alternative that retains an outlook important to the user is here called 'modified feedback effect' evaluation. Feedback effect freezes the ranks of all documents presented to the user on earlier feedback iterations, and assigns the first document retrieved on the ith iteration a rank of iN+1, if N documents are used for feedback on each iteration. Modified feedback effect freezes the ranks of all previously retrieved <u>relevant</u> documents, and assigns the first document retrieved on the ith iteration the rank next below that of the last retrieved relevant document. Non-relevant documents retrieved with ranks higher than that of the last retrieved relevant document retain their earlier ranks, and non-relevant documents retrieved with lower ranks are re-ranked by the modified query. Like modified total performance evaluation, modified feedback effect retrieves N new documents on each iteration regardless of the rank of the N^{th}. It answers the question 'What is the best performance that can be achieved this iteration given the performance indicated by (i.e. without changing the ranks of) the relevant documents already seen by the user?' Hall and Weiderman define ranking effect as "changes in the rank of relevant documents previously seen by the user" (underscore mine) [17]. By this definition, modified feedback effect evaluation is the appropriate measure of feedback effect. Since non-relevant documents retrieved below that last retrieved relevant document are re-ranked rather than being pushed to the bottom of the list, all performance improvement between iterations can be attributed to changes in the query.

Feedback effect evaluation and modified feedback effect evaluation have a common characteristic; performance on each feedback iteration is limited by the early retrieval performance already achieved. Thus neither way of measuring 'feedback effect' directly answers the Hall-Weiderman question 'How many new relevant documents are retrieved as a result of feedback?' Rephrased in terms of overall performance so that 'retrieved' need not be defined, this question is 'What is the performance of the modified query with respect to the relevant documents that have not yet been presented to the user?' By Rocchio's theory [9], this is equivalent to the question 'How close is the modified query to the optimum query for the documents not yet presented to the user?' Therefore, to answer Hall and Weiderman's question the evaluation method must treat the remainder of the document collection as a complete collection and the remainder of the relevant documents as a complete set of relevant documents, and perform a total performance evaluation of the modified query in this new environment. The evaluation method constructed to answer the three equivalent questions posed above is called 'residual collection' evaluation.

Three problems are encountered in the construction of a residual collection evaluation method. The first and most obvious problem occurs when all relevant documents are retrieved before all requested iterations are completed. This problem is solved by dropping all queries that retrieve all relevant documents from the query sample for later iterations, and reporting the number of queries remaining in the sample for each iteration. This solution has the advantage of eliminating all meaningless information from the evaluation of each iteration to give an unbiased indication of the improvement obtained as a result of feedback. Of course, a user in a real environment might conduct fruitless searches, not knowing that all relevant documents available had been retrieved. Residual collection evaluation ignores this user's dilemma because there is no unambiguous way to take account of it in the experimental environment.

The second problem occurs in averaging the performance of different queries. Each query may have a different residual collection for a given iteration. This poses no problem unless the number of documents used for feedback is not the same for all queries, in which case the residual collections are not the same size. The variable feedback situation does not change the meaning of normalized measures or of recall-precision curves as long as the appropriate collection size is used for averaging. However, two possible methods of document curve construction exist. Recall and precision could be averaged after an absolute number of documents had been retrieved, or at percentiles of the document collection. Since recall and precision values change rapidly at the higher ranks and since all queries would be averaged into the earliest retrieval points, the absolute number of documents retrieved is an appropriate evaluation dimension for early retrieval. Percentile of the collection is a better evaluation dimension for document curves intended to summarize overall performance.

The third problem involves comparison of results between iterations. For two retrieval algorithms, performance measures obtained by residual collection evaluation could be directly compared for each feedback iteration. However, if one iteration of a feedback strategy is compared to a previous iteration of the same strategy, the question asked of the comparison must be specified. Direct comparison is appropriate if the question asked is of this type: 'How much more improvement occurred as a result of first iteration feedback than occurred as a result of second iteration feedback?' This is a meaningful question that cannot be asked of other evaluation methods. However, a quite different type of question is often asked: 'Would it be better for the user to perform a second feedback iteration than to look at the later retrieval of the first iteration?' The latter question is not answered by direct comparison of residual collection measures, because it is equivalent to the question 'Is the second iteration query closer to the optimum query for the second iteration residual collection than the first iteration query is?' Thus residual collection evaluation must provide the option of re-evaluating the performance of queries used for previous searches in the residual collection constructed for a later search. This re-evaluation is not difficult if the ranks assigned to relevant documents by earlier iterations are saved. To calculate the performance of the first iteration query in the second iteration residual collection, for example, all relevant documents presented for feedback on the second iteration are deleted from the saved list (or not saved) and the lowest rank assigned by the first iteration to a document presented for second iteration feedback is subtracted from each rank assigned to a relevant document by the first iteration. (This 'lowest rank' is the number of documents fed back on the second iteration.) The adjusted ranks of relevant documents are used to calculate all measures and the size of the second iteration residual collection is used for averaging.

In spite of the greater complexity of calculation, residual collection evaluation is recommended for future experiments with relevance feedback, because it directly answers a hitherto uninvestigated question considered most relevant to the evaluation of feedback strategies. Moreover, the viewpoint presented by residual collection evaluation is appropriate to other areas of information retrieval research. Some of these areas are discussed in later sections of this report.

An evaluation method has been proposed that avoids the controversy between feedback and ranking effect. The document collection is randomly separated into two halves here called subset one and subset two. The feedback and query alteration are performed based on subset one, then the original query and all altered queries are tested on the documents of subset two. Subset two thus performs the function of a residual collection not containing documents used for feedback. This evaluation method, here called test collection evaluation, has the same advantages as residual collection evaluation. It shares two residual collection evaluation disadvantages. First, both methods require that previous queries be re-evaluated in the residual or test collection. Of course, residual collection evaluation provides several test collections while test collection evaluation supplies only one. Second, both methods may encounter queries with no relevant documents in the residual or test collection and that therefore must be dropped form the evaluation. This condition is less likely in residual collection evaluation . because the residual collection is as large as possible. Test collection evaluation has one advantage and two disadvantages as compared to residual collection evaluation. It has the advantage of using the same collection to test all queries, while residual collection evaluation uses a different residual collection for each query, causing the problems stated earlier. On the other hand, residual collection evaluation has the advantage of providing the largest possible collection for test purposes in every case. Also, residual collection answers directly a question not answered by test collection evaluation, which is the Hall and Weiderman question 'How many new relevant document are retrieved as a result of feedback.' This question requires the use of different test collections for each query, because the 'new relevant documents' available to each query in the document collection are different.

Test collection evaluation thus provides a method of evaluation distinct from residual collection evaluation and having several advantages over the evaluation methods previously employed. Its major disadvantage is the need to halve the size of the experimental collection. However, test collection evaluation promises to be a useful technique for providing direct comparison between varied feedback strategies, search techniques, vector constructions, and other dissimilar experiments conducted over a long period and using the same large document collection divided into the same subsets. It parallels the commonly accepted procedure of providing a control group and a test group for each experiment, and should be added to the evaluation methods employed in information retrieval as soon as a large enough collection is obtained.

From consideration of the problems encountered in evaluating the experiments reported, five recommendations for evaluation of relevance feedback algorithms are made. First, larger document collections with larger query samples should be obtained and statistical tests should be used to support all average results. Second, weighted recall and precision, summary measures analogous to normalized recall and precision, are recommended to attach greater significance to early retrieval than to later retrieval. Third, Quasi-Cleverdon interpolation is recommended over Neo-Cleverdon interpolation for constructing average recall-precision curves in the experimental environment of this study, and further investigation of possible definitions of 'equivalent performance' for queries having different numbers of relevant documents is suggested. Fourth, three new evaluation methods are constructed that are more appropriate for relevance feedback study than existing methods. The three methods are called modified total performance evaluation, modified feedback effect evaluation, and residual collection evaluation. Each answers a different question that is relevant to the study of interactive document feedback. Fifth, a previously suggested evaluation method, here called test collection evaluation, is distinguished from residual collection evaluation and is recommended to provide directly comparable studies of different types of retrieval and classification methods.

C. Feedback of Non-Relevant Documents

The results reported in Section VI-D and VI-E indicate that feedback of non-relevant documents provides excellent retrieval for certain queries and very poor retrieval for certain others. Although the causes of this variability are not clear from this study, promising indications for further research are found in Section VI-E. Similar investigation of subgroup properties should be conducted in larger document collections with larger query samples, because the sizes of some subgroups investigated are marginally small for the statistical test used, especially when tied observations occur. With a larger query sample, comparisons within subgroups as w^1l as between subgroups would be meaningful. Four research areas suggested by results reported in Section VI-E are listed below.

1) Three findings indicate that for the Rocchio strategy, movement of the modified query away from the original query between the first and second iterations is correlated with

poor performance, especially with poor initial search results. This direction of movement could be an effect of inadequate feedback, or it could be an attempt to compensate for a poor original query. Residual collection evaluation could determine whether the movement of the second iteration Rocchio query further from the original query results in performance improvement or performance degradation, and modified total performance evaluation could determine whether the movement is toward or away from the optimum query for all relevant documents. A difference in the results of these two evaluation methods would raise implications for multiple query strategies (discussed in Section VII-D).

2) Recall-precision curves (but not normalized measures) indicate that the queries resulting in performance degradation with the Rocchio strategy give poorer performance on the initial search and poorer performance and less first iteration improvement with the Q_o+ strategy than other queries. If this relationship holds in other collections, this type of query should be studied separately to discover the causes of this poor performance. It is possible that the relevant documents for these queries form two or more separated clusters in document space. The implications of this possibility are discussed shortly.

3) Further study of those queries that retrieve no relevant documents on the initial search should be conducted in an environment containing more such queries. The ingenuity of Steinbuhler and Aleta [13] in artificially creating the same retrieval situation by omitting retrieved relevant documents from the collection leads to valid conclusions about negative feedback, but does not provide a valid means of investigating the type of query that results in poor initial retrieval.

4) Finally, the joint relationship of number of concepts and number of available relevant documents to the performance

of positive and negative feedback strategies should be explored in several ways. Some relevant questions are:

1) Does the relationship found in the Cranfield 200 collection hold in other environments? Does it hold for residual collection evaluation?

2) Is query vector length a better predictor than number of concepts? If so, is the reported relationship caused by the failure of this study to normalize the components of the Rocchio query modification? Does the change in query vector length after feedback have some relationship to the reported phenomenon?

3) Can the number of documents retrieved on the initial search be used as an estimator of the number of available relevant documents? If so, can the system select an appropriate strategy for each query before iteration? If not, can another estimator be found that is known to the system before iteration?

4) Do the queries with many concepts and few relevant have similarities to the queries with few concepts and many relevant other than that of poor performance on the Rocchio strategy? Do these two groups differ in characteristics other than number of concepts and number of relevant? Does the Rocchio strategy fail for the same reason or for a different reason in each group?

A hypothesis is presented that explains some of the observed performance differences between the negative feedback strategies and the positive feedback strategies investigated, and is consistent with all experimental results reported. Hypothesis:

> For most queries, for every vector v contained in the set R of relevant document vectors there exists at least one vector s contained in the set S of non-relevant document

> vectors such that for some other vector r contained in R, cos(r,s) is greater than cos(v,r). Further, for a significant number of queries the prevalence of such relationships effectively prevents the retreival of some relevant documents with reasonable precision by any relevance feedback strategy that constructs only one query on each iteration.

This hypothesis states in effect that the documents relevant to a single query are usually found in two or more distinct clusters in the concept vector space, and that these clusters of relevant documents are separated from each other by non-relevant documents. Further, it states that for a significant number of queries this phenomenon will seriously interfere with the retrieval of some relevant documents regardless of the relevance feedback strategy employed. For any collection in which this hypothesis is true, all relevance feedback algorithms tested in this study are inappropriate for a significant percentage of retrieval requests. Algorithms constructing more than one query on each feedback iteration are necessary in such an environment.

The anomalous results of the reported comparisons of positive and negative feedback support the conclusion that the stated hypothesis is true in the Cranfield 200 collection. Because this collection is a carefully chosen subset of a larger collection representative of a well-defined, technical, limited subject area, this conclusion suggests that multiple query algorithms or other means of simplifying the distribution of relevant document vectors in the vector set being searched will be needed in practical automatic retrieval systems.

The most ubiquitous indication of separated relevant clusters is the typical negative feedback drop in normalized recall on the first iteration. This decrease in normalized recall is coupled with a rise in total performance normalized precision and in both total performance and feedback effect precision at all recall levels. As was stated earlier, this combination of measurements indicates that the Rocchio strategy raises the ranks of some high-ranking relevant documents and lowers the ranks of other low-ranking relevant documents. In fact, Figure 21 shows that both negative feedback strategies tested are superior to positive feedback within the top 8% of the ranked collection, but greatly inferior in recall after 20% of the collection has been scanned. Both negative feedback strategies maintain a slight first iteration advantage in precision. The early negative feedback advantage is evident in spite of the freezing of the top ranks for feedback effect evaluation. Therefore it is evident that the ranks of high-ranking unretrieved relevant documents are being raised more by negative feedback than by positive feedback, but that the ranks of low-ranking relevant documents are being lowered much more by negative feedback to cause a precipitous drop in the average recall difference between negative and positive feedback. Rephrasing the previous sentence in terms of query vector movement, the use of nearby non-relevant documents as well as nearby relevant documents for feedback causes the query to move closer to other nearby relevant documents than to nearby non-relevant documents, but at the same time to move farther from relevant documents already relatively distant than from relatively distant non-relevant documents. Such a description of vector position change is easiest to explain by assuring the presence of non-relevant documents between the 'nearby' and 'distant' groups of relevant documents. In particular, Figure 21 might indicate that the non-relevant documents used for feedback are between the retrieved relevant documents and the 'distant' relevant documents, and actively push the modified query away from low-ranking relevant documents.

Several characteristics of the groups of queries chosen by strategy in Section VI-E are consistent with the hypothesis of separated relevant clusters. The criterion for selection of the Q_o+ and Rocchio groups is retrieved within i N documents on the i 'th iteration, ranging from one to three. The differences in normalized recall and precision between these groups are caused by the Rocchio strategy. The normalized measures for the Q_o+ strategy are not significantly different between the Q_o+ group and the Rocchio group, but in the Q_o+ group the Rocchio strategy degrades performance and in the Rocchio group it improves performance. In addition, the recall-precision curves of Figure 28 and 29 show that the initial search curve of the queries in neither group is the highest, while the initial search curve of the Q_o+ group is the lowest at low and medium recall levels.

The findings summarized above can be explained in terms of the hypothesis as follows: The strategy differences are caused by the Rocchio strategy because it uses negative feedback to discriminate better between the retrieved relevant and retrieved non-relevant documents. If the retrieved non-relevant documents are badly positioned relative to some unretrieved relevant documents, the Rocchio strategy specifically moves the query away from these relevant documents while the Q_o+ strategy merely moves toward retrieved relevant documents. Because the Rocchio strategy discriminates better between relevant clusters represented by feedback, it can have inferior retrieval only if the Rocchio query is pushed away from all relevant documents by negative feedback (only 3 cases) or if it moves away from many relevant documents in order to better discriminate between a relatively small relevant cluster and nearby non-relevant documents. Since both strategies use the same relevant documents for feedback on the first iteration, only the hypothesis of separated relevant clusters can explain how negative feedback moves the query further away from relevant documents than positive feedback. Since the described movement of the Rocchio query cannot occur if the original query retrieves relevant documents that represent the largest relevant clusters, the Q_o+ group has poor initial search performance. The Rocchio strategy has better early retrieval if the original query retrieves documents representing the largest clusters without retrieving the entire cluster, that is, if the original query is good but not optimal. If the original query is already near-optimal both strategies will have equally good performance. This reasoning explains the high average performance of queries in neither group at all recall levels. The Rocchio group and the Q_o+ group are equally low in initial precision at high recall, indicating that in both groups the original query is far from some separated clusters of relevant documents. Also, in Figure 30 the Rocchio strategy in the Rocchio group has lower normalized recall on the first iteration than the Q_o+ strategy in the Rocchio group, indicating that even when the Rocchio strategy provides better early retrieval, it still lowers the ranks of distant relevant documents relative to the ranks assigned by the Q_o+ strategy to these documents. If no queries in the Rocchio group had separated clusters of relevant documents, the higher early retrieval of the Rocchio strategy would lead to higher normalized recall also.

The first statement of the hypothesis is thus consistent with reported results. The stronger statement that the presence of separated clusters of relevant documents will prevent full retrieval for a singificant number of queries with any single-query feedback strategy is supported by the low average precision at 100% recall. The highest reported total performance average precision at full recall is 45%, and the highest feedback effect average precision is 33%. To further support this stronger claim, the performance of the individual queries for the Q_o+ and Rocchio strategies are examined. Twenty-eight of the 42 queries display performance indicating the presence of separated relevant clusters; that is, as the correlation of one relevant document rises, the rank of another relevant document falls. Twenty-two of these queries display this behavior with the Q_o+ strategy, proving that the phenomenon is not caused only by negative feedback. Eighteen queries seem seriously affected by the presence of separated relevant clusters. For 12 of these queries, one or more relevant documents are not retrieved within 20% of the collection, or 40 documents, by either positive or negative feedback after three feedback iterations. The average precision at 100% recall for these queries is 7.4% when the best strategy is chosen for each query. Six more queries have at best less than 20% precision at full recall and must search at least 10% of the document collection. The average precision of these six queries at full recall is 16% when the best strategy is used for each query. The average rank of the last relevant document retrieved by these queries is 73.5 at best. By contrast, the average precision at full recall of the remaining 24 queries is 52.7% and the average rank of the last relevant document is 10.4, at best. When the worst strategy is chosen for each query the average final precision only drops to 45.2%. The ccnclusion that either relevance feedback strategy is inappropriate for 43% of the query sample is inescapable.

Examples of the retrieval behavior caused by separated clusters of relevant documents are given in Figures 36, 37, and 38. Query 9 has only two relevant documents, but these are separated from each other so that as one rises in rank, the other falls. Positive feedback retrieves one of these relevant documents and negative feedback retrieves the other. Figure 37 gives a more complex example. The Q_o+ strategy uses only document 173 for feedback, thereby raising the ranks of five relevant documents and lowering that of document 174. The second Q_o+ iteration provides no feed-

Query 9:	Q_o+ Strategy				Rocchio Strategy		
Rank	Iteration			Rank	Iteration		
	0	1	2		0	1	2
1	179	179	179	1	179	179	179
2	112	112	112	2	112	112	112
3	39	39	39	3	39	39	39
4	42	42	42	4	42	42	42
5	181	181	181	5	181	181	181
6	45	45	45	6	45	25	25
7	62	62	62	7	62	71	71
8	116R	116R	116R	8	116R	41	41
9	97	97	97	9	97	64	64
10	188	188	188	10	188	3	3
11	31	31	117	11	31	85	98
12	57	57	3	12	57	88	178
13	117	117	2	13	117	23	82R
14	2	2	158	14	2	101	160
15	25	25	185	15	25	17	101
33	82R	82R	0	16	0	82R	0
42	0	0	0	18	0	116R	0
47	0	0	82R	20	0	0	116R
				21	0	0	0
				33	82R	0	0

An Example of an Individual Query

With Separate Clusters of Relevant Documents

Q_o+ and Rocchio Strategies

Figure 36

Query 30:	Q_o+ Strategy					Rocchio Strategy			
Rank	Iteration				Rank	Iteration			
	0	1	2	3		0	1	2	3
1	39	39	39	39	1	39	39	39	39
2	173R	173R	173R	173R	2	173R	173R	173R	173R
3	188	188	188	188	3	188	188	188	188
4	42	42	42	42	4	42	42	42	42
5	7	7	7	7	5	7	7	7	7
6	199	156	156	156	6	199	176R	176R	176R
7	41	41	41	41	7	41	27	27	27
8	23	44	44	44	8	23	97	97	97
9	30	199	199	199	9	30	156	156	156
10	156	23	23	23	10	156	101	101	101
11	178	176R	30	30	11	178	49	96	96
12	181	101	178	178	12	181	134	141R	141R
13	44	118	101	101	13	44	96	44	44
14	131	27	181	181	14	131	31	89	89
15	73	172R	172R	172R	15	73	118	118	118
19	0	0	176R	0	19	0	0	172R	172R
23	172R	0	0	0	20	0	172R	0	0
25	174R	0	0	0	22	0	141R	0	0
27	0	0	0	176R	23	172R	0	0	0
28	0	141R	0	0	25	174R	0	0	0
30	0	0	174R	0	54	0	174R	0	0
32	0	0	0	141R	67	0	0	171R	0
35	0	0	0	171R	69	0	0	0	171R
39	0	174R	141R	0	71	176R	0	0	0
69	0	171R	0	0	103	0	171R	0	0
71	176R	0	0	0	111	0	0	0	174R
77	0	0	0	174R	112	0	0	174R	0
80	0	0	171R	0	115	0	0	175R	0
109	0	0	0	175R	121	0	0	0	175R
118	0	175R	0	0	130	0	175R	0	0
123	0	0	175R	0	198	141R	0	0	0
198	141R	0	0	0	199	171R	0	0	0
199	171R	0	0	0	200	175R	0	0	0
200	175R	0	0	0					

An Example of Complex Retrieval Behavior

Figure 37

Query 3:	Q_o+ Strategy				Rocchio Strategy		
Rank	Iteration			Rank	Iteration		
	0	1	2		0	1	2
1	179	179	179	1	179	179	179
2	42	42	42	2	42	42	42
3	112	112	112	3	112	112	112
4	39	39	39	4	39	39	39
5	117	117	117	5	117	117	117
6	181	181	181	6	181	4R	4R
7	57R	45	45	7	57R	71	71
8	45	57R	57R	8	45	57R	57R
9	152	152	152	9	152	30R	30R
10	62	62	62	10	62	32R	32R
11	182	182	31R	11	182	182	31R
12	153	153	4R	12	153	152	200
13	31R	31R	182	13	31R	43	189
14	43	43	30R	14	43	3	184
15	116	116	189	15	116	199	34
17	0	0	0	20	30R	0	0
20	30R	30R	32R	23	32R	0	0
23	32R	32R	0	25	4R	0	0
25	4R	4R	0	27	0	31R	0
124	33R	33R	0	36	0	0	33R
181	0	0	33R	85	0	0	0
195	0	0	0	118	0	33R	0
				124	33R	0	0

An Example of Good Rocchio Performance
On Separate Clusters of Relevant Documents

Figure 38

back, so the original query is increased in weight, lowering the ranks of four relevant documents and raising document 174. Feedback of document 172 raises three of the lowered relevant documents, further lowers document 176, and lowers document 174 again. The movement of the Q_o+ query vector is not consistent in direction, and little overall improvement in performance is accomplished. Negative feedback achieves better early retrieval by retrieving document 176 on the second iteration. All unretrieved relevant documents except the obviously separated document 174 rise in rank after the first and second iterations. However, after retrieval of 141 and 172 the ranks of 171 and 175 are lowered and that of 174 is raised slightly. In Figure 38 the Rocchio query moves immediately to a cluster of relevant documents including 4, 30, and 32, using only negative feedback. Document 57 drops slightly in rank and document 31 drops considerably. Retrieval of document 57 by positive feedback raises document 31, but is much less effective than negative feedback in raising 4, 30, and 32. Feedback of documents 4, 57, 30, and 32 to the Rocchio strategy is needed to raise the ranks of documents 31 and 33 at the same time; in two other cases the ranks of these two documents change in opposite directions.

The inconsistent changes in rank from one iteration to the next displayed in these three figures are typical, and indicate that neither the Rocchio nor the Q_o+ strategy is optimal in the experimental collection.

In summary, four areas of future research are recommended involving feedback of non-relevant documents. Queries retrieving no relevant documents on the first iteration should be studied, the relationship between the correlation of the modified query to the original query and performance should be determined, and the joint relationship of query size and number of relevant documents to positive and negative feedback differences should be explored.

A hypothesis explaining the observed performance differences between positive and negative feedback is presented, and evidence of its validity is found in the reported results. Many queries have separated clusters of relevant document vectors, and are modified by both positive and negative feedback algorithms in such a way as to make early retrieval of some relevant documents impossible. The conclusion that all strategies tested in this study are inappropriate to this retrieval environment because of the prevalence of queries having separated clusters of relevant documents is supported by investigation of individual queries. In Section VII-D, a strategy more appropriate to the environment of this study is proposed. Study of the relative distribution of the vectors describing relevant documents in other collections is recommended.

D. Partial Search and Multiple Query Algorithms

All relevance feedback algorithms evaluated in this study require a search of the entire document collection for each iteration. In a document collection one hundred times as large as the experimental collection, several full searches per query would be prohibitively expensive and time-consuming on present computers. Since collections of 20,000 documents or more are often encountered in practice, the use of partial search strategies is imperative. No attempt to investigate partial search algorithms is made in this study because the subdivisions of the collection would be far too small to be realistic. However, some of the discussion earlier in Section VII can be extended to partial search algorithm experimentation.

In this section, prior investigations of partial search algorithms in the Cranfield 200 collection are briefly reviewed. Next the evaluation of cluster search techniques is discussed and measures for the evaluation of partial searches and of the general usefulness of a clustering scheme are suggested. Then a new cluster search algorithm is suggested, based on the hypothesis stated in the previous section.

The hypothesis discussed and supported in Section VII-C strongly suggests that an algorithm employing more than one query is needed in the environment of this study. A cluster search algorithm employing relevance feedback and constructing a separate query for each selected cluster is presented in detail. Then an earlier study of a query splitting algorithm in the Cranfield 200 collection is briefly reviewed. Suggestions for other multiple query algorithms involving relevance feedback are made based on the conclusions of Section VII-C. Finally the clustering of previous requests, suggested by Salton, and the modification of document descriptions based on user requests and relevance judgments are discussed as possible solutions to the problems presented by the hypothesis of that section.

Rocchio [9] proposes an algorithm that assigns every document vector to one or more clusters of similar document vectors, using the distance function that is employed for retrieval in the collection. He suggests that the centroid vectors of the clusters formed by the algorithm be used as a pseudo-collection for a preliminary search, and that only the document vectors in those clusters with centroids nearest the query vector be examined for retrieval. (Hereafter the phrase 'the cluster nearest a query' refers to the cluster with its centroid vector nearer to the query vector than the centroid vector of any other cluster.) Rocchio's clustering algorithm has the following advantage over other methods of partitioning the documents of a collection.

a) clusters are generated automatically.

b) The cluster size and number of clusters in the collection can be controlled by parameters.

c) A document may be assigned to more than one cluster. This feature allows for documents concerning more than one subject, and may increase the probability that all documents relevant to a query can be found by searching only a few clusters.

Results of two studies of Rocchio's algorithm in the Cranfield 200 collection are here summarized. Salton [22] reports search results after using Rocchio's algorithm to cluster the ADI regular thesaurus vectors and the Cranfield 200 word stem vectors. At all attainable recall levels, precision is lower for the cluster searches than for the full search, except after the 6 clusters nearest the query (30.9% of the document vectors) are searched in the Cranfield 200 collection. Salton concludes that a significant reduction in processing time is achieved with relatively little precision loss (maximum 15%), and recommends cluster search as a money-saving possibility for users not requiring high recall. He also suggests that the queries submitted by previous users be clustered in collections in which either the document space or the subject classifications are subject to rapid change. He proposes a general search algorithm combining cluster search with relevance feedback and other techniques. This algorithm first performs a query cluster search, and then chooses progressively more accurate techniques as needed to retrieve relevant documents. Document vectors for relevance feedback may be selected from the results of a full search or of a partial search.

Leech and Matlack [23] compare the results of clustering the Cranfield 200 regular thesaurus vectors with those of clustering the Cranfield 200 word stem vectors. They conclude that in the regular thesaurus vector collection clusters of a size equivalent to five percent of the collection size are optimal, but that larger clusters are needed in the word stem collection. A cluster search of the thesaurus collection gives better recall-precision results than a cluster search of the word stem collection except for large clusters at less than 28% recall. The recall-precision curve generated by searching the two clusters nearest the query using the best set of clusters formed from the thesaurus vectors is slightly higher than the full search recall-precision curve at all recall levels. This result does not indicate that searching two clusters provides better precision than a full search at all recall levels. Because all relevant documents may not be found in the nearest two clusters, some recall levels cannot be achieved for some queries. Extrapolated values for these unattainable recall levels are nevertheless averaged into the recall-precision curve. The average 'recall ceiling', that is the average value of the highest attainable recall level for each query, is 53.4% for the two nearest clusters. On the average, 9.6% of the collection is scanned to obtain this recall ceiling. It appears that performance improvement is achieved for low recall levels and search cost is significantly reduced by a two-level search of Rocchio clusters formed from the Cranfield 200 regular thesaurus vectors.

The evaluation of partial search algorithms presents several problems. In the previous paragraph, difficulty is encountered in interpreting a comparison of performance measures obtained from a partial search and from a full search. In the SMART system at Cornell [24] the full number of relevant documents is used to calculate all recall and precision measures. Thus the evaluation of partial search results is intended to answer the question 'How well can a partial search retrieve all relevant documents from the total collection?' This question is answered incompletely by partial search recall-precision curves, because these curves give no indication that some recall levels cannot be achieved for some queries, and in fact extrapolated precision values are assigned to unattainable recall levels. The SMART system reports the average recall ceiling for every partial search to give some indication of the recall levels that can be attained. However, because this reported recall ceiling is an average value, some queries may achieve higher recall levels and some may not achieve the ceiling level. Salton [22] and others report partial search results as recall-precision curve segments. For a search of the nearest n clusters, only the curve segment from the recall ceiling of the search of n-1 clusters to the recall ceiling of the search of n clusters is graphed. This type of graph recognizes the recall ceiling problem inappropriately, because some achieved recall levels below and above the bounds of the reported curve segment are ignored. The implied assumption that the performance of the n-cluster search is the same as that of the n-1 cluster search up to the n-1 cluster recall ceiling is false, because all documents in the n clusters are ranked together by the search, so all documents from the n^{th} nearest cluster are not necessarily retrieved at the bottom of the ranked list. Leech and Matlack [23] report the full recall-precision curve for each partial search and indicate the recall ceiling as a point on the curve. Their solution of the evaluation problem is better than that of reporting curve segments, because no attained recall levels are ignored. However, the problem of distinguishing attainable from unattainable performance remains.

By extension of the discussion of recall-precision interpolation in Section VII-B, the SMART rightward extrapolation method for partial search recall-precision curves defines an equivalence relation between partial search performance and full search performance at all recall levels not attained by the partial search. The results of performance comparisons between partial and full searches are largely dependent on the equivalence relation defined by the choice of a rightward extrapolation method. The definition of an equivalence relation between queries with different numbers of relevant documents by precision interpolation at unattained recall levels seems reasonable. The definition of an equivalence relation between a partial search and a full search of the same query by precision extrapolation at unattainable recall levels is less easily justified. A possible alternative is to refuse to extrapolate to the right, but instead to average at each recall level only queries that attain equivalent or higher recall. For each point on the recall-precision graph of a partial search, the number of queries attaining that recall level would be reported.* This alternative as proposed above eliminates doubt of the validity of partial search recall-precision curves at high recall levels, but still does not provide direct performance comparison to a full search curve because different queries would be used for averaging the high recall points. It would be possible to construct for each partial search curve a matched full search curve that averages at each recall level the full search precision of the queries attaining equal or higher recall on the partial search. This second alternative gives a directly interpretable comparison between full and partial search recall-precision curves by failing to report all full search results. Each of the proposed partial search recall-precision curves illuminates the experimental situation from a different angle; all three curves may be needed in some

*The SMART system now reports the number of queries achieving a given recall or less without extrapolation so that the extent of leftward extrapolation at low recall levels can be estimated.

cases to provide even and unshadowed lighting.

Though partial search and full search recall-precision performance is difficult to compare, the document curves provide a direct answer to another question relevant to partial search strategies: 'What performance has been achieved by each search after the same percentage of the total collection has been scanned?' The document curves report recall and precision at several possible cut-off ranks, so they can be used to answer questions of the form 'Is it better to give the user all n documents in the nearest cluster or the top n documents of the full search?' These curves provide direct and meaningful comparability between partial and full search strategies and between alternative partitions* of the same collection.

In the preceding discussion the distinction between attained performance and attainable performance arises. Recall ceiling is a measure of the highest recall attainable in a cluster, though that recall may be attained after only part of the cluster has been searched. Since different multilevel search strategies might use the same set of document clusters, attainable performance may provide a better indication of the general usefulness of a given partition of the document collection than the performance attained by one particular search strategy. In a study of clustering in the ADI collection, Grauer and Messier [25] use three measures that are not related to the search strategy employed, but that may be used jointly to indicate the utility of a given partition of a document collection. One of these measures is recall ceiling, an indicator of attainable performance. The other two measures are called 'user percentage scanned' and 'machine percen-

*Hereafter, a set of clusters such that their union includes all document vectors in a collection is called a partition of the collection.

tage scanned'. These three measures are defined below in terms unrelated to any specific search strategy:

Let N = number of documents in the collection

C = number of clusters in the partition being evaluated

Q = number of queries in a representative query sample used for evaluation

Then given a number of clusters n and a query i, let

R_i^n = the number of documents relevant to query i in the n clusters closest to query i.

D_i^n = the number of documents in the n clusters closest to query i.

R_i = the total number of documents relevant to query i in the collection.

Then

$$\text{recall ceiling } (n) = \frac{1}{Q} \sum_{i=1}^{Q} \frac{R_i^n}{R_i}$$

the average ratio of the number of documents in the nearest n clusters to the total number of relevant documents.

$$\text{user percentage scanned } (n) = \frac{1}{Q} \sum_{i=1}^{Q} \frac{D_i^n}{N}$$

the average ratio of the number of documents in the nearest n clusters to the collection size.

$$\text{machine percentage scanned } (n) = \frac{1}{Q} \sum_{i=1}^{Q} \frac{D_1^n + C}{N}$$

the average ratio of the number of vectors searched by a two level partial search of the nearest n clusters to the number of vectors searched by a full search of the document collection. Machine percentage scanned is a system-independent indicator of the search time or search cost of a partial search relative to a full search. (For a partial search strategy involving a different number of vectors, the machine percentage scanned strategy could be changed to indicate the changed search cost.)

As Grauer and Messier [25] point out, these three measures do not provide direct comparability between alternative partitions of the collection. The type of question asked of these measures is 'If partition A yields an average recall ceiling of 25% for the nearest two clusters, and these two clusters include 30% of the collection, while partition B yields an average recall ceiling of 35% and the nearest two clusters include 40% of the collection, which partition is better?' An answer to this type of question is here proposed that leads to two directly comparable and meaningful measures of the utility of alternative partitions of a document collection. The first measure is based on the notion of generality number used by Cleverdon and Keen [18]. The generality number of a collection is the ratio of the average number of documents relevant to a query (calculated from a representative query sample) to the number of documents in the collection. In a collection with a higher generality number, precision is generally higher [18]. The goal of a two level search using a partition of the documents collection is to find the same relevant documents by searching fewer document vectors. Therefore, the partition used should effectively increase the generality number of the searched collection for each query, that is, it should select for each query a subset of documents containing more relevant documents in proportion to the subset size than the entire collection contains in proportion to its size. The 'generality factor' defined below is a strategy-independent measure of the extent of which a given partition of the document collection accomplishes this aim:

$$GF\ (n) = \frac{1}{Q} \sum_{i-1}^{Q} \frac{R_i^n}{D_i^n} \div \overset{R}{\frac{Ri}{N}}$$

the average factor by which the proportion of relevant documents to searched documents is multiplied by clustering the document vectors and selecting the n clusters closest to each query.

A second measure, called the cost factor, is based on the comparative cost of a partial search to a full search, as is the machine percentage scanned. The cost factor is defined with the same structure as the generality factor:

$$CF\ (n) = \frac{1}{Q} \sum_{i=1}^{Q} \frac{R_i^n}{D_i^n + C} \div \frac{Ri}{N}$$

the average factor by which the proportion of relevant documents to searched vectors is multiplied by clustering the document collection and selecting the n clusters closest to each query. Note that a cost factor greater than 1 indicates that the cost of a partial search is lower than that of a full search.

The generality factor and cost factor each define an equivalence relation between two partitions that may achieve different recall ceilings with document subsets of different sizes.

It is interesting to note that a re-evaluation of the Grauer and Messier [25] results using estimates of the generality factor and cost factor measures clearly shows that clustering the 82 document ADI collection isn't worth the trouble. Only a few runs have generality factors as high as 2.0 and for these runs the cost factor is less than one, indicating a search cost greater than that of a full search. By contrast, the Leech and Matlack [23] clusters in the Cranfield 200 collection yield estimated generality factors from 3 to 9 and estimated cost factors from 1.5 to 2.6. In larger collections, the difference between the generality factor and the cost factor of a run would probably be smaller. For comparison of different partitions of a document collection, it is suggested that for each partition n (number of clusters searched) be increased until a recall ceiling of 100 is reached, and that the generality factor and/or the cost factor be plotted against the recall ceiling for each possible n.

One further suggestion for cluster search algorithms can be made on the basis of the hypothesis stated in the previous section. The Rocchio clustering algorithm has been used with only one two level search strategy, that of choosing the nearest n clusters and ranking in one search operation all documents in these n clusters. This procedure may not be ideal for most queries. If n equals 2, for example, the centroid of the second cluster may be farther from the original query than that of the first cluster, indicating that in general the documents in the second cluster are farther from the original query than those in the first cluster. It is possible, therefore, that some if not all relevant documents in the second cluster are retrieved later in a joint search of both clusters than are some non-relevant documents in the first cluster. If all relevant documents form a single cluster in the unpartitioned document space, this problem does not occur. However, according to evidence in Section VII-C the relevant documents are usually separated from each other in the document space.

If each Rocchio cluster is searched separately, however, the user's query is only required to separate the relevant documents in each cluster from the nonrelevant documents in the same cluster, rather than to separate all relevant documents from all nonrelevant documents in the clusters searched. Within a single Rocchio cluster, the occurrence of separated clusters of relevant documents might be less evident than in the full collection. In fact, for some queries each separated cluster of relevant documents might be found in a different Rocchio cluster, thus providing within each cluster a retrieval situation that a single query can resolve.

The foregoing argument suggests a cluster search algorithm that ranks each document relative to other documents in the same cluster, and retrieves the highest ranking documents from each cluster searched. Construction of such an algorithm presents a strategic problem - in what order are the documents to be presented to the user? This problem can be rephrased in terms of performance evaluation - given the ranks of all documents relative to other documents in the same cluster, how are ranks to be assigned to all documents in the collection for comparison with other strategies not using the same partition of the document space? The simplest method is to assign the first n ranks in rotation to the first document of each cluster searched, and so on. This 'rotation' method of ranking all documents makes no special provision for clusters of different sizes or for clusters that might be expected to contain more relevant documents. Modified rotation methods might be constructed that automatically assign more high ranks to documents in the larger clusters, or to documents in the clusters nearer to the original query. Another alternative worth testing is to rank all documents according to the distance of each document from the original query <u>relative</u> to the distance of the cluster containing that document from the query. Coefficients providing this ranking could be obtained by subtracting from the correlation coefficient of each document the coefficient of the centroid of the cluster containing that document. Because the Rocchio clustering algorithm allows cluster overlap, an overall ranking method must define the rank of a document appearing in more than one cluster. Such a document might be assigned the highest of the possible ranks, or perhaps the rank assigned by its position in the cluster nearer the original query.

Investigation to determine the most appropriate ranking method for combining separate cluster searches should be conducted. Residual collection evaluation, defined in Section VII-B, is a valuable tool for such a study. If each cluster is evaluated separately, the efficiency of the query in separating the relevant documents from the nonrelevant documents within each cluster can be determined, and can be compared to the ability of the same query to separate all relevant from all nonrelevant documents in the searched clusters. With this information the feasibility of separate cluster searches, and of some of the possible ways of combining them, can be estimated.

It is evident from Section VII-C that a multiple query algorithm is usually needed to separate all relevant documents from all nonrelevant documents in the full collection. The preceding discussion indicates that a partial search algorithm might take advantage of the possibly simplified retrieval task within each selected cluster of documents by searching each cluster separately. However, even if only one query is required for ideal retrieval within each cluster, it is very unlikely that the same query can accomplish this task for every selected cluster. A combination of relevance feedback and cluster search techniques is indicated, to tailor a specific query for each retrieval situation encountered in processing a user's request.

The partial search relevance feedback technique proposed here treats each cluster as a separate document space, and could use any relevance feedback algorithm to construct a query intended to separate relevant from non-relevant documents within that cluster. Any technique using relevance feedback to construct a single query for each document cluster on the lowest search level of a partial search algorithm is herein called 'cluster feedback'. A detailed description of a general two-level cluster feedback algorithm is presented below. Two considerations in defining this combined algorithm have not been encountered in the cluster search or relevance feedback strategies discussed in this report. The first is the possibility of using relevance feedback to select additional clusters to be searched, seen in steps 6-8 below. The second is the economic need to abandon the search of unproductive clusters as soon as possible. The methods of discarding queries that are incorporated into the suggested cluster feedback algorithm could also be used for full search relevance feedback and for the multiple query feedback algorithms discussed later in this section.

The detailed algorithm description below includes some explanation and lists alternative strategies for critical steps. Figure 39 displays an abbreviated algorithm description in flowchart notation.

A Two-Level Cluster Feedback Algorithm:

Step 1 Search all cluster centroid vectors and select the clusters closest to the original query q_o.
The number of clusters selected might be the same for each information request. However, other possibilities should be investigated, such as selecting all clusters with centroid correlations to q_o greater than some Π, or choosing a cut-off after the nearest centroid c_i such that $\cos(c_i, q_o)-\cos(c_{i+1}, q_o)$ is greater than some Δ.

Step 2 Search all clusters selected in Step 1, selecting the documents nearest to q_o.

The number of documents to be selected from a given cluster might be partly determined by the number of clusters selected, by cluster size, and/or by the distance of the cluster from the query. The variable feedback and combination feedback techniques used for full search relevance feedback might be employed, but a maximum number of documents to be retrieved from a given cluster should be defined to avoid unnecessary feedback from unproductive clusters.

Step 3 Obtain user relevance judgments on the documents selected from all clusters.

The remaining steps may be iterated as indicated until the user is satisfied with the search results or until all clusters have been discarded as candidates for further search.

Step 4 Construct a new query for each cluster that has been searched, using the original query, any other query or queries that have been used to search that cluster, and the documents retrieved from that cluster.

Though any relevance feedback algorithm could be used in Steps 4 and 6, the Rocchio algorithm is suggested for use with Rocchio clusters and the cosine correlation coefficient.

Step 5 Discard any query constructed in Step 4 that contains fewer than k concepts. Also discard the associated cluster.

This step is optional, and is needed only when negative feedback is used in Step 4. See Step 10b for a related method of discarding unproductive queries.

Step 6 Construct a new centroid search query using the original query, any previous centroid search query, and the documents retrieved from all clusters.

Steps 6-8 optional. The utility of this process in retrieving additional clusters containing relevant documents should be investigated. An experimental system should include the possibility of omitting steps 6-8 after j iterations.

Step 7 Select the clusters with centroids closest to the centroid search query of Step 6.

The numbers of clusters to be selected in this step may be determined in the same manner or in a different manner than in Step 1. The number of additional clusters selected might be allowed to influence the number of documents to be selected from each cluster in Steps 8 and 9.

Step 8 Search the cluster just selected by Step 7 using the centroid search query constructed in Step 6.

Step 9 Search all other clusters that have not been discarded.

The number of documents retrieved from each cluster might be determined in the same manner or in a different manner than in Step 2.

Step 10 Discard any query and associated cluster that does not meet the following criteria as a result of Step 9.

a) All documents in the cluster have been retrieved.

Present the documents last retrieved in Step 9 to the user but do not ask for relevance judgments.

b) Of all unretrieved documents in the cluster, the span between the highest and lowest correlation is less than some d.

This condition indicates that the query is too general to select more documents from the cluster, since all remaining documents are about the same distance from the query. Checking for this condition may make Step 5 unnecessary.

c) The highest correlation of any unretrieved document in the cluster with the original query is less than c.

This condition indicates that the query is too specific, because the cluster contains no more documents similar to it. The later discussion of multiple query algorithms suggests alternate queries for this condition.

Step 11 Obtain relevance judgments on all documents selected in Steps 8 and 9 except those documents selected from clusters discarded in Step 10.

Step 12 Discard any query and associated cluster that has retrieved no new relevant documents in M iterations.

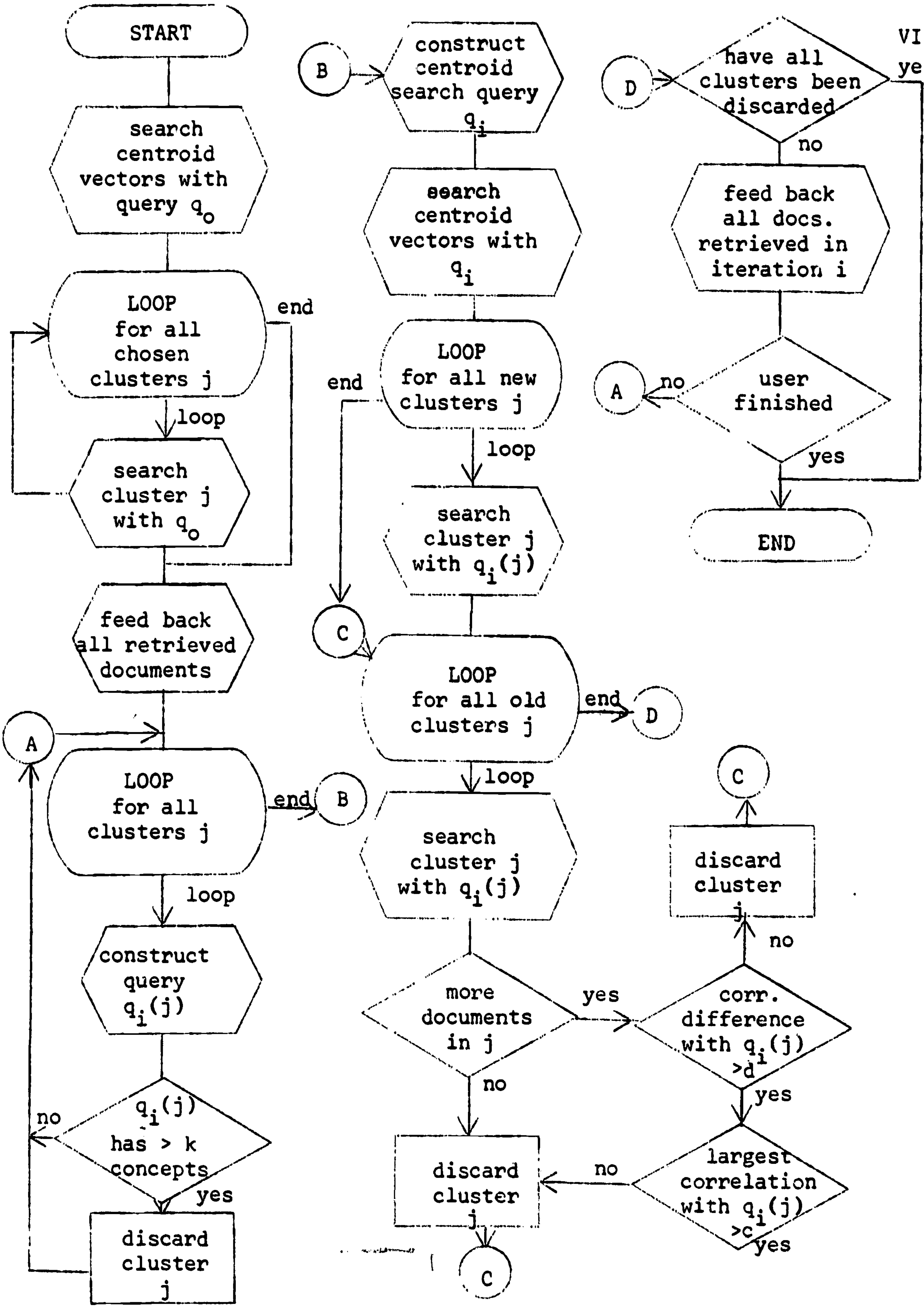
Step 12 may not be needed if all conditions suggested in Step 10 are checked.

Return to Step 4 to search all clusters that have not been discarded, including those new clusters last selected by Step 8, if any.

Compared to full search relevance feedback, the above algorithm will provide improved retrieval at decreased search cost if the following conditions are true:

1. The partition of the document space must not overlap so much that more documents are processed by searching the selected clusters separately than by searching the full document collection.
2. The retrieval problem within each cluster must be simpler than the retrieval problem in the full collection. In the ideal case each cluster would require only one query for ideal retrieval.
3. The cluster selection in Steps 1 and 6 must select those clusters containing relevant documents, and must select few unproductive clusters. If unproductive clusters are selected, they must be discarded early in the iterative process.

Condition 1 can be controlled by the Rocchio clustering process. Condition 2 is likely to be true in environments similar to that of the present experiments. Investigation to determine the document vector collections, document space partitions, query types, algorithm variations, and algorithm parameters (k,c,d, etc. in algorithm description) resulting in improved performance at lower search cost should be conducted. Large document and query collections (at least 1000 documents and 500 queries)


START
search centroid vectors with query q_o
LOOP for all chosen clusters j
end
loop
search cluster j with q_o
feed back all retrieved documents
A
LOOP for all clusters j
end
B
loop
construct query $q_i(j)$
$q_i(j)$ has > k concepts
no
yes
discard cluster j
B
construct centroid search query q_i
search centroid vectors with q_i
LOOP for all new clusters j
end
loop
search cluster j with $q_i(j)$
C
LOOP for all old clusters j
end
D
loop
search cluster j with $q_i(j)$
more documents in j
yes
no
discard cluster j
C
corr. difference with $q_i(j)$ >d
no
yes
discard cluster j
C
largest correlation with $q_i(j)$ >c
no
yes
D
have all clusters been discarded
yes
no
feed back all docs. retrieved in iteration i
user finished
no
A
yes
END

A General Two-level Cluster Feedback Algorithm

Figure 39

should be used for all experiments with this algorithm.

Although the retrieval situation within each cluster is probably simplified by cluster feedback, separated clusters of relevant documents might still be encountered, particularly in large document collections divided into relatively large clusters. Therefore, it may still be necessary to investigate the possibility of constructing a query for each separated relevant cluster in a set of documents. In this report, a 'multiple query' relevance feedback algorithm is defined as a strategy that constructs more than one query to search the same set of documents on the same feedback iteration, whether that set of documents is a standard cluster or the full document collection. This definition is used to stress an important distinction between multiple query algorithms and simple cluster feedback, which constructs only one query per iteration to search each selected document cluster. Although cluster feedback constructs more than one query, it may still use the feedback algorithms based on Rocchio's assumption that all relevant documents are grouped together in the document set being searched. Multiple query algorithms are constructed to provide improved retrieval in cases when this assumption is not valid, so such algorithms require the development of relevance feedback strategies radically different from those studied previously.

The only previous investigation of a multiple query algcrithm in the SMART system uses the Cranfield 200 collection. Borodin, Kerr, and Lewis [26] study a straightforward technique for constructing multiple queries, called 'query splitting'. Whenever the relevant documents retrieved by some query q form two or more clusters that are relatively far from each other in the document space, each such cluster of retrieved relevant documents is used separately to form a new query. The two highest nonrelevant documents retrieved by q are used for negative feedback in forming each new query. If all retrieved relevant documents are near each other in the document space or if no relevant documents are retrieved, only one new query is formed using the Dec 2 Hi strategy. A retrieved relevant document is considered 'far' from another if the correlation between them is less than some constant times the average correlation of q with all documents retrieved by q on that iteration.

The algorithm described is tested on the 24 Cranfield queries that retrieve more than one relevant document on some iteration with N equal to 5. A 'user measure' table details the relative performance of each query for which the retrieval of the first 25 documents is changed by query splitting. Borodin, Kerr, and Lewis conclude that the result in this table 'favor query splitting', and add that the relative performance of their query splitting algorithm would be better in larger collections. They suggest that an additional query formed by negative feedback alone should be constructed for each iteration, and that methods of discarding unproductive queries be included in the algorithms.

The following facts can be ascertained from the data available from the experiments presented herein and the user measure table presented by Borodin, Kerr, and Lewis.

1. The early retrieval of 11 of the 24 queries is changed by query splitting (when 12.5% of the collection has been retrieved).
2. Only 4 of these 11 queries are improved by query splitting. Performance of the other 7 changed queries is degraded.
3. None of the 12 queries for which the Rocchio strategy performs more poorly than positive feedback (the Q_o+ group) are improved by query splitting. One of them is degraded.
4. Only 2 of the 18 queries seriously affected by the presence

of separated clusters of relevant documents are assisted by query splitting. Four of these queries are degraded.

The above findings contradict the conclusion of Borodin, Kerr, and Lewis, and indicate that query splitting does not solve the problem for which it was constructed. The contention of the three authors that query splitting would be more effective in a larger collection is probably true, but it is evident from the previous section of this report that there is considerable room for improvement in the Cranfield 200 collection. The failure of query splitting in the collection studied indicates that the algorithm tested is inadequate as a solution to the retrieval problems caused by separated clusters of relevant documents.

The query splitting strategy tested constructs a specific query for each relevant cluster represented by retrieved documents. However, it is probable that the queries displaying the poorest performance do not retrieve relevant documents from each separated relevant cluster. Query splitting is still based on the Rocchio assumption found invalid in this collection that the retrieved relevant documents are representative of all relevant documents. Cluster feedback as suggested earlier in this section assumes that separated relevant clusters will not seriously affect retrieval within the standard document clusters used to partition the document space. Unless this assumption can be verified in typical document collections by outstanding cluster feedback results, less optimistic strategies should also be investigated.

Two considerations uniquely characteristic of multiple query algorithms are the basis of the following discussion. The first is the possibility of constructing more than one query from relevance feedback on retrieved documents. The second is the need to construct useful queries under the assumption that the documents used for feedback are not necessarily representative of the documents remaining in the collection being searched.

Borodin, Kerr, and Lewis [26] compare the correlation between retrieved relevant documents to the average query-document correlation for a given iteration, in order to define clusters of retrieved relevant documents. One query is then constructed using each retrieved cluster. However, the discussion in Section VII-C of this report indicates that if negative feedback is used, the distance between two relevant document vectors may be less important than the presence of a nonrelevant document vector between them. Therefore, it is suggested that separated rather than separate relevant clusters be sought. Two relevant document vectors r and v would be assigned to different clusters if there exists a nonrelevant document n retrieved previously or concurrently such that cos (n,v) is greater than cos (r,v) and cos (n,r) is greater than cos (r,v). Any retrieved relevant document vector that is in this way assigned to more than one cluster could be assigned only to the cluster closest to it, or if the distances between alternative clusters are near equal, could form a separate cluster. It is clear from Figure 40 that the suggested clustering criterion is quite strong. Even though an ideal single query could retrieve all three relevant documents in the situation symbolized, each of them is assigned to a different cluster because the one nonrelevant document is closer to each than are the other two relevant documents.

If the clustering criterion suggested above is used, there is a possibility that a defined cluster of retrieved relevant documents could be broken up by a nonrelevant document retrieved on a subsequent iteration. The defined relevant cluster could be broken into smaller clusters and a new query formed from each. This re-clustering would require that the algorithm have access to all vectors of relevant documents retrieved on previous iterations and that it determine the relationship of each nonrelevant document used for feedback to each document of the relevant cluster defining the query being altered. The clustering criterion defined by Borodin, Kerr, and Lewis [26] does not raise this problem. However, the suggested choice of separated clusters of relevant documents guarantees no feedback conflicts between relevant and nonrelevant documents used to alter the same query, and also minimizes the number of queries formed by avoiding the formation of different clusters of relevant documents until such a feedback conflict is likely to occur.

It has been established that the information obtained from retrieved relevant documents may not be sufficient to retrieve all relevant documents. In the present SMART system, the only available source of information about relevant documents not represented by retrieved documents is the user's original query. A multiple query strategy should make specific use of the concepts chosen by the user to express his needs, and should ensure that none of these concepts are ignored.

There are three ways in which a concept from the original query can be effectively cancelled from the search by a strategy using positive and negative feedback. First, other concepts found in the retrieved relevant documents may have much larger weights and thus greater effect on subsequent iterations than a user-selected concept not found in the first relevant documents retrieved. Strategies using relevant documents only and giving extra weight to the original query do not solve this problem (see Section VI-B) because a concept appearing in the original query and in relevant documents will still outweigh a concept found only in the query. To give an extreme example, a query on 'the aerodynamics of birds' addressed to the Cranfield collection would quickly become in effect a question on 'aerodynamics'. By contrast, a human librarian confronted with this situation might make a special effort to find any document concerning 'birds'.

There are several ways in which similar stress on concepts not immediately found in retrieved documents can be incorporated into an automatic multiple query search. The construction of one query using only negative feedback (also suggested by Steinbuhler and Aleta [13]) would eliminate concepts found in nonrelevant documents without disproportionately increasing the weight of any concept. In some cases however, it might be more effective to pinpoint precisely the concepts that are being ignored. In the example given, any query that still contained the word 'aerodynamics' would probably not retrieve a document containing only the rare concept 'birds'. Therefore, a query constructed by subtracting the retrieved relevant documents from the original query might be useful, on the theory that if any user-chosen concepts remain after such drastic negative feedback, something should be done about it.

The second and third ways in which a user-selected concept can be ignored are caused by negative feedback. The user may employ a concept occurring in the given collection only in documents not relevant to his needs. In this case, of course, negative feedback appropriately eliminates the misleading concept. The third possibility is that an initial query concept is used in the collection with more than one meaning, and thus is relevant in one context and irrelevant in another. If nonrelevant documents containing this concept are retrieved first, negative feedback may erase meaningful information. This third case might explain the type of behavior displayed in Figure 37, and the complete query erasure occurring with all negative feedback to query 34. There is no way in the present SMART system to distinguish the third possibility from the second, to judge whether a user concept erased by negative feedback should be re-inserted or forgotten.

Adding information about concept-concept relationships to the system might enable a negative feedback algorithm to distinguish a completely irrelevant initial concept from a concept relevant in the appropriate context. For each concept in the thesaurus, a weighted list of other concepts often occurring in the same document as the given concept should be stored. For a concept used in two ways in the collection, this list would include related concepts from both possible contexts. The suggested lists could be constructed automatically from the document vectors, perhaps by using the asymmetric coefficient of concept similarity suggested by Salton for automatic hierarchy construction [3]. If a concept from the original query is eliminated by negative feedback, the query could be immediately reformulated by adding some number of related concepts to the previous query and then repeating the same negative feedback. Added concepts appropriate to irrelevant contexts might be eliminated by negative feedback while added concepts from the relevant context might be retained, preserving by context the intended meaning of the eliminated concept in the user's query. This suggestion is a variation of the strategy suggested by Kelly [14] (see Section III), but differs in that concepts closely related to eliminated initial query concepts rather than concepts occurring frequently in the collection are added to the query. The idea detailed above is here called the 'related concept Kelly strategy' and is appropriate to both single and multiple query feedback algorithms.

Another way of supplying context information to the SMART system without requiring permanent storage involves the use of negative query weights. If all document weights are positive, a concept weight below zero in a query tends to indicate that documents containing that concept are not relevant. (it may be important to realize that with negative weights, the cosine correlation coefficient as calculated by the SMART system has a range from -1 to +1 and no longer corresponds to the cosine of the angle between the vectors.) Negative query weights could be used to supply context information to differentiate possible meanings of original query concepts as follows: A 'nonrelevant context' vector is constructed by adding together the vectors of the nonrelevant documents retrieved, and then setting every concept occurring in the original query to zero. In other words, concepts contained in the original query are ignored when they appear in nonrelevant documents. Then a new search query is formed by multiplying the nonrelevant context vector by some constant less than one and then subtracting the resulting vector from the original query vector. The suggested procedure preserves all original query concepts with their original weights. However, any other concepts that occur in nonrelevant documents are given a negative weight. Thus if two unretrieved documents contain the same original query concepts with the same weights, the document having the fewest other concepts in common with retrieved nonrelevant documents is retrieved first. The use suggested above for negative query weights is here called 'selective negative weighting' and is appropriate to single query or multiple query strategies. Selective negative weighting avoid the negative weight problem encountered by Kelly [14], who did not preserve the original query and found that all cosine correlations with the new query were often negative. If all cosine correlations are negative after selective negative weighting, either the suggested multiplier constant is too large or else no documents in the collection contain the original query concepts in a context that has not been declared nonrelevant by negative feedback. Selective negative weighting may also be used when relevant documents are retrieved, in which case only the concepts from the original query are set to zero in the non-relevant context vector. In this way the deletion of superfluous or misleading concepts from relevant document feedback by negative feedback may still occur. If negative weighting is used with the related concept Kelly strategy, preservation of the original query concepts may not be necessary.

The discussion thus far has described two distinct types of queries that could be constructed by a multiple query algorithm, each type of query serving a distinct purpose. The 'specific' query is a type of query constructed from a cluster of retrieved relevant documents to retrieve similar documents. The 'general' query is constructed to retrieve documents not represented by the retrieved relevant documents. These two types of query contrast in structure as well as in purpose. The specific query is largely constructed from document abstracts and contains many concepts of high weight, at least at first. Because the specific query vector is long and contains many concepts, few document vectors will have high correlations with it. The general query is constructed from the original query and possibly from related concepts, and has fewer concepts with lower weights. Thus in the specific query discarding superfluous and misleading concepts is a prime consideration, while in the general query preserving and clarifying the meaning of the original query is the chief aim. A multiple query algorithm should therefore use different relevance feedback strategies for general than for specific queries. Some of the consideration important in altering each type of query are listed below:

1. The specific query is intended to select only relevant documents similar to a retrieved cluster of relevant documents. Therefore by Rocchio's theory, the optimum query to differentiate the retrieved cluster from all other documents, including other relevant clusters, should be approached by iteration. That is, the retrieved relevant cluster should provide positive feedback and all other documents retrieved by any query, relevant or nonrelevant, should be used for negative feedback. Since general queries have fewer positively weighted concepts, each general query should perhaps be altered only by the nonrelevant documents it retrieves.

2. The Crawford and Melzer study may indicate that the original query need not be used in constructing specific queries. By contrast, only the original query is used for positive feedback to general queries.

3. Since a specific query is intended to select documents similar to retrieved relevant documents, it should be discarded quickly if no similar relevant documents are found, or if no spread in query-document correlation is produced (if all remaining documents are roughly the same distance from the query). However, the highest query-documen' -orrelation can be fairly low without indicating that a specific query is useless as a selector. The Rocchio strategy does not necessarily construct a query that is close to relevant documents, but rather one closer to relevant than to nonrelevant documents. In the situation symbolized in Figure 40, the single query that best separates the relevant documents from the nonrelevant documents is some distance from each depicted document. Therefore specific queries should be discarded quickly on the criteria described in steps 10b and 12 of the cluster feedback algorithm presented earlier but should not be evaluated by the criterion described in step 10c. On the other hand, a general query is intended to retrieve relevant documents of types not previously encountered. The shorter and less detailed general query typically correlates more strongly with more documents

than the specific query, and has a smaller spread in query-document correlations. Therefore, the criteria of steps 5 and 10c are of greater importance in judging the worth of a general query than the criteria of steps 10b and 12.

4. Since each specific query searches a relatively small area in the document space, the immediate construction of a new specific.query for any retrieved relevant documents that cannot be added to the cluster defining the retrieving query may not be redundant. However, since a general query might retrieve a wide variety of relevant documents, any relevant document retrieved by a general query that has been previously or concurrently retrieved by any specific query should be ignored in processing the general query. Further, any relevant document retrieved by a general query that can without conflict be added to the cluster defining one and only one specific query should be so treated rather than being used to form a new specific query.

5. If the retrieved relevant cluster defining a specific query is subsequently separated by a retrieved nonrelevant document to be used for feedback to that query, a new query should be formed for each subcluster without using the previous query defined by the original cluster. In this way each specific query is defined by a single cluster of relevant documents and no relevant documents separated from that cluster are included in the positive weight of the defined query.

Although single query algorithms are known to be inadequate for retrieval, it is not clear whether all the complexities suggested in the foregoing discussion are necessary. Before further experimental effort is invested in multiple query algorithms of this type, the related concept Kelly strategy and selective negative weighting could be tested in the Cranfield 200 collection by ignoring retrieved relevant documents as Steinbuhler and Aleta do [13]. If both of these strategies prove ineffective, the usefulness of the general query in a multiple query algorithm is doubtful, unless some other means of clarifying the meaning of the original query is found.

Figure 41 details a multiple query algorithm incorporating separated clusters of retrieved relevant documents, the related concept Kelly strategy, all suggested query deletion procedures, two general queries, and distinct feedback algorithms for specific and general queries. Selective negative weighting can be incorporated into this algorithm in a straightforward manner. Some strategic choices in the construction of the charted algorithm were made to simplify programming and to reduce computer time, others were made to illustrate the possibilities for generality and may not be the most efficient choices. The level of detail presented in Figure 41 is intended to aid the serious experimenter in constructing similar algorithms, and may not be of general interest. The algorithm charted may be simple enough to be meaningfully tested in the Cranfield 200 collection. More complex algorithms should be tested with larger query collections so that several examples of each possible alternative in the algorithm are encountered.

.Any multiple query search algorithm increases the cost of retrieval by repeated searching of the same set of documents. It might therefore be economical to invest more time in procedures not taking place for each search if this off-line effort would create single query retrieval situations for most users, either in the full collection or in standard subsets of the collection. Better methods of document vector construction and clustering are thus important fields for research.

Two promising off-line techniques that might improve most retrieval situations are discussed briefly. The first of these is the clustering of

n_r = number of relevant documents retrieved this iteration

n_s = number of nonrelevant documents retrieved this iteration

$r_i (s_i)$ = a relevant (nonrelevant) document retrieved this iteration

v_i^j = a relevant document in cluster j

cluster j = a cluster of retrieved relevant documents such that there do not exist two documents v_i^j and v_k^j and a nonrelevant document s retrieved on any previous iteration by any query such that $\cos(v_i^j, s)$ is greater than $\cos(v_i^j, v_k^j)$ and $\cos(v_k^j, s)$ is greater than $\cos(v_k^j, v_i^j)$.

Q_o = user's original query

G_k = a general query constructed by the algorithm

S_j = a specific query constructed from the documents in cluster j

$n_r^k (n_s^k)$ = a number of relevant (nonrelevant) documents retrieved by query S_k or G_k on the present iteration.

$r_i^k (s_i^k)$ = a relevant (nonrelevant) document retrieved by query S_k or G_k this iteration

d_i^k = any document retrieved by query S_k or G_k this iteration

ng(k) (ng(j)) = an indicator set to 1 the first time query G_k (S_j) retrieves no relevant documents

a,b,c = control parameters for tests of query usefulness

con (v) = number of concepts in vector V

ADCON (G_k, G_l) = is a subroutine that constructs a new query by adding related concepts to query G_k for all concepts l on list L and not in query

$$(n_s^k G_k - \sum_{i=1}^{n_s^k} s_i^k).$$

A Multiple Query Algorithm

Figure 41

After concepts related to l has been added to one query, concept l is removed from list L. The constructed query is stored in G_l.

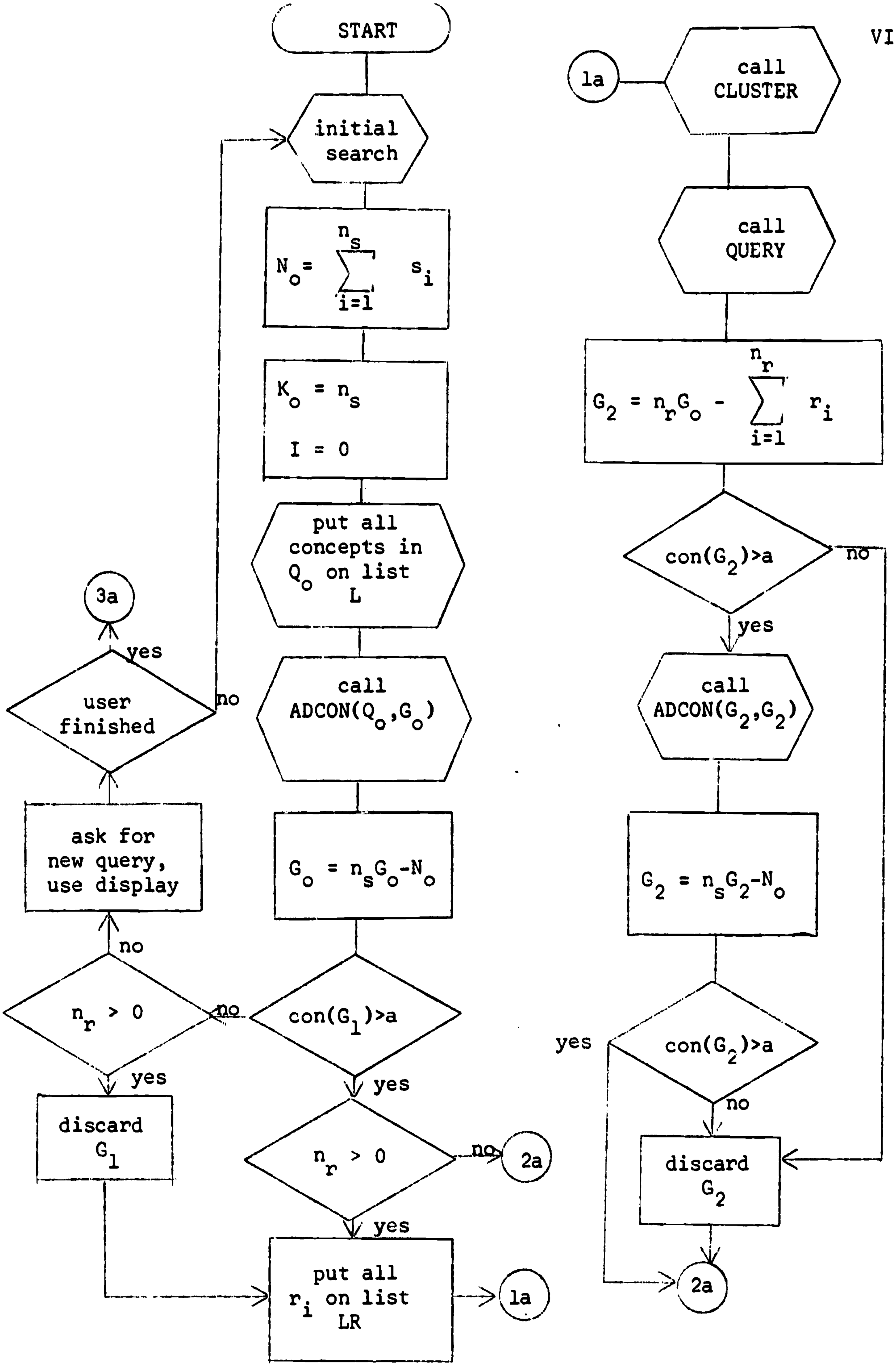
CLUSTER = clusters all documents on list LR by the criterion described in the definition of 'cluster j', adds the new clusters to the set J of clusters j, and clears list LR.

QUERY = forms a specific query S_j for each cluster j on iteration I as follows:

$$S_j = K_I \sum_{i=1}^{n_r^j} r_i^j - n_r^j \; N_I$$

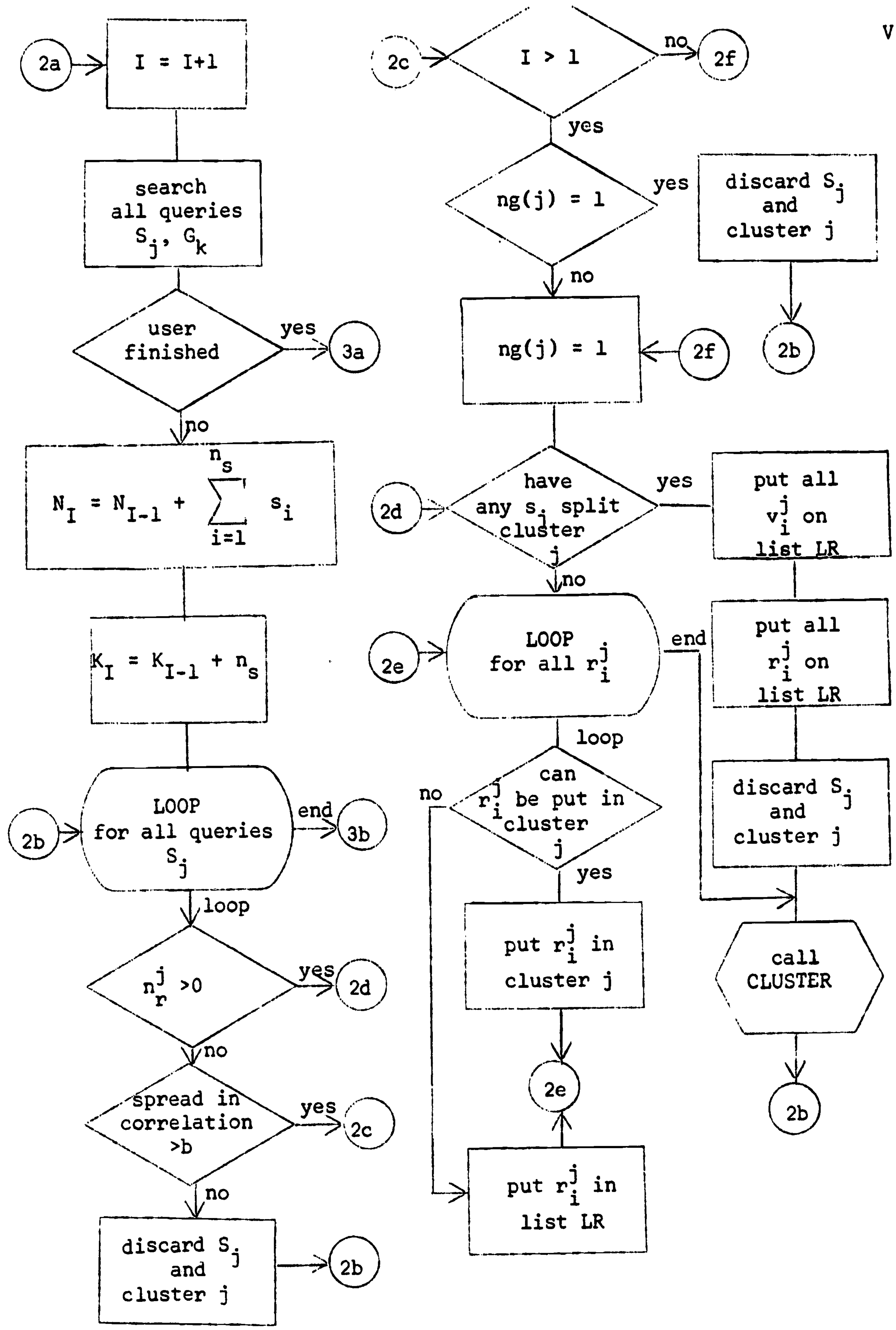
A Multiple Query Algorithm

Figure 41 (contd)



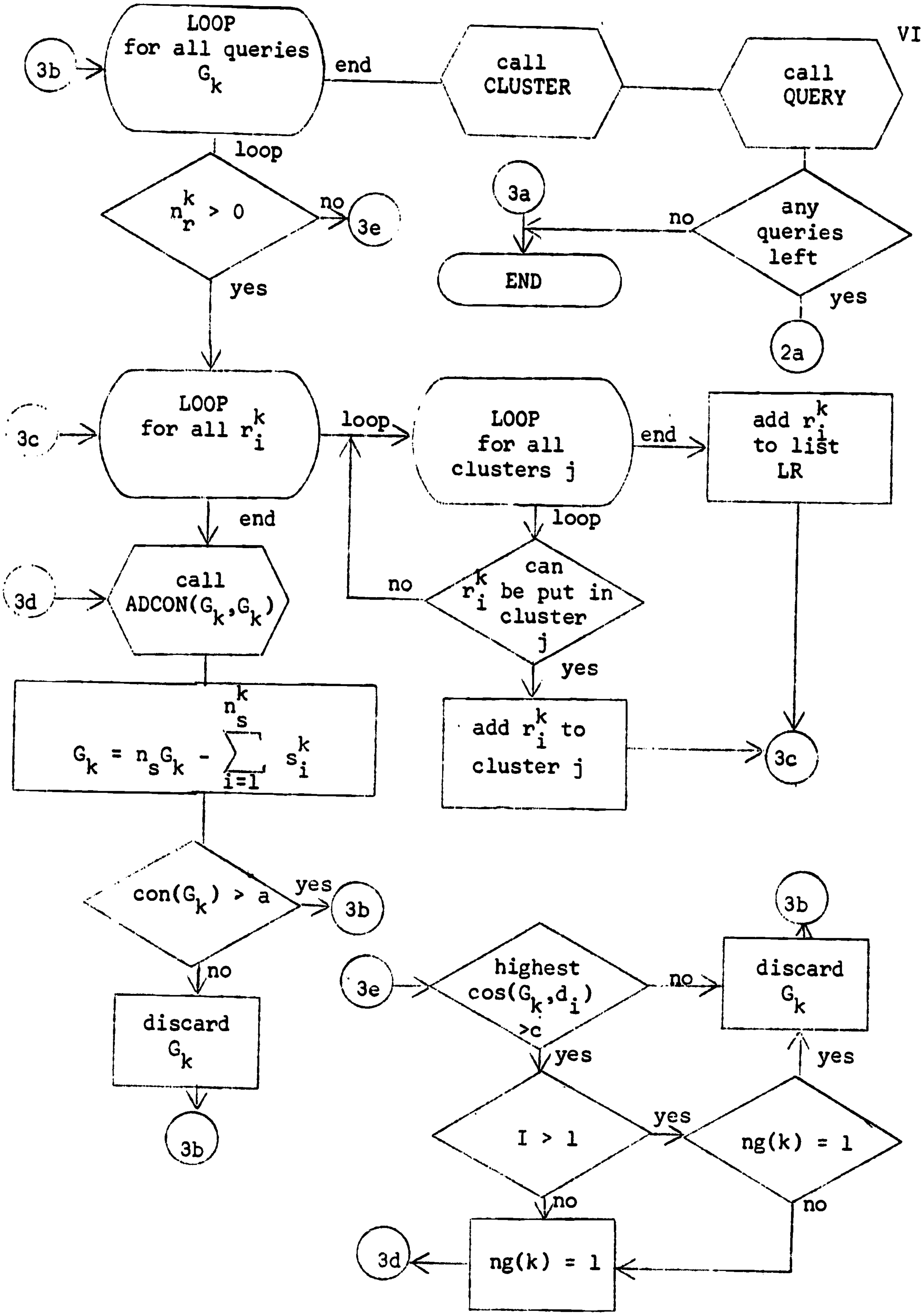
A Multiple Query Algorithm

Figure 41 (contd)


2a
I = I+1
search all queries S_j, G_k
user finished
yes
3a
no
$N_I = N_{I-1} + \sum_{i=1}^{n_s} s_i$
$K_I = K_{I-1} + n_s$
2b
LOOP for all queries S_j
end
3b
loop
$n_r^j > 0$
yes
2d
no
spread in correlation >b
yes
2c
no
discard S_j and cluster j
2b
2c
I > 1
no
2f
yes
ng(j) = 1
yes
discard S_j and cluster j
2b
no
ng(j) = 1
2f
2d
have any s_i split cluster j
yes
put all v_i^j on list LR
no
2e
LOOP for all r_i^j
end
put all r_i^j on list LR
loop
can r_i^j be put in cluster j
no
yes
discard S_j and cluster j
put r_i^j in cluster j
call CLUSTER
2e
2b
put r_i^j in list LR

A Multiple Query Algorithm

Figure 41 (contd)



A Multiple Query Algorithm

Figure 41 (cont'd)

previous queries suggested by Salton [22]. All original queries submitted to the retrieval system are saved, along with the documents found relevant to each query by the user during the search. When enough queries have accumulated, the query vectors are clustered by Rocchio's clustering algorithm or a similar method. Then the documents found relevant to the query vectors in each query cluster are grouped and used as a standard cluster for search. The user's query is first compared to the centroid vectors of the query clusters (not of the documents in the cluster) then to the concept vectors of the documents in the standard clusters defined by the query clusters closest to the user's query. Salton mentions that 'request clustering' would provide a means of automatically adjusting the retrieval algorithm to vocabulary shifts in a fast-moving technical field, especially for a document collection that attracts a homogenous user population. Request clustering offers another possible advantage, that the standard clusters of document are not based on the location of the document vectors in the document space. That is, the documents in the standard cluster defined by a query cluster (the cluster of documents relevant to the queries in the cluster) may not be adjacent in the document space, but may be intermingled with documents from other standard clusters. It now appears that the documents relevant to a query are usually found intermixed with nonrelevant documents. Request clustering may offer a greater possibility of simplifying such a retrieval situation than does document clustering.

The idea of request clustering is based on several assumptions. These assumptions deserve review as indications of initial dire tions for investigation.

1. It is assumed that the relationship of each document in the collection to one or more request clusters is well-defined. For this assumption to be valid, each document should be relevant to several queries in the clustered sample.

2. For a user environment, it is assumed that relevance information obtained during search is an adequate representation of the needs of the user formulating the query. This may not always be the case, since an inadequate search may fail to reveal relevant documents that the user does not know are available. The appropriateness of the relevance judgments obtained for experimentation is even more important in request clustering than in other retrieval experiments.

3. It is assumed that similar queries have similar sets of relevant documents, and that dissimilar queries tend to have non-overlapping sets of relevant documents. The failure of the first half of this assumption casts doubt on the basic rationale of request clustering. The failure of the second half might mean that a costly degree of standard cluster overlap is unavoidable. Both halves of this assumption can be tested statistically in various document and query collections before request clustering is implemented.

4. If request clustering is used in preference to document clustering, it is assumed that documents retrieved by similar queries are more appropriately related for retrieval purposes than are documents with similar descriptor vectors. This assumption can only be tested by using the same cluster search algorithm with request clusters and with document clusters. It might be found true for some search algorithms and false for others.

If assumption 4 is true, it suggests that the document vectors should be altered to correspond with the relationships indicated by user requests and relevance judgments. Means of dynamically altering the document space using previous user queries and relevance judgments have been investigated, and two algorithms that permanently alter the document vectors have been suggested. Both algorithms are here discussed.

Davis, Linsky, and Zelkowitz [27] base their approach to document space modification on two assumptions, here quoted:

a) "For a given query, concepts which appear more frequently in relevant documents than in nonrelevant documents probably contribute significantly to the relevance of the pertinent documents. The significant concepts are related to one another and often occur in conjunction with one another. Thus by raising the weights of these concepts in all documents within the entire space which contain occurrences of these concepts, similar documents are brought closer together".

b) "Any relevant document (as determined by user feedback) which does not contain an instance of a given concept determined to be significant is likely to contain material which nonetheless relates to this concept. Therefore, this concept is added to that relevant document. It is expected that by increasing the weights of these concepts more relevant documents will be clustered together and ultimately retrieved when a similar query is processed in the future".

The algorithm suggested by Davis, Linsky, and Zelkowitz is almost completely described by their assumptions. From user relevance judgments on the first 15 documents retrieved by the initial search, a vector of 'significant' concepts is formed by subtracting the sum of the vectors of retrieved nonrelevant documents from the sum of the vectors of retrieved relevant documents and setting all negative weights in the resulting vector to zero. This vector is then divided by the sum of the vectors of all retrieved documents. Each significant concept thus is assigned a positive fractional weight proportional to its significance. Every document in the space is then multiplied by $(1 + d_i)$. Also, each significant concept i is added to every relevant document vector not containing it, in accordance with assumption b).

A closer examination of the assumptions quoted predicts the effects of the resulting algorithm. Assumption a) states that because concept i is important in distinguishing the retrieved relevant documents from the retrieved nonrelevant documents, the importance of concept i in the document space should be emphasized by raising the weights of every occurrence of concept i. The algorithm based on this assumption tends to increase the correlation coefficients among all documents containing concept i. It tends to decrease the correlation of any document containing concept i with any document not containing concept i, because concept i appears only in the denominator of the cosine correlation between two such documents. However, since the documents used to select concept i as 'significant' are all retrieved by the user's query, all have relatively high correlations with that query; that is, both the relevant and nonrelevant documents retrieved contain a relatively high proportion of the concepts found in the query. Therefore the concepts that best distinguish between retrieved relevant and retrieved nonrelevant documents are not likely to be found in the user's query. The suggested algorithm is thus likely to decrease the correlation of most altered documents with the user's query. Assumption b), by suggesting that concept i be added to every retrieved relevant document not containing it, guarantees that the correlation of every retrieved relevant document with the user's query is lowered.

In fact, Davis, Linsky, and Zelkowitz report that while their algorithm does bring relevant documents closer together in the document space, it degrades the retrieval performance of the user's query. The three authors then argue that the resulting clustering of relevant documents is a desirable result and that relevance feedback can be used to overcome the initial degradation of performance and provide ultimately better retrieval. In their examples relevance feedback in the modified document space provides better retrieval than relevance feedback in the unmodified space. However, the examples given of document-document correlations show that while some unretrieved relevant documents are brought closer to some retrieved relevant documents and to each other, these affected documents are moved further away from still other relevant documents.. This result indicates that not all relevant documents contain the concepts selected as significant discriminators.

Ignoring for the moment the unfortunate reported effects on initial retrieval and examining only assumption a), the suggested algorithm can be questioned on theoretical grounds. Assumption a) states that all concepts found useful in distinguishing relevant from nonrelevant documents in the existing document space should be emphasized by increasing all weights assigned to that concept. The resulting process is essentially circular in that it uses the characteristics of the document vectors to change the document vectors. Imagine an ideal document collection in which every document is equally needed by users and every concept is equally useful in distinguishing among documents. Given a representative sample of information requests, the suggested algorithm would emphasize each concept in turn, resulting in no effective change to the document space. In a typical collection, this algorithm would eventually eliminate concepts that are either relatively useless for discriminating between documents or that are useful only for discriminating among documents not often requested. Only the first of these effects can be considered useful. In short, the suggested algorithm does not accomplish what should be its prime aim, to alter the document space in such a way as to provide retrieval performance closer to that expected by the users.

Brauen, Holt, and Wilcox [28] suggest a simpler document vector modification algorithm that does accomplish this aim. The concept vector of the user's query is added to the vectors of documents selected by the user as relevant to that query. The document vector modification formula is:

$$d_i = (1-\alpha)\ d_i + \alpha\overline{q}_o$$

where $\overline{q}_o$ is the user's query vector normalized to be equal in length to the document vector d_i. This formula does not change the length of the document vector.

A 425 document subset of the Cranfield 1400 collection, in which each document is relevant to at least one query, is used to test this algorithm. The 155 available queries are partitioned randomly in two ways into an update sample of 124 queries and a test sample of 31 queries. For values of α from 0.05 to 0.4, an average 3.3% improvement in normalized recall and 15.5% improvement in normalized precision are obtained. Every improvement is significant at the 1% level, as measured by the T-test. The changes in α cause no significant change in performance.

In one special experiment the modification algorithm is applied to a document space of zero vectors; that is, document vectors are derived from the queries and relevance judgments only. Results approaching the performance in the original document space are obtained, with normalized recall 1.2% lower and normalized precision 13.4% lower than the original results. Since 425 document vectors are being defined entirely by the information contained in only 124 queries, these results are surprisingly good.

The results reported by Brauen, Holt, and Wilcox indicate that queries and relevance judgments contain information useful to future retrieval. The speical experiment strongly suggests that given a larger query sample to use for document vector modification, this information may in fact be of more value than that contained in the original document vectors. Thus assumptions 3 and 4, stated in the earlier discussion of request clustering, are supported. Two cautions in generalizing these results to practical retrieval systems are necessary. First, Brauen, Holt, and Wilcox force assumption 1 to be true by their intelligent selection of a document collection. In an actual system, there is no guarantee that every document will be relevant to at least one query in a modification sample. Second, the relevance judgments supplied for experimental evaluation are used for document vector modification even though some of the relevant documents would not have been retrieved by an initial search of the user queries. Therefore assumption 2 is not tested by these experiments. Further investigation of these two assumptions in realistic document and query collections is needed. Nevertheless, the results reported by Brauen, Holt, and Wilcox encourage the investigation not only of document space modification but also of request clustering.

An analogy to document space modification is found in the more fully explored field of adaptive pattern recognition. An adaptive system first described by Nilsson [29] and studied by many later experimenters is directly comparable to the SMART system in several meaningful respects. The task of a pattern recognition system is to assign each pattern presented to the correct class of patterns; for example, to recognize each spoken word as a 'one', 'two', or other single digit. For each class i of patterns to be recognized, a weight vector w_i is constructed. A pattern x is assigned to class if and only if $w_i \cdot x$ is greater than $w_j \cdot x + \Theta$ for all classes j not equal to i. The following adaptive algorithm adjusts the weight vectors to a set of patterns used for 'training'.

If a pattern x belonging to class i is presented to the system, and $w_i \cdot x$ is greater than $w_j \cdot x + \Theta$ for all j not equal to i, no adjustment to the weight vectors takes place. However, if for some k not equal to i the dot product $w_i \cdot x$ is less than $w_k \cdot x + \Theta$, the pattern x is added to the vector w_i and subtracted from the vector w_k. The parameter Θ is greater than 1 and is called the 'training threshold'.

The concept vector of a user's query is analogous to the input pattern in such a pattern recognition system. The query 'pattern' is assigned to a 'class' by the SMART system when a document is selected as relevant to the query. The document vectors thus correspond to the weight vectors w_i. Just as similar patterns are assigned to the same class by the pattern recognition system, similar queries select similar documents in the SMART system. In fact the vector dot product $w_i \cdot x$ equals the sum $\sum_j w_i^j x^j$, and therefore corresponds exactly to the cosine correlation coefficient whenever the two vectors are of length 1.

The one weakness in the suggested analogy is obvious - each query is expected to select more than one document as 'relevant', while each pattern is assigned to only one class. Nevertheless, a 'training algorithm' for document vectors can be constructed that should improve this 'multi-class assignment' in the same way that the adaptive pattern recognition algorithm improves single class assignment. Such an algorithm would modify the SMART document vectors using queries as patterns and relevant documents as 'correct responses'. An adaptive algorithm for document space modification is here stated in information retrieval terminology.

Given a set of user queries with relevance judgments, the document vectors are altered as follows:

If for all i such that document i is relevant to the submitted query q_o, and for all j such that document j is not relevant to q_o, $\cos(d_i,q_o)$ is greater than $\cos(d_j,q_o) + \Theta$, no adjustment to the document vectors is made. However, if this condition does not hold each vector d_i denoting a relevant document i is processed in order of its correlation with q_o as follows:

If there exists a document k such that vector d_k has not yet been adjusted by this query q_o and k is not relevant to q_o, and $\cos(d_k,q_o) + \Theta$ is greater than $\cos(d_i,q_o)$, then the query q_o is added to the vector d_i and subtracted from the vector d_k having the highest correlation with q_o. If there exists no nonrelevant document meeting all these requirements, but there exists a document k with previously adjusted vector d_k meeting the other requirements, the query q_o is added to d_i but not subtracted from d_k. The suggested order of processing insures that a different nonrelevant document is decremented for every relevant document incremented whenever this is possible. The suggested multi-class adaptive algorithm is more cautious than the single-class adaptive algorithm in that the vector associated with each correct response is incremented only once while in the pattern recognition algorithm the vector associated with the single correct response is incremented once for each incorrect response that is decremented. Also, the single-class adaptive algorithm decrements all incorrect responses that are stronger than the correct response. In a document retrieval situation, a similar procedure could decrement every nonrelevant document in the space. The algorithm suggested above limits the number of document vectors decremented to the number of relevant document vectors incremented. An alternative way to limit negative document vector adjustment is to decrement the vectors of all nonrelevant documents retrieved within the first n. If computting time allowed, this document space modification algorithm could adapt during any relevance feedback algorithm suggested in this study by incrementing the retrieved relevant document vectors and decrementing the retrieved nonrelevant document vectors. Only the initial query rather than the queries modified by relevance feedback should be used to adjust the document vectors.

In a document retrieval application, some means of controlling the length of the modified vectors is needed. A decrementing formula analogous to the length-preserving incrementing formula suggested by Brauen, Holt, and Wilcox, is $d_i = (1 + \alpha)\ d_i - \alpha\overline{q}_o$.

The further investigation of the adaptive document space modification algorithm suggested is encouraged by two findings, the successful performance of the analogous single-class adaptive algorithm in many different pattern recognition applications, and the improved results reported by Brauen, Holt and Wilcox, whose algorithm increments relevant document vectors in the same manner as the suggested algorithm without adjusting nonrelevant document vectors. Since the algorithm suggested discourages incorrect responses as well as encouraging correct responses, it shoud be even more effective than the Brauen, Holt, and Wilcox algorithm in adjusting the responses of the retrieval system to the expectations of its users.

In this final section of this thesis, implications for future research are drawn from the conclusion reached in Section VII-C that the documents relevant to one query are not normally clustered in an exclusive area of the document space. With reference to partial search algorithms, new measures for evaluating the potential usefulness of a given partition of the document collection regardless of the search algorithm used are suggested. The cluster search algorithm is shown to be inappropriate in environments similar to the experimental collection, and a better cluster search algorithm is proposed. A combination of cluster search with relevance feedback that constructs a separate feedback query to search each cluster is supported as a possible solution to the problem posed by separated groups of relevant documents. If cluster feedback is found inadequate, strategies that construct more than one query to search the same set of documents are shown to be necessary. Several suggestions for the design of such multiple query algorithms are made, culminating in a detailed flowchart of an algorithm that can be meaningfully tested in the Cranfield 200 collection. Finally, request clustering and permanent document space modification are discussed as ways of possibly providing single query retrieval situations for most users by investing time in off-line processing rather than lengthening the search process. An algorithm for adaptive document space modification using queries and relevance judgments is constructed by analogy to a well-tested method that performs a similar function in adaptive pattern recognition systems.

Summary

Several algorithms that allow user interaction with an automatic document retrieval system by requesting relevance judgments of selected sets of documents are investigated. All relevance feedback algorithms tested improve the average retrieval obtained. The improvement caused by relevance feedback is greater in a larger, more realistic document collection (Cranfield 200) than in a smaller and less realistic collection (ADI).

No single feedback strategy is found to give superior retrieval for all queries. Algorithms using only relevant documents for feedback can be made effective for queries that retrieve no relevant documents on the initial search in two ways, by supplying additional documents for relevance judgments or by using nonrelevant documents for feedback.

In general, the performance of negative feedback algorithms (those using nonrelevant documents for feedback) is variable. Negative feedback gives worse performance on some queries and better performance on others than the more consistent positive feedback strategies. On the average, negative feedback moves the query closer to the optimum query defined by Rocchio than does positive feedback, but does not differ significantly from positive feedback from the viewpoint of the user during feedback.

A study of selected subgroups of queries provides three interesting results. First, movement of the query constructed by the Rocchio strategy further away from the original query rather than back toward it on the second iteration is related to poor performance and poor initial search results. This direction of query movement could be a result of inadequate feedback, or it could be an attempt to compensate for a poor original query. Second, the queries for which the Rocchio strategy gives less improvement than positive feedback have the worst initial search performance, queries for which positive feedback is inferior have much better initial search performance, and queries for which negative and positive feedback are equal have the best initial search retrieval. Third, queries having few concepts and few relevant documents and queries having many concepts and few relevant documents tend to give less improvement with negative feedback than with positive feedback, while queries for which the number of concepts and number of relevant documents are directly related give more improvement with negative feedback. If a predictor of the number of relevant documents available for each submitted query can be found, this relationship could be used to select the feedback algorithms appropriate to each query.

The observed contrasting behavior of negative and positive feedback algorithms can be explained by a hypothesis presented in Section VII-C.

Hypothesis:

A hypothesis is presented that explains some of the observed performance differences between the negative feedback strategies and the positive feedback strategies investigated, and is consistent with all experimental results reported.

Hypothesis:

For most queries, for every vector v contained in the set R of relevant document vectors there exists at least one vector s contained in the set S of nonrelevant document vectors such that for some other vector r contained in R, cos(r,s) is greater than cos(v,r). Further, for a significant number of queries the prevalence of such relationships effectively prevents the retrieval of some relevant documents with reasonable precision by any relevance feedback strategy that constructs only one query on each iteration.

This hypothesis states in effect that the documents relevant to a single query are usually found in two or more distinct clusters in the concept vector space, and that these clusters of relevant documents are separated from each other by nonrelevant documents. Further, it states that for a significant number of queries this phenomenon will seriously interfere with the retrieval of some relevant documents regardless of the relevance feedback strategy employed. For any collection in which this hypothesis is true, all relevance feedback algorithms tested in this study are inappropriate for a significant percentage of retrieval requests. Algorithms constructing more than one query on each feedback iteration are necessary in such an environment.

The anomalous results of the reported comparisons of positive and negative feedback support the conclusion that the stated hypothesis is true in the Cranfield 200 collection. Because this collection is a carefully chosen subset of a larger collection representative of a well-defined, technical, limited subject area, this conclusion suggests that multiple query algorithms or other means of simplifying the distribution of relevant document vectors in the vector set being searched will be needed in practical automatic retrieval systems.

Section VII contains many recommendations for future research in relevance feedback. Positive feedback with the combination feedback strategy that presents more documents to users that judge no documents relevant on a given iteration is recommended in Section VII-A for environments similar to the Cranfield 200 collection. Nonrelevant document feedback is recommended for users who find no relevant documents after a maximum number of additional documents have been presented. Section VII-B discusses the evaluation problems encountered in this study. New global measures similar to normalized recall and normalized precision are suggested. The Quasi-Cleverdon interpolation method is recommended in perference to Neo-Cleverdon interpolation for recall-precision curves. Three new evaluation viewpoints for relevance feedback are suggested, one of which is appropriate to other areas. A fourth evaluation method is discussed and recommended for general use.

Section VII-D discusses the implications of the conclusion reached from the hypothesis of Section VII-C for partial search strategies, multiple query strategies, request clustering, and document space modification. Two new measures for evaluating the usefulness of a given partition of the document collection regardless of the partial search algorithm employed are presented. The cluster search algorithm used in earlier studies of Roochio's clustering algorithm in the SMART system is inappropriate where groups of documents relevant to the same query are separated by nonrelevant documents, so a new cluster search algorithm is presented. This new algorithm is combined with relevance feedback to form a cluster feedback algorithm that creates a different query to search each document cluster. Several design considerations for multiple query algorithms, strategies that construct more than one query to search the same set of documents, are proposed, culminating in a detailed algorithm simple enough to be meaningfully tested in a small document collection. Finally, request clustering and document space modification are discussed as possible ways of making multiple query algorithms unnecessary by additional processing that does not take place during the search. An algorithm for permanent adaptive alteration of the document vectors using queries and relevance judgments is constructed by analogy to a well-tested algorithm that performs a similar function in adaptive pattern recognition systems.

Bibliography

[1] Richmond, Phyllis Allen, "What Are We Looking For?", Science, Vol. 139, February 22, 1963.

[2] G. Salton, Automatic Information Organization and Retrieval, McGraw Hill Book Co., 1968, Chapter I.

[3] G. Salton, Automatic Information Organization and Retrieval, McGraw Hill Book Co., 1968, Chapter II.

[4] G. Salton and M. E. Lesk, Computer Evaluation of Indexing and Text Processing, Scientific Report No. ISR-12 to the National Science Foundation, Section III, Department of Computer Science, Cornell University, June 1967.

[5] G. Salton, Automatic Information Organization and Retrieval, McGraw Hill Book Co., 1968, Chapter VII.

[6] M. E. Lesk and G. Salton, Evaluation of Interactive Search and Retrieval Methods Using Automatic Information Displays, Technical Report No. 68-17, Department of Computer Science, Cornell University, revised September 1968.

[7] M. Rubinoff, S. Bergman, W. Franks, and E. Rubinoff, Experimental Evaluation of Information Retrieval Through a Teletypewriter, Communications of the ACM, Vol. 11, No. 9, September 1968.

[8] J. J. Rocchio, Relevance Feedback in Information Retrieval, Report ISR-9 to the National Science Foundation, Chapter 23.

[9] J. J. Rocchio, Document Retrieval System Optimization and Evaluation, Harvard University Doctoral Thesis, Report ISR-10 to the National Science Foundation, Harvard Computation Laboratory, June 1965.

[10] J. J. Rocchio, G. Salton, Search Optimization and Iterative Retrieval Techniques, Proceedings of the Fall Joint Computer Conference, Las Vegas, November 1965.

[11] O. W. Riddle, T. Horwitz, R. Dietz, Relevance Feedback in Information Retrieval Systems, Report ISR-11 to the National Science Foundation, Chapter 6, Department of Computer Science, Cornell University, June 1966.

[12] R. Crawford and H. Melzer, The Use of Relevant Documents Instead of Queries in Relevance Feedback, Scientific Report No. ISR-14 to the National Science Foundation, Section XIII, Department of Computer Science, Cornell University, October 1968.

Bibliography (contd)

[13] D. Steinbuhler and C. Aleta, Negative Relevance Feedback Process, student reports, Computer Science 435, Department of Computer Science, Cornell University, Spring 1968.

[14] J. Kelly, Negative Response Relevance Feedback, Scientific Report No. ISR-12 to the National Science Foundation, Section IX, Department of Computer Science, Cornell University, June 1967.

[15] G. Salton, The Evaluation of Automatic Retrieval Processes, Selected Test Results Using the SMART System, American Documentation, Vol. 16, No. 3, July 1965.

[16] F. Wilcoxon, R. Wilcox, Some Rapid Approximate Statistical Procedures, Lederle Laboratories, America Cyanamid Company, 1964.

[17] H. Hall, N. Weiderman, The Evaluation Problem in Relevance Feedback Systems, Scientific Report No. ISR-12 to the National Science Foundation, Section XII, Department of Computer Science, Cornell University, June 1967.

[18] C. Cleverdon, E. M. Keen, Factors Determining the Performance of Indexing Systems, Vol. 2, ASLIB Cranfield Research Project, 1966, Chapter III.

[19] M. Lesk, G. Salton, Design Criteria for Automatic Information Systems, Report ISR-11 to the National Science Foundation, Chapter V, Department of Computer Science, Cornell University, June 1966.

[20] G. Salton, Automatic Information Organization and Retrieval, McGraw Hill Book Co., 1968, Chapter VI.

[21] R. Steel, J. Torrie, Principles & Procedures of Statistics, McGraw Hill Book Co., 1960, Chapter 21.

[22] G. Salton, Search Strategy and The Optimization of Retrieval Effectiveness, Scientific Report No. ISR-12 to the National Science Foundation, Section V, Department of Computer Science, Cornell University, June 1967.

[23] Leech and Matlack, Information Retrieval: Dictionary Representation and Cluster Evaluation, Scientific Report No. ISR-12 to the National Science Foundation, Section VII, Department of Computer Science, Cornell University, June 1967.

Bibliography (contd)

[24] Williamson, Williamson, and Ide, The Cornell Programs for Cluster Searching and Relevance Feedback, Scientific Report No. ISR-12 to the National Science Foundation, Section IV, Department of Computer Science, Cornell University, June 1967.

[25] R. Grauer and M. Messier, An Evaluation of Rocchio's Clustering Algorithm, Scientific Report No. ISR-12 to the National Science Foundation, Section VI, Department of Computer Science, Cornell University, June 1967.

[26] A. Borodin, L. Kerr, F. Lewis, Query Splitting in Relevance Feedback Systems, Scientific Report No. ISR-14 to the National Science Foundation, Section XII, Department of Computer Science, Cornell University, October 1968.

[27] M. Davis, M. Linsky, M. Zelkowitz, A Relevance Feedback System Employing a Dynamically Evolving Document Space, Scientific Report No. ISR-14 to the National Science Foundation, Section X, Department of Computer Science, Cornell University, October 1968.

[28] T. Brauen, R. Holt, T. Wilcox, Index Space Modification Based on Relevance Feedback, Scientific Report No. ISR-14 to the National Science Foundation, Section XI, Department of Computer Science, Cornell University, October 1968.

[29] N. Nilsson, Learning Machines, New York: McGraw Hill, 1965.