V. Search Strategy and the Optimization of Retrieval Effectiveness

G. Salton

Abstract

Future real-time information retrieval systems may be expected to utilize automatic text analysis procedures for the preparation of analyzed search requests, and user feedback information for the generation of a useful search strategy. The analysis procedures and the search strategies to be used will vary to some extent with the equipment used in the system, with the type of service to be furnished, and with the user population. If the user population is large, and service is to be rendered simultaneously to many users, then it is not possible to process each search request against an entire collection of stored items. Instead, a number of partial searches may be used to replace a single full search of the collection.

In the present study, various partial search strategies are described, based partly on document and request groupings, and partly on user feedback information. The SMART system is used to evaluate these strategies, and to postulate an efficient, real-time, user-controlled search strategy.

1. Introduction

Presently operating mechanized information systems are based on mechanized information files which can be searched mechanically. All other operations, including in particular the input operations, the indexing and analysis operations, and the processing of the final output are normally carried out with the help of human experts. In the foreseeable future such mechanized systems may be modified in two important respects: first, the analysis of incoming documents and search requests may be carried out automatically, instead of manually, using for this purpose a variety of stored dictionaries and tables, as well as statistical and syntactic text analysis methods; second, the operations may be based on time-sharing equipment, where access to the central store can be provided to a number of different users, more or less simultaneously by means of special input-output consoles.

A great deal of work has been done over the last few years in the area of automatic indexing in an attempt to generate indexing methods which could be incorporated into operating information systems. [1,2] Several evaluation studies have also been carried out to determine the effectiveness of many kinds of automatic text analysis procedures, and tentative conclusions have been reached concerning the relative effectiveness of the analysis methods under consideration. [3,4,5,6,7]

The area dealing with search strategies and with procedures designed to make the user participate in the search process has received much less attention. Instead, even in the experimental situations, searches are carried out in such a way that each analyzed search request is compared in turn against each analyzed document. Documents, or citations which exhibit a sufficiently high matching coefficient with a search request are then withdrawn from the file and handed to the appropriate user. The user population does not in general participate in the search process, over which it has no real control.

When time-sharing equipment becomes available in operational situations, the search process previously described can no longer be carried out efficiently. In those circumstances the search and retrieval system must overcome two substantial constraints of the existing time-sharing organizations:

a) the small amount of internal storage which can normally be allocated to any given user (users must compete for memory space with many other users);

b) the rudimentary nature of the input-output console equipment likely to be made available to each user, which permits the introduction or withdrawal of only limited amounts of information.

At the same time, the information system should profit from the fact that the customer can now be made a part of the system, by asking him periodically to provide feedback information designed to clarify his information need.

The limitations inherent in the restricted available storage space and in the simple typewriter-like input-output devices may be overcome by fast search algorithms, confined to only small subsections of the stored file, and by limited interactions with the user. Such fast, user-controlled search algorithms are described in the next few sections, and evaluation results obtained by using the SMART automatic retrieval system are given to illustrate the effectiveness of the various search and retrieval procedures. [8,9]

2. Cluster Search Process

A) Overall Process

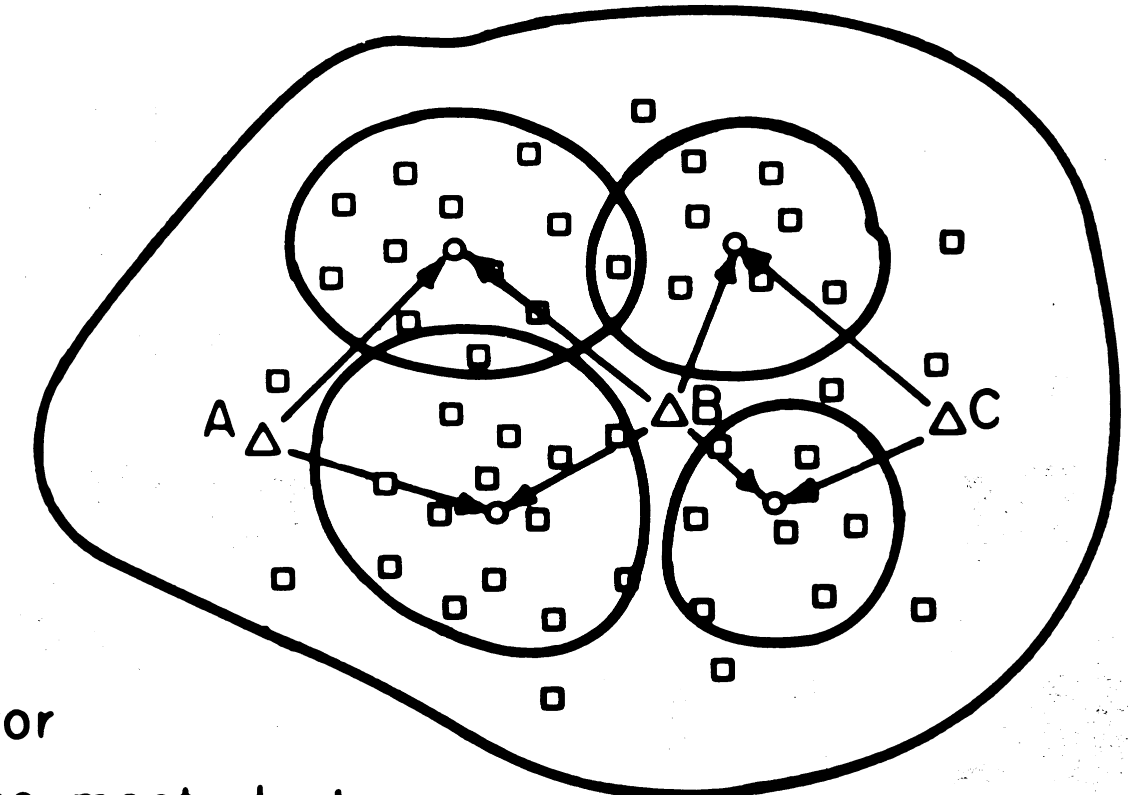
In a traditional library environment, answers to information requests are not usually obtained by conducting a search through an entire document collection. Instead, the items are first classified into subject areas, and a search is restricted to items within a few chosen subject classes. This same device can also be used in a mechanized system by constructing groups of related documents, and confining the search to certain groups only. Specifically, the following overall strategy can be used:

a) groups, or clusters of related documents are constructed by comparing the identifiers for a given document with the identifiers of all other documents, and by grouping those documents whose sets of identifiers are sufficiently similar;

b) for each such document group, a representative element, also known as the <u>centroid vector</u>, is chosen; this centroid vector is then used to represent the whole document set in that group;

c) the search proceeds in two steps: a given search request is first compared against the centroids of all document groups; a second search is then used to match the request against the individual documents located in groups with highly matching centroids.

A stylized picture of such a two-level cluster search is shown in Fig. 1, where each document is represented by a small square, and each search request by a triangle. It is seen that requests A and C lie close to the centroid vectors of two of the document clusters; the similarity coefficient between the requests and the corresponding centroids may therefore



□ Document vector

○ Centroid of document cluster

△ Request vector

Sample Clustered Document Space

Fig. 1

be expected to be large, and the document search is then confined to documents in the two respective groups only. Request B, on the other hand lies close to the centroid of four clusters, thus necessitating a detailed search of these four groups.

Obviously, the two-level search can be extended to a three-level, or even higher level search by grouping the centroid vectors themselves into broader groups of larger coverage, followed by a grouping of these broader groups into still broader ones, and so on. In that case, a search is first made of the centroids for the highest level groups; this isolates some centroid groups on the next lower level; a search of these identifies certain groupings on the next lower level, and so on down, until some document clusters are found which must be individually searched.

The efficiency of such a <u>multi-level</u>, or <u>cluster search</u> varies with the clustering process used, and with the collection under consideration. It is greatest when the collection can be subdivided into nonoverlapping groups of approximately identical size. It diminishes as the amount of overlap between groups increases, and the size of the groups begins to deviate from a common ideal value. Obviously, a cluster search will not avail if the documents of interest to the user are not in fact included in the groups which are to be searched individually, since such relevant documents are not then retrievable. This fact will be brought out further when the systems evaluation is discussed.

B) Cluster Generation

The problem which consists in taking sets of items identified by certain properties, and in grouping them in such a way that items identified by a common property set are placed into a common class, is well known in many fields. A number of mathematical techniques have been used in the past with varying degrees of success in the implementation of a clustering program, including matrix eigenvalue analysis, factor analysis, latent class analysis, and others. Some of these techniques have also been applied to the documentation area, where the items to be grouped are documents, and the properties used to effect the grouping are keywords, or index terms attached to the documents.[10,11,12]

The process to be described here is due to Rocchio and differs from some of the others in that the number of clusters to be generated can be controlled, as well as the cluster size, and the amount of overlap between clusters.[13] Such controlled clusters may be more useful in an application to documentation, than clusters which are subject to large size variations and to a great degree of overlap.

All documents are initially considered to be unclustered, and each document is first subjected to a region density test to determine whether a sufficient number of other documents are located in the same vicinity. This test specifies that more than n_1 items should have a correlation higher than some parameter p_1 with the candidate, and that more than n_2 items should have correlations higher than p_2. The test insures that items on the edge of large groups do not become centers of groups, and that annular regions where items are concentrated in a ring-like area around the candidate item are not accepted as clusters. An example of a density test failure is shown in Fig. 2, where an attempt is made to pick document 13 as a cluster center. In the example, the requirement that at least five documents have a correlation greater than 0.25 with document 13 is not met, since the fifth highest correlation (with document No. 19)

Document Rank	Document Number	Correlation
1	13	1.0000
2	24	0.3664
3	26	0.3071
4	74	0.2643
5	19	0.1979
6	22	0.1453
7	59	0.1248
8	45	0.1172
9	78	0.1166
10	38	0.1161
11	46	0.1077
12	75	0.0882
13	17	0.0844
14	23	0.0722
15	36	0.0641
16	4	0.0640
17	63	0.0507
18	81	0.0447
19	55	0.0447
20	35	0.0369
21	57	0.0358
22	80	0.0207
23	16	0.0181
24	25	0.0175
25	77	0.0149
26	44	0.0135
27	82	0.0000
28	73	0.0000
29	69	0.0000
30	54	0.0000
31	50	0.0000
32	14	0.0000

Density Test Failure

(less than 5 documents exhibit correlation greater than 0.25)

Fig. 2

is only 0.1979. Items which fail the density are considered to be "loose" and are not again chosen as potential cluster centers.

If a document passes the density test, a cut-off value is chosen as a function of the preestablished minimum and maximum number of permissible items per cluster, and items whose correlation with the central document is larger than the cut-off value are used to define a cluster. In the example of Fig. 3, items are grouped around document 7, which previously passed the density test, and the six top documents (nos. 7, 42, 9, 20, 32, and 31) with a correlation above cut-off define an initial cluster. The cut-off is picked at the point of maximum correlation difference between two adjacent documents to produce the shortest boundary between identified subset and neighboring unclustered items.

Given the set of documents D defining a cluster, the centroid vector is chosen as the center of gravity of the set of document vectors derived from the elements of D. Specifically, if each document is identified by a property, or keyword vector, $\underline{d}$, the centroid vector is defined as

$$\underline{C} = \sum_{\underline{d}^{(i)} \in D} \underline{d}^{(i)} \quad .$$

The centroid vector C_1 which results from the addition of the six document vectors identified in Fig. 3, is shown in Fig. 4. The documents defining the group are listed at the top of the figure, and the centroid vector itself consists of 65 concepts (represented by 3-digit numbers) each with a specified weight.

The centroid vector thus derived is now matched against the <u>entire</u> document collection, and the cut-off parameters on category size are

Document Rank	Document Number	Correlation
1	7	1.0000
2	42	0.4352
3	9	0.3935
4	20	0.3541
5	32	0.3002
6	31	0.2789
cut off →		
7	25	0.2374
8	22	0.2130
9	73	0.1984
10	57	0.1949
11	81	0.1826
12	55	0.1826
13	75	0.1801
14	78	0.1705
15	36	0.1527

Correlation of Top 15 Documents with Document No. 7

Fig. 3

Document Rank	Document Number	Correlation
1	7	0.7853
2	42	0.7028
3	9	0.5593
4	20	0.5497
5	31	0.5007
6	32	0.4425
7	73	0.3518
8	40	0.3049
9	56	0.2957
10	75	0.2950
11	1	0.2685
12	51	0.2516
13	25	0.2473
14	57	0.2468
15	55	0.2463

Correlation of Top 15 Documents with Centroid C_1
(cluster contains Docs. 7, 9, 20, 31, 32, 42)

Fig. 5

Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights
1	24	3	120	5	12	6	24
7	12	8	24	10	24	19	24
23	24	24	36	28	12	30	24
32	24	33	24	40	48	43	12
44	36	47	24	50	12	54	6
57	12	58	36	67	12	70	24
71	12	72	12	73	24	76	48
78	12	79	36	87	12	89	24
95	12	103	24	108	12	113	12
114	24	122	12	130	78	134	12
149	12	152	12	172	12	180	12
181	12	205	12	207	24	211	12
222	12	246	36	258	72	259	12
261	12	262	12	278	12	285	12
291	12	298	12	322	12	323	12
349	12	532	12	594	24	600	6

Formation of Centroid C_1 Using Documents
(7,9,20,31,32,42)

Fig. 4

reapplied to create an altered cluster. The results of this matching operation are shown in Fig. 5 for the centroid C_1 of Fig. 4. The cutoff again falls between the sixth and seventh documents, and the resulting cluster identified in the example of Fig. 5 is the same as that which originally defined the cluster in Fig. 3. Such a result is of course not necessarily obtained in all cases.

This clustering process is now repeated with all unclustered items and the first pass ends when all items are either clustered or loose. Since the centroid vectors are correlated against the entire collection, some items may of course end up in several different clusters. If the number of categories formed is less than the number originally specified, a second pass could be made with relaxed density conditions. Alternatively, the density test could be made more restrictive, or the category size limits could be increased.

At the end of this initial clustering operation, a relatively large number of items might remain loose. Furthermore, the amount of overlap between clusters might be considerable. Under these circumstances, it is possible to use an additional optional clustering pass based on the formation of a partition class for each centroid vector. Specifically each document is assigned to that centroid with which it exhibits the highest correlation, and the document groups so obtained are used to define a new centroid. For the centroid C_1 of Fig. 4, this maximum correlation partition specifies documents 9, 20, 31, 32, and 42. These five documents in turn define the new centroid C_2 shown in Fig. 6.

It may be noted that document No. 7 which was originally used as the center for the clustering operation given in the example is no longer

Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights
1	24	3	84	5	12	6	12
7	12	8	24	10	12	19	12
22	6	23	24	24	36	28	12
30	24	32	24	33	24	40	36
44	24	47	12	50	12	54	6
58	24	67	12	70	24	71	12
72	12	73	24	76	36	78	12
79	24	87	12	89	24	103	24
108	12	113	12	114	24	172	12
180	12	181	12	205	12	207	12
222	12	246	36	258	48	259	12
261	12	262	12	278	12	285	12
291	12	322	12	349	12	532	12
594	24	600	6				

Formation of New Centroid C_2 from Minimum Correlation Partition
(using documents 9,20,31,32,42)

Fig. 6

Document Rank	Document Number	Correlation
1	42	0.7271
2	7	0.6246
3	20	0.5647
4	9	0.5609
5	31	0.5298
cut off →		
6	32	0.4482
7	73	0.3712
8	75	0.3061
9	56	0.2746
10	40	0.2701
11	49	0.2649
12	1	0.2502
13	55	0.2438
14	78	0.2400
15	57	0.2400

Correlation of Top 15 Documents with Centroid C_2

Fig. 7

Document Rank	Document Number	Correlation	
1	42	0.7271	original documents
2	7	0.6246	
3	20	0.5647	
4	9	0.5609	
5	31	0.5298	
6	32	0.4482	loose documents added by "blending" routine
7	73	0.3712	
8	75	0.3061	
9	78	0.2400	
10	25	0.2252	
11	54	0.2044	
12	38	0.1790	
13	63	0.1592	

Final Cluster around Centroid C_2 after Blending

Fig. 8

present, since its highest centroid correlation occurs with a centroid other than C_1. The centroid C_2 of Fig. 6 lacks some of the concepts originally present in C_1, and the weights are generally lower.

The new centroid is now correlated against the complete document collection as before, and a cut-off determines a new cluster, consisting for the case used as an example of documents 7, 9, 20, 31, and 42, as shown in Fig. 7. A "blending" routine is now used to assign loose documents to that group with which they exhibit the highest correlation. For the example given in Figs. 3 to 7, the results of the blending operation are shown in Fig. 8.

To summarize, the complete process consists of three grouping operations: the first around the initial items which pass the density test; the second around the centroids of the clusters previously generated; and the third around the new centroids obtained after partition of the previous sets. For the example, the changes in the generated cluster are summarized in Fig. 9.

Fig. 10 lists the parameters which enter into the cluster generation process, including density control parameters, and cluster size parameters. These parameters are used to control the number of clusters, and amount of overlap desired, and also to exclude certain items from the clustering process, or to delete concepts of low weight from the document and centroid vectors.

Fig. 11 shows in summary form the results of a clustering operation for a collection of 82 documents in the documentation area. Each cluster is identified by a different numeric digit, ranging from 1 for the first cluster to 7 for the last. In each case, the correlation coefficient of

Generator	Resulting Cluster
1) Document 7	7,9,20,31,32,42
2) Centroid C_1	7,9,20,31,32,42
3) Minimum Correlation Partition	9,20,31,32,42
4) Centroid C_2	7,9,20,31,42
5) Centroid C_2 with Blending	7,9,20,25,31,32,38,42,54, 63,73,75,78.

Summary of Generation Process for Typical Cluster

Fig. 9

Type of Control	Function
Master Control	Use of maximum correlation partition to redefine clusters Placement of loose documents in clusters Documents to be included in clustering process
Density Test Control	Minimum number of documents with correlation exceeding p_1 Minimum number of documents with correlation exceeding p_2 Minimum significant correlation Documents to be considered as cluster roots
Cluster Size Control	Type of correlation doefficient Minimum number of documents per cluster Maximum number of documents per cluster Minimum significant correlation difference Correlation difference sufficient to force a break between clusters Weight of concept to be deleted from vector Type of centroid definition

Clustering Parameters

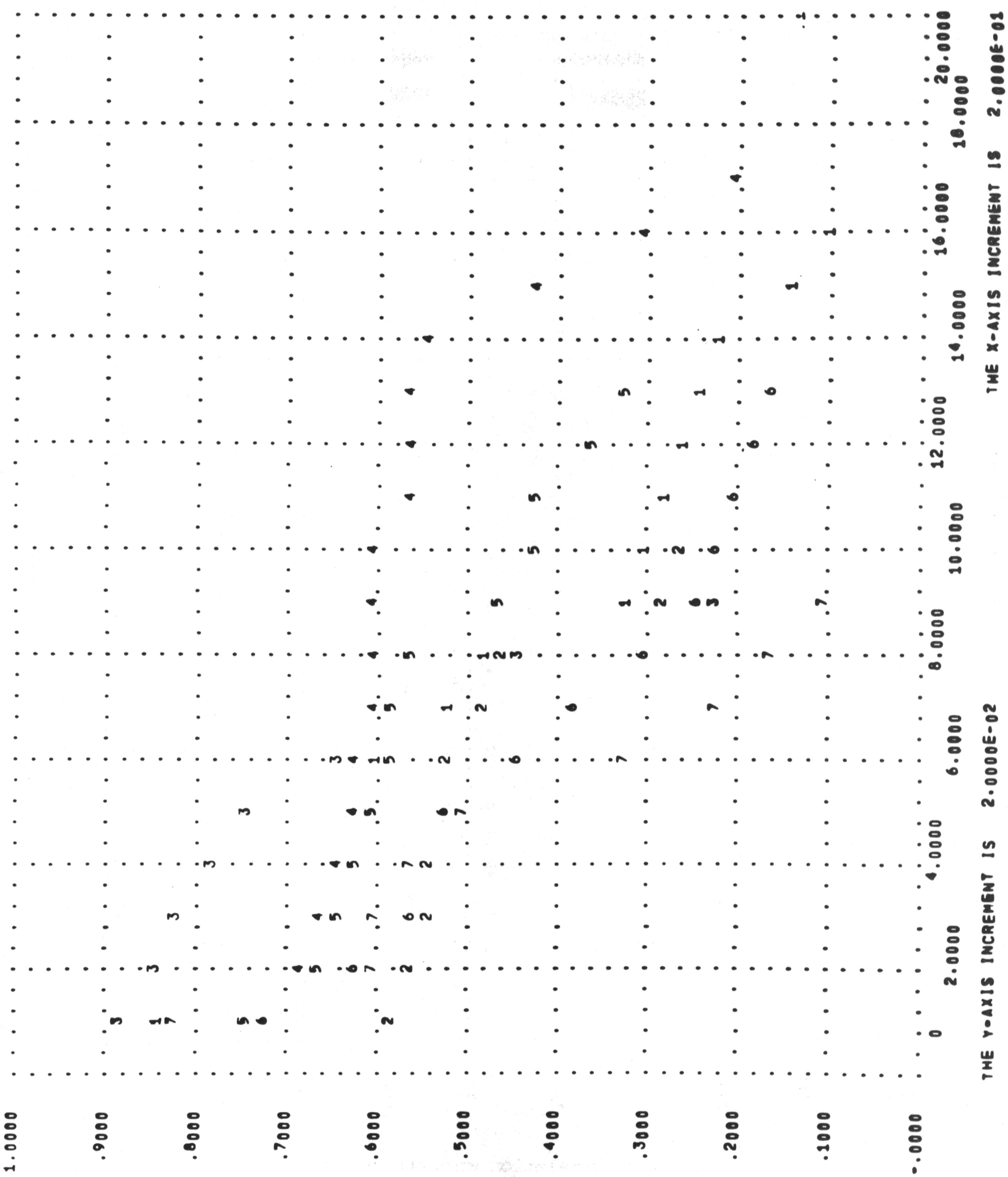
Fig. 10

a given document with its respective centroid can be read off on the ordinate, and the number of documents in each cluster is given by the abscissa of the right-most entry for the given cluster. Thus Fig. 11 shows for example, that cluster 4 contains 17 documents, while cluster 2 contains only 10. The more useful clusters are generally those where all documents have high correlations with their respective centroid.

C) Cluster Searching and Evaluation

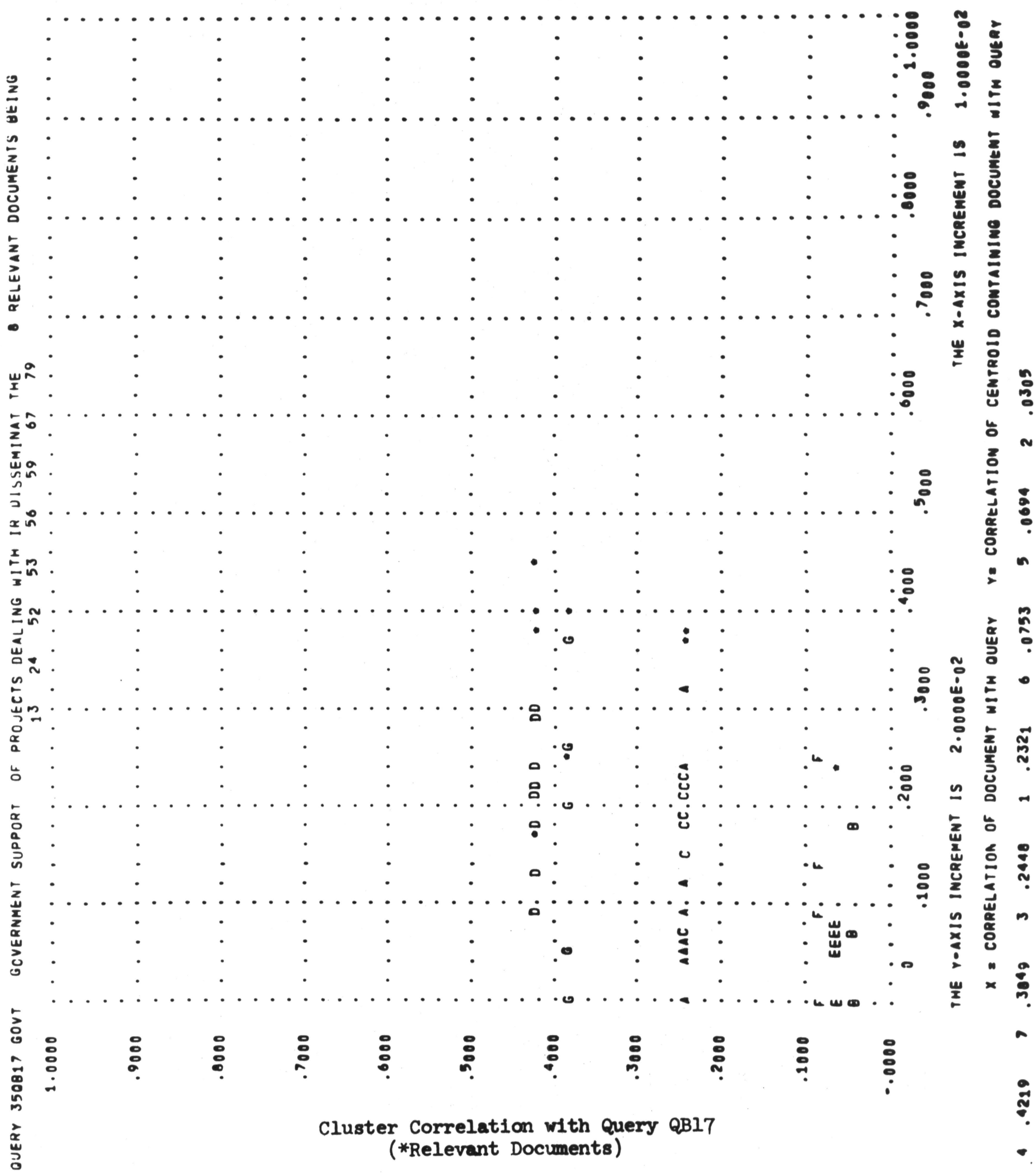
After a given document collection is available in clustered form, the search operation can be conducted in two steps: an incoming request is first correlated with the centroid vectors of all the clusters. For the collection of 82 documents previously used as an example in Fig. 11, this requires seven comparisons for each request. This preliminary operation is followed by a match of each search request with the individual documents included in the n clusters exhibiting the highest correlation with the given request, or alternatively with the documents in all clusters for which the centroid-request correlation exceeds a given threshold.

A typical cluster match is shown in Fig. 12 for the collection of 82 documents in documentation processed against request QB17. The ordinate corresponds to the correlation coefficient between the request and each of the seven centroid vectors, labelled from A to G for centroids 1 to 7 respectively. Thus, the highest correlation with the request (0.42) was obtained for centroid 4 (labelled D), the next highest (0.38) for centroid 7 (labelled G), and so on. The abscissa, on the other hand, represents the correlation coefficient between the request and each of the individual documents within the various clusters. Documents which are relevant to



Correlation of Clustered Documents with their Respective Centroid Vectors (82 documents - 7 clusters - 5 overlapping documents)

Fig. 11



Cluster Correlation with Query QB17
(*Relevant Documents)

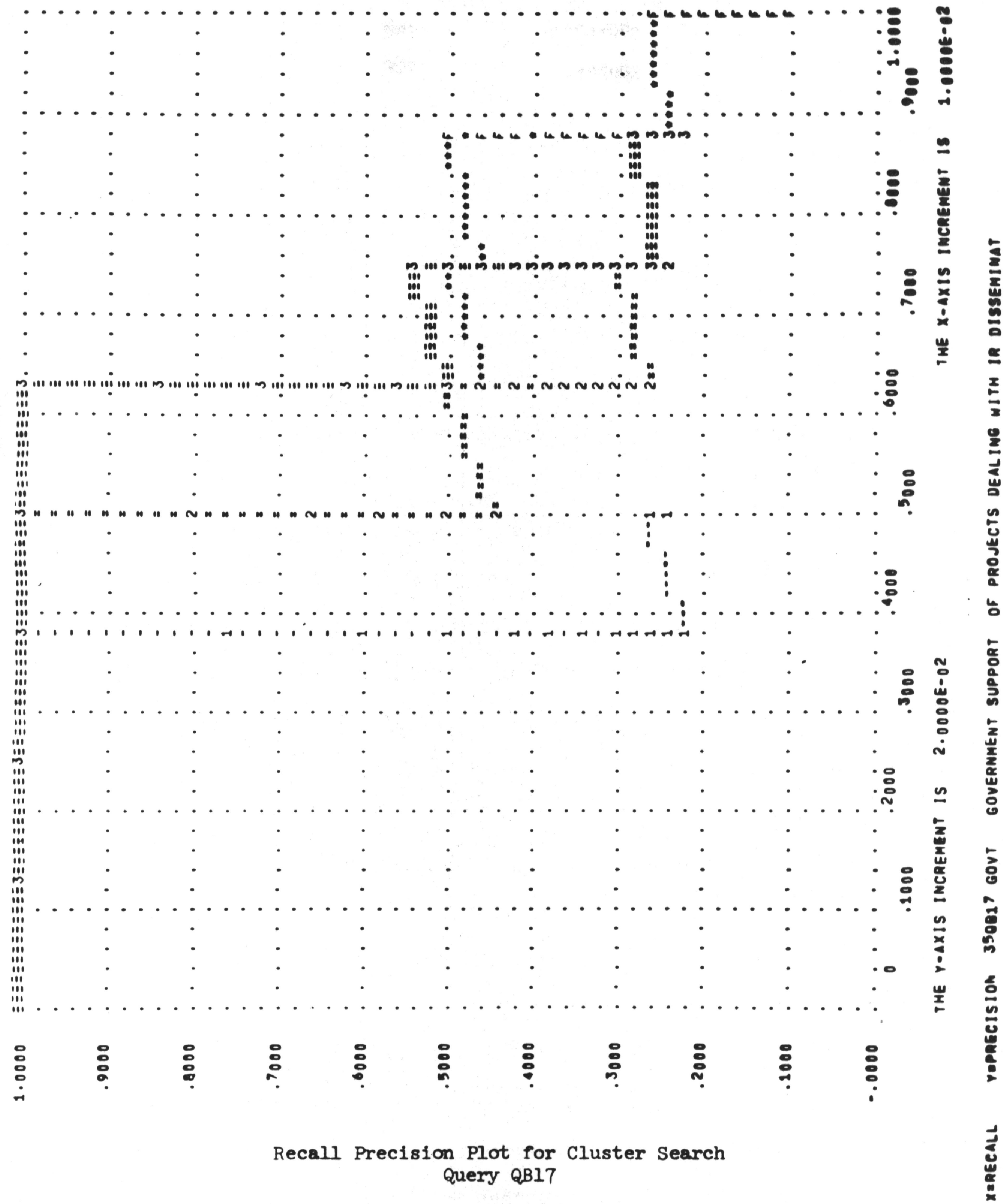
Fig. 12

the given request, as determined outside of the system by human subject experts, are identified by an asterisk in the graph of Fig. 12. Thus, there are four relevant documents in cluster D (the cluster with the highest correlating centroid with the request), and two additonal ones in cluster G (the cluster with the next highest correlation).

Assuming that the search strategy chosen requires that clusters with a centroid correlation exceeding 0.30 be individually examined, the seven centroid comparisons must then be followed by 17 comparisons for cluster D, plus 9 comparisons for cluster G (only 12 characters appear in Fig. 12 for cluster D, and only 7 for cluster G, since several documents with identical correlation coefficients are represented by a single character). Documents included in clusters other than D and G are never examined, thus reducing the search time to a fraction of that needed for the "full" search which consists in an examination of every document in the collection. At the same time, the partial search limits the number of relevant documents actually retrievable to those included in the first two clusters —— a total of 6 out of 8 relevant for query QB17, shown in the example. This accounts for the <u>recall ceiling</u>, or limitation in the amount of retrievable relevant material inherent in all partial search algorithms; clearly, relevant items which are never examined in the first place can of course never be retrieved.

The evaluation of the effectiveness of the cluster search algorithm can be based on the standard <u>recall</u> and <u>precision</u> measures, where recall is defined as the proportion of relevant matter retrieved, and precision as the proportion of retrieved material actually relevant. As in the other evaluation work carried out with the SMART system [6,7], manually derived, exhaustive relevance judgments are used in which the relevance of each document is determined with respect to each of the search requests. By varying the cut-off used to produce a variable number of retrieved documents, a number of recall-precision pairs are obtained which can then be displayed as a graph showing recall against precision. The recall-precision plots for the individual search requests can then be averaged and a single curve can be obtained representing the average performance of the system over many search requests. Recall-precision plots are particularly useful if it is desired to select search and analysis methods to fit certain operating ranges: thus, if it is desired to pick a procedure which favors the retrieval of <u>all</u> relevant material, then one must concentrate on the high recall region; similarly, if <u>only</u> relevant material is wanted, the high precision region is of importance. (In general, it is possible to obtain high recall only at a substantial cost in precision, and vice-versa [4,6,7].)

A typical recall-precision plot is shown for query QB17 in Fig. 13. Recall is plotted along the abscissa, and precision along the ordinate. Fig. 13 contains four superimposed curves: the curve labelled with 1's and single hyphens corresponds to a cluster search in which only a single cluster is examined (cluster D); the curve labelled with 2's and double hyphens represents the cluster search based on the examination of the two top clusters (clusters D and G); similarly, the curve labelled with 3's or triple hyphens is produced by an examination of the three clusters with the highest centroid correlations (D, G, and C). For purposes of comparison, the results of the full search in which all documents are examined, is also shown in Fig. 13, represented by F's and asterisks. When several of the curves have identical values and ought therefore to be superimposed in the output of Fig. 13, only the curve of highest rank is shown, the ranking going from F, to 1, 2, and 3 in that order. For example, in



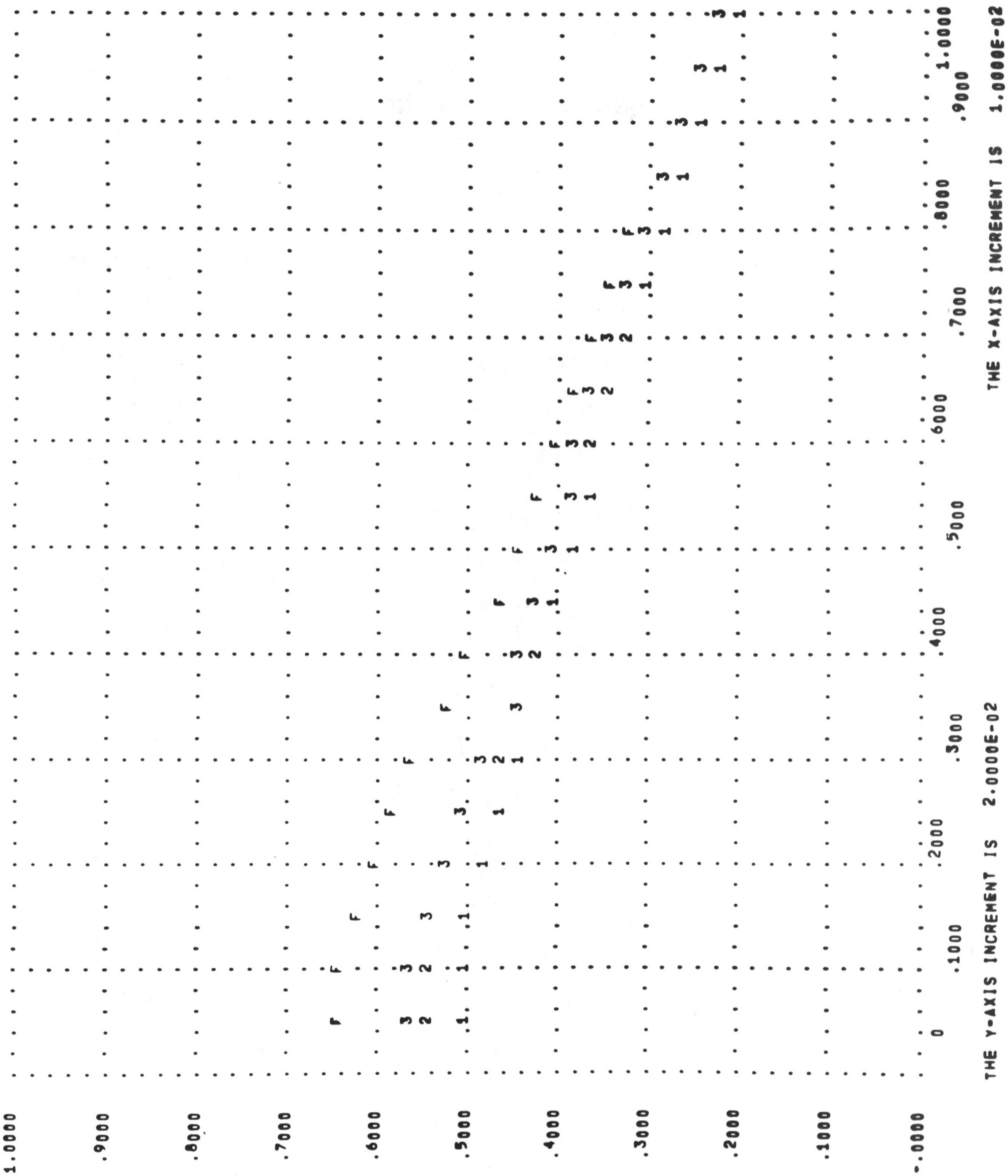
Recall Precision Plot for Cluster Search
Query QB17

Fig. 13

Fig. 13, all four curves exhibit the same recall performance up to a value of 0.375. This accounts for the single curve labelled with 3's in that region.

It may be noted that the curve corresponding to a single cluster search stops at a point where the recall is 0.5, and the precision 0.23; these values are obtained when all 17 documents in the first cluster are examined. Higher recall, or lower precision values are not possible in this case, since cluster D does not contain additional items. For the two-cluster search, the limits are reached when the recall is 0.75 and the precision 0.24; finally, for the three-cluster search, the values are 0.875 and 0.2188, respectively. The full search, corresponding to an exhaustive examination of the collection is not subject to any recall ceiling below 1, since all relevant documents can then be compared with the request and retrieved. For the full search, the value of the precision is 0.2286 at recall 1. In the example of Fig. 13, the precision of the three-cluster search is actually equal or superior to that of a full search up to a recall of 0.75.

Performance figures for the cluster searches are shown averaged over 35 search requests in the output of Fig. 14. The curves labelled with 1's, 2's, and 3's again represent 1-cluster, 2-cluster, and 3-cluster searches, and F's are used for the full search. It may be noted that the precision difference between 3-level and full search amounts to less than ten percent for most recall levels, and actually becomes much smaller than that for high recall values. The average maximum precision difference between the one-cluster and full searches is only about fifteen percent (at recall of 0.10), and diminishes for higher recall values. Obviously,



Averaged Recall-Precision of Cluster Search
showing Comparison with Full Search
(averages over 35 requests)

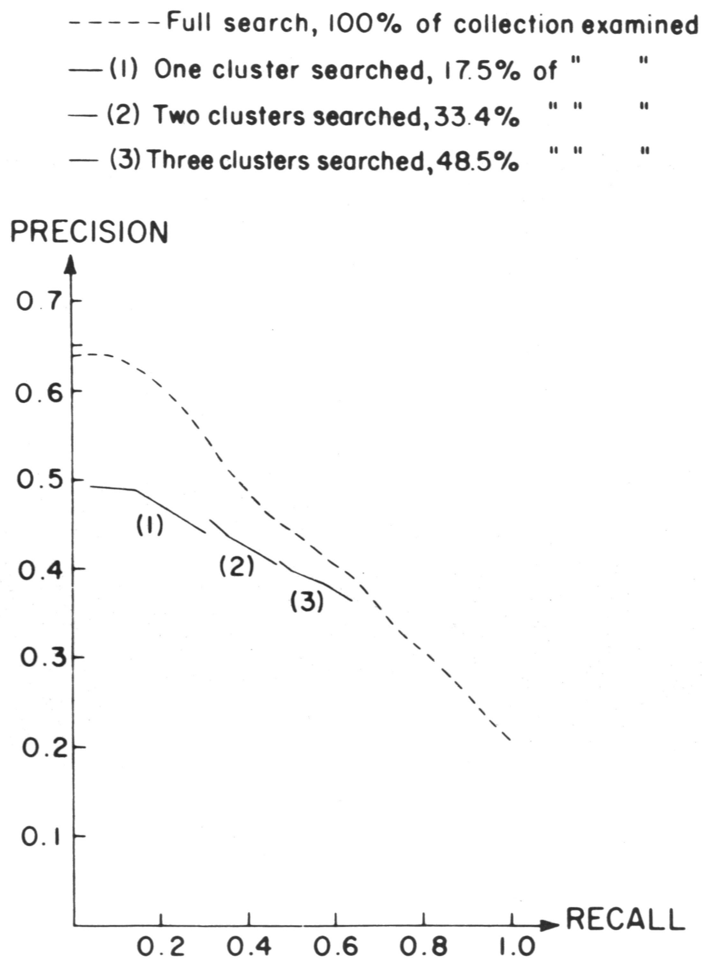
Fig. 14

the performance of the cluster search improves when additional clusters (beyond the first) are examined, but the improvement is modest for the collection used in the example.

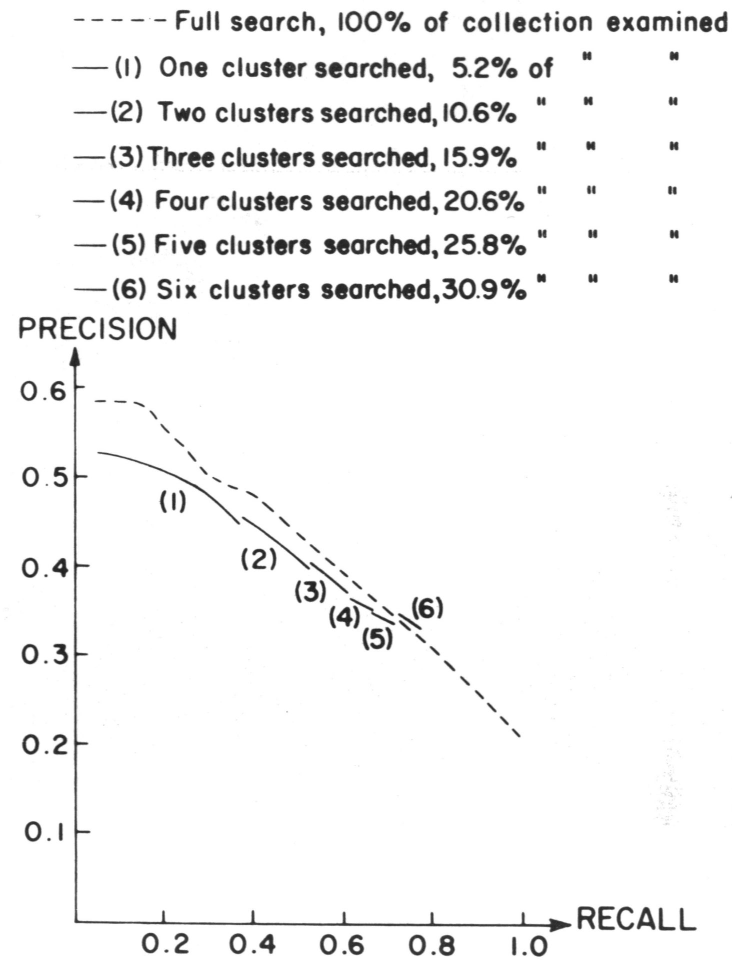
The output graph of Fig. 14 may not be directly usable for the evaluation of systems performance, since the recall ceiling is not shown for the cluster searches. The curves in fact represent averages over a variable number of requests, depending on the recall level considered. A more useful evaluation output is shown in Fig. 15 for two collections of 82 documents in documentation, and 200 documents in aerodynamics, respectively. An n-cluster search is represented by a curve labelled n , and the curves for the cluster searches terminate at their respective recall ceilings. For the documentation collection the average recall ceilings are 0.31, 0.47, and 0.64 for the one-, two-, and three-cluster searches, respectively.

It is clear from the output of Fig. 15, that nothing but a full search will avail, if very high recall is demanded; on the other hand, for average recall levels, a two- or three-cluster search, involving only about one fifth of the number of matches compared with those needed in a full search, appears to result in very little less in precision (for the aerodynamics collection a 6-cluster search, involving about 31 percent of the total collection, is actually found to be superior to a full search); for low recall levels, the precision of a one-cluster search is from five to fifteen percent smaller than that of a full search.

If these results are taken as typical for document collections in other technical areas as well, cluster searching appears to offer large savings in search time, at no substantial loss in recall and precision for all searches not requiring either a very high recall performance, or a very high precision.



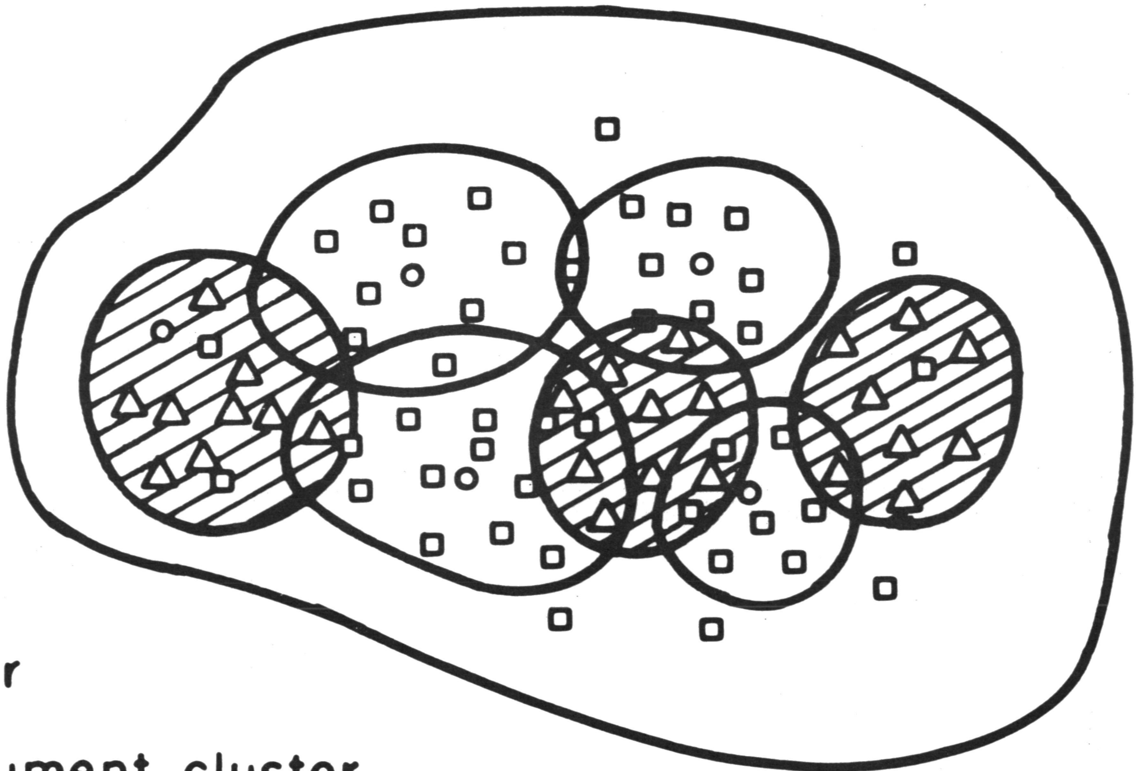
Documentation Collection, 82 documents, results averaged over 35 requests, using abstracts and thesaurus dictionary.
Cluster generation produced 7 clusters, with mean of 12 documents in each.



Aerodynamics Collection, 200 documents, results averaged over 42 requests, using abstracts and word stem dictionary.
Cluster generation produced 23 clusters, with mean of 13 documents in each.

Cluster Search Evaluation

Fig. 15



□ Document vector

○ Centroid of document cluster

△ Request vector

/// Query clusters

Clustered Document Space with Request Clusters

Fig. 16

The preceding discussion, based on preconstructed document clusters, can be extended to partial searches involving other types of clustering strategies. If, for example, the document collection under consideration changes very rapidly, and the retrieval system is very active, it may not be useful to operate with standard document clusters, since the quality of these clusters is then bound to deteriorate as time goes on. In such a case it may be more appropriate to operate with <u>clusters of requests</u> previously processed by the system, rather than with document clusters. Such a situation is pictured in Fig. 16 where the cross-hatched request clusters are superimposed on the document cluster space. A document cluster is then assumed to exist in association with each request cluster, consisting of documents previously found useful in answering the corresponding requests. A two-level search can then be performed in the following manner:

a) a new incoming request is first compared with the centroid vectors of all request clusters;

b) the documents associated with the highest matching request clusters are then individually compared with the new requests, and documents with a sufficiently high correlation coefficient are retrieved as before.

The request clustering process may be expected to be particularly efficient in situations where a homogeneous user population is to be serviced, in which case, new incoming requests might be similar in nature to requests previously handled for other customers. If, on the other hand, the set of request clusters used produces the same configuration in the document space as the original set of document clusters —— a situation which does not obtain in the example of Fig. 16 —— then the request clustering method will offer few advantages. The request clustering procedure remains to be evaluated more fully. [14]

3. Relevance Feedback

A) Overall Process

A variety of different methods can be used in an attempt to have the customer participate in the search process. These procedures range from relatively simple dictionary print-out routines, where dictionary excerpts supplied to the user serve as an aid in rephrasing poorly worded search requests, to more sophisticated methods in which the reformulation of the requests is automatically performed based on feedback information obtained from the user population. [15, 16]

The relevance feedback process about to be described is particularly well-suited to a time-sharing computer organization and to the simple console equipment likely to be available to the customers, since it requires only a minimum of interaction with the user, and places most of the burden on internally stored routines. Specifically, an initial search is first performed for each request received, and a small amount of output, consisting of some of the highest scoring documents, is presented to the user. Some of the retrieved output is then examined by the user who identifies each document as being either relevant (R) or not relevant (N) to his purpose. These relevance judgments are later returned to the system, and used automatically to adjust the initial search request in such a way that query terms or concepts, present in the relevant documents are promoted (by increasing their weight), whereas terms occurring in the documents designated as nonrelevant are similarly demoted. [17, 18] This process produces an altered search request which may be expected to exhibit greater similarity with the relevant document subset, and greater dissimilarity with the nonrelevant set.

The altered request can next be submitted to the system, and a second search can be performed using the new request formulation. If the system performs as expected, additional relevant material may then be retrieved, or, in any case the relevant items may produce higher correlations with the altered request than with the original. The newly retrieved items can again be examined by the user, and new relevance assessments can be used to obtain a second reformulation of the request. This process can be continued over several iterations, until such time as the user is satisfied with the results obtained.

The actual method used for the request alteration consists in picking at each point that request formulation which maximizes the difference in request-document correlation between relevant and nonrelevant document subsets. Specifically, if D_R is the nonempty document subset designated as relevant, then an optimal query is the one which provides the maximum discrimination of the subset D_R from the rest of the collection $(D-D_R)$. More formally, if $\sigma(\underline{q},\underline{d})$ is the distance function (correlation method) used in the matching process between query $\underline{q}$ and document $\underline{d}$, then the optimal query $\underline{q}_o$ may be defined as that query which maximizes the function

$$F = \underset{\underline{d}^{(i)} \notin D_R}{\tilde{\sigma}} (\underline{q},\underline{d}^{(i)}) - \underset{\underline{d}^{(i)} \in D_R}{\tilde{\sigma}} (\underline{q},\underline{d}^{(i)}),$$

where $\tilde{\sigma}$ is the average distance function, and decreasing distance implies stronger query-document correlation. [17]

In practice, the preceding equation is of no immediate use, even under the assumption that the optimal query $\underline{q}_o$ can be determined as a function of D and D_R, since knowledge of the set D_R (the relevant document subset obviates the need for retrieval. Instead of producing the optimal query $\underline{q}_o$ directly, it is then necessary to generate a series of approximations to $\underline{q}_o$, starting with some initial query which identifies a part of the set D_R. As new relevant documents are identified, the subset of known relevant documents approaches D_R, and the sequence of modified queries comes close to $\underline{q}_o$. One may hope that in practice only a few iterations will suffice for the average user; in any case, the rate of convergence is reflected in the stability of the retrieved set.

The query modification algorithm which produces an optimal query to differentiate the partial set of relevant documents identified by the user from the remaining documents may be written in the form:

$$\underline{q}_{i+1} = n_1 n_2 q_i + n_2 \sum_{i=1}^{n_1} \frac{\underline{r}_i}{|\underline{r}_i|} - n_1 \sum_{i=1}^{n_2} \frac{\underline{s}_i}{|\underline{s}_i|} \qquad (1)$$

where $\underline{q}_i$ is the i^{th} query of the sequence, $R = \{\underline{r}_1, \underline{r}_2, \ldots, r_{n_1}\}$ is the set of relevant documents retrieved in response to query $\underline{q}_i$, and $S = \{\underline{s}_1, s_2, \ldots, \underline{s}_{n_2}\}$ is the set of nonrelevant document vectors retrieved in response to $\underline{q}_i$. [17] The specification of the sets R and S constitute the feedback from the user after the i^{th} iteration of the process.

The programmed experimental feedback system uses a somewhat more general modification algorithm which allows additional variations in several parameters, as follows:

$$\underline{q}_{i+1} = \alpha \underline{q}_i + \beta \underline{q} + \gamma \sum_{i=1}^{n_1} c_i \underline{r}_i + \delta \sum_{i=1}^{n_2} c_i \underline{s}_i \quad , \qquad (2)$$

where α, β, γ, and δ are variable weighting parameters; $\underline{q}$ is the initial query before any alteration; and c_i is either set equal to 1 for all i, or to the magnitude of the correlation coefficient between query $\underline{q}$ and document $\underline{d}^{(i)}$, depending on the setting of an additional variable parameter. The first two terms on the right-hand side of equation (2) permit the generation of $\underline{q}_{i+1}$ either from $\underline{q}_i$, or from $\underline{q}$, and the parameters c_i present in the last two terms are used to alter more heavily concepts which are derived from relevant documents exhibiting a high correlation with the query, than others included in documents which are further removed from the original query.

Evaluation results for the feedback procedure are given in the next section.

B) Feedback Evaluation

An example of the request modification process is shown in Fig. 17 for request Q147 processed against a collection of 200 documents in aerodynamics. The concept numbers and weights derived for the original request by the machine process are given in Fig. 17(a). Following a search with the original request, the user identifies document No. 94 as relevant. The altered request produced by the addition of new terms from document 94 is shown in Fig. 17(b). Several of the original concepts are reinforced in the process, (for example, concept 2558), while many others appear for the first time in Fig. 17(b). When this altered request is processed, the user next identifies as relevant documents 94, 90, and 95, thereby producing a new altered query represented in Fig. 17(c). When this last query is used, the set of relevant documents increases to four, consisting of documents

Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights
1282	12	1307	12	1534	12	1597	12
1626	12	2308	12	2450	12	2547	12
2552	12	2558	12	2576	12		

a) Initial Query Vector Q_o for Query Q147

Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights
60	12	224	12	358	12	411	12
633	12	639	12	1010	12	1109	12
1263	12	1282	12	1307	12	1308	12
1534	12	1545	48	1597	12	1626	12
1662	12	1663	12	1665	24	1794	12
1894	12	1915	12	1930	24	1936	12
1950	12	1981	12	1986	24	2011	48
2034	12	2068	36	2100	12	2163	12
2173	12	2209	12	2226	12	2278	48
2300	12	2308	12	2313	12	2335	24
2346	84	2363	48	2364	24	2370	12
2380	24	2388	12	2390	24	2393	12
2394	12	2411	36	2422	12	2450	12
2457	12	2473	12	2479	60	2496	24
2506	24	2507	12	2510	24	2521	12
2530	24	2536	24	2545	48	2547	36
2552	12	2577	48	2558	48	2566	24
2567	12	2571	60	2575	12	2576	48
2585	12	2586	48	2589	48	2594	24
2596	12	2597	12	2601	48	2603	60
2607	72	2619	72	2621	12	2622	24
2624	24	2626	120	2627	24		

b) Query Vector Q_1 after Identification of Relevant Document No. 94

Request Modification Process

Fig. 17

Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights
60	36	115	24	157	24	168	24
224	36	290	24	358	60	411	36
522	24	633	36	639	36	826	24
1010	36	1109	36	1200	24	1203	24
1206	24	1218	48	1221	48	1259	24
1263	36	1282	12	1307	12	1308	36
1534	12	1545	144	1597	36	1626	12
1644	48	1662	60	1663	60	1665	72
1750	24	1763	72	1765	24	1794	36
1818	72	1836	24	1888	24	1894	36
1915	36	1930	72	1936	36	1950	36
1981	36	1986	72	2011	144	2018	48
2034	36	2068	108	2100	36	2134	48
2163	26	2171	24	2173	60	2191	48
2192	24	2198	24	2209	36	2220	96
2224	72	2226	36	2278	240	2283	24
2300	108	2308	12	2313	36	2320	24
2335	72	2337	48	2346	252	2363	144
2364	72	2370	36	2380	72	2388	36
2390	72	2393	36	2394	36	2396	24
2399	96	2409	24	2410	24	2411	108
2422	36	2444	48	2450	36	2457	60
2465	24	2473	36	2477	24	2479	180
2496	192	2498	24	2501	24	2506	96
2507	84	2510	72	2514	24	2519	24
2521	36	2528	24	2530	72	2536	72
2542	48	2545	144	2547	84	2552	12
2557	216	2558	168	2566	96	2567	60
2571	276	2575	84	2576	144	2580	24
2581	24	2585	60	2586	168	2589	240
2594	96	2595	24	2596	60	2597	60
2599	96	2601	168	2603	228	2607	288
2611	48	2619	240	2621	36	2622	120
2623	96	2624	96	2626	408	2627	192

c) Query Vector Q_2 after Identification of Relevant Documents 94,90,95

Request Modification Process

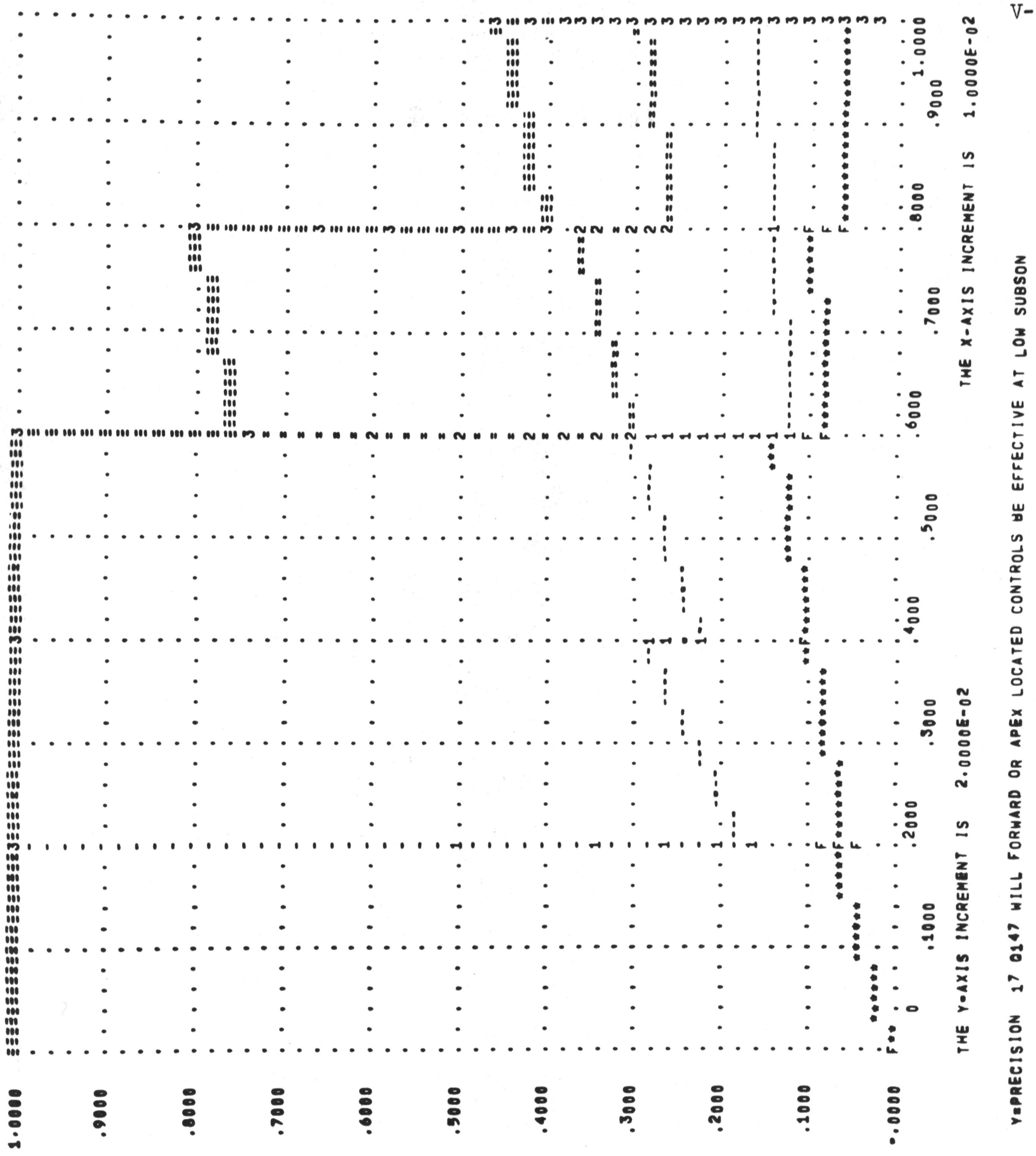
Fig. 17 (contd.)

Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights	Concept Numbers	Weights
60	72	115	60	157	60	168	60
224	72	290	60	358	168	411	72
522	60	633	72	639	72	826	60
852	36	1010	72	1109	72	1200	60
1203	60	1206	60	1218	120	1221	120
1259	60	1263	72	1282	12	1307	12
1308	72	1534	12	1545	288	1558	36
1579	108	1597	72	1626	12	1631	36
1644	120	1656	36	1662	132	1663	132
1665	144	1750	60	1763	180	1765	60
1794	72	1818	252	1836	60	1888	60
1894	72	1915	72	1930	144	1936	72
1950	72	1981	72	1986	144	2011	288
2018	120	2034	72	2068	216	2094	72
2100	72	2133	36	2134	120	2163	72
2171	60	2173	132	2187	36	2191	120
2192	60	2198	60	2209	72	2220	312
2224	216	2226	72	2241	36	2278	564
2283	60	2300	288	2308	12	2313	72
2320	60	2335	144	2337	120	2346	504
2363	288	2364	144	2370	72	2378	72
2380	144	2388	72	2390	144	2393	72
2394	72	2396	60	2399	240	2409	60
2410	60	2411	216	2422	72	2423	36
2444	120	2450	72	2457	132	2465	60
2467	36	2473	72	2477	60	2479	360
2496	444	2498	60	2501	60	2504	72
2506	204	2507	228	2510	144	2514	60
2519	60	2521	72	2528	60	2530	144
2536	144	2542	120	2545	288	2547	156
2552	12	2557	540	2558	348	2566	240
2567	132	2571	636	2575	192	2576	324
2580	60	2581	60	2585	132	2586	384
2589	600	2594	240	2595	60	2596	132
2597	132	2599	240	2601	348	2603	480
2607	648	2608	36	2611	156	2619	492
2621	72	2622	264	2623	240	2624	204
2626	840	2627	480				

d) Query Vector Q_3 after Identification of Relevant Documents 95,94,91,90

Request Modification Process

Fig. 17 (contd.)



Recall Precision Plot for Query Q147
(original query and three alterations)

Fig. 18

95, 94, 91, and 90. This generated the third modification of the original query, reproduced in Fig. 17(d). A comparison of Figs. 17(a) to (d) reveals a considerable increase in the number of concepts used, as well as a large increase in the concept weights.

The recall-precision plot produced by the feedback process for query Q147 is shown in Fig. 18 for the original query (represented by F's and asterisks), as well as for the three subsequent iterations (1's and single hyphens, 2's and double hyphens, and 3's and triple hyphens). It is seen in Fig. 18 how the recall and precision values improve from one iteration to the next, until a near perfect output is produced for the last iteration.

This same phenomenon can be observed in more detail in the tables of Fig. 19, containing a complete record of the process for query Q147. For each of the four iterations, an output ranking is given for the whole document collection. The documents are listed in decreasing correlation order together with the respective correlation coefficients, as well as recall and precision figures. The relevant document set, determined manually outside of the system, consists of documents 90, 91, 93, 94, and 95. For the original query, these relevant documents identified by an R in Fig. 19, receive ranks of 22, 76, 21, 14, and 41, respectively, for the sample collection of two hundred documents.

The user is now assumed to look at the top 15 documents retrieved, thereby identifying document 94 with rank 14 as relevant. This leads to the first modification with improved rankings of the relevant set. The top 15 now include three relevant items: 94, 90, and 95 with ranks 1, 7, and 10 respectively. A second iteration leads to further improvements in the rankings of the relevant set, and to the addition of relevant document 91 to the top 15. This generates the last query form, which in turn produces the near perfect ranking of the relevant document set (ranks 1, 2, 3, 5, and 11). The recall-precision figures included in Fig. 19 reflect the excellent performance of query Q147.

Average performance characteristics are shown in the recall-precision plot of Fig. 20 for the relevance feedback process, using 42 search requests with a collection of 200 documents in aerodynamics. In each case, it is assumed that the user looks at the top fifteen documents produced by the computer search, and identifies those that are relevant. This information is used to update the request using equation (2) with $\alpha=1$; $\beta=0$; $\gamma=1, 2, 3$ for the first, second, and third alterations, respectively; all $c_i=1$; and $\delta=0$. The increase in the value of γ from one iteration to the next is motivated by the thought that the user becomes increasingly more informed as he sees more output, and that his relevance judgments should therefore be weighted increasingly more heavily.

Fig. 20 shows the large increase in precision for each given recall value between initial searches and first feedback runs. A smaller increase is present between the first and second feedback runs, with very little increase thereafter. The same large-scale improvements are noted also for document collections in other subject areas. Fig. 21 shows relevance feedback data for three collections in computer science, aerodynamics, and documentation, averaged over 24, 42, and 35 requests, respectively. In each case, the increase between initial requests and first feedback runs is very large, and diminishes thereafter. The output of Fig. 21 suggests that if low-recall, high-precision performance is desired, a single feedback step may be sufficient; in the high recall region, additional iterative steps may be useful.

17 Q147 WILL FORWARD OR APEX LOCATED CONTROLS BE EFFECTIVE AT LOW SUBSON

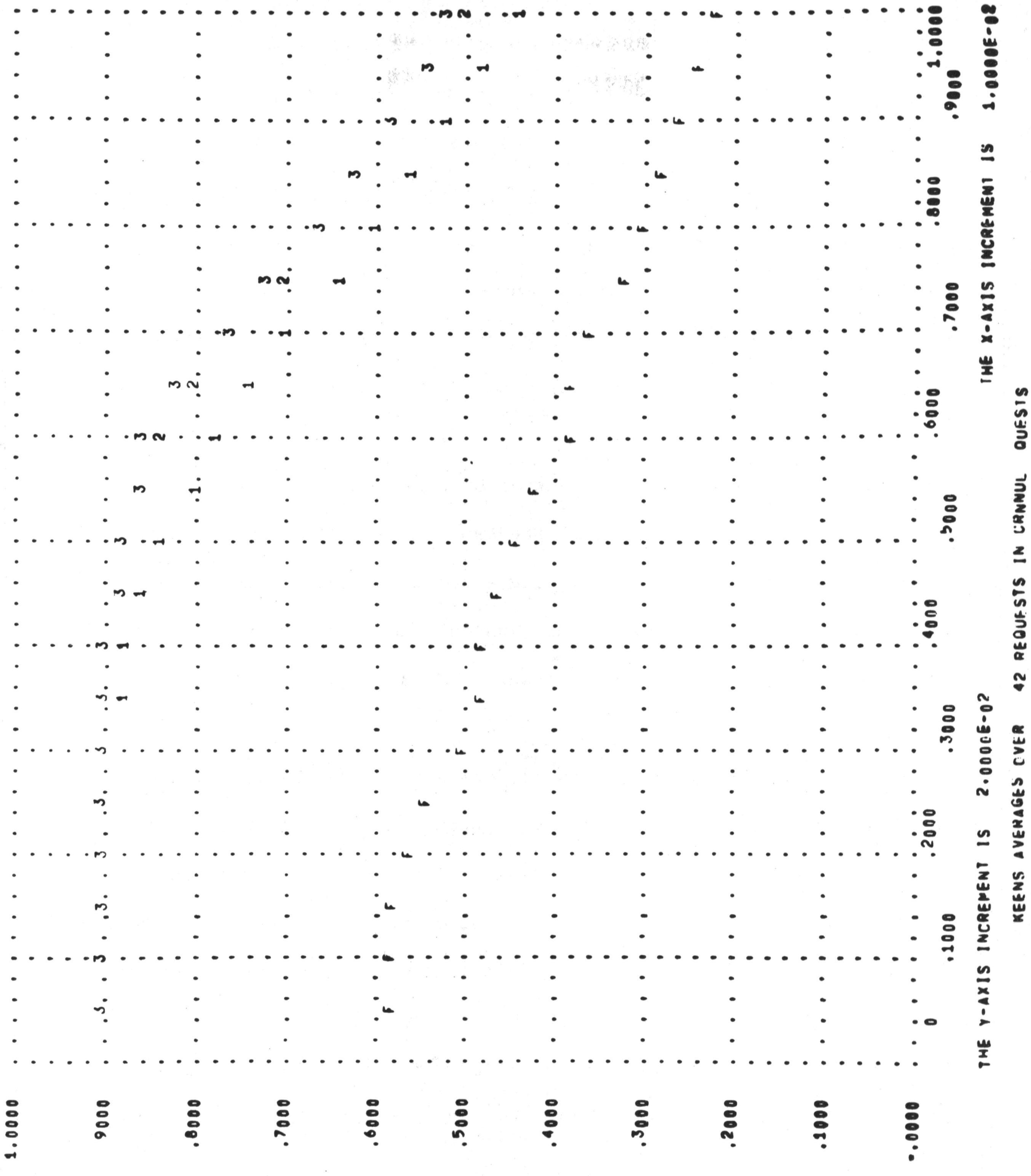
INITIAL SEARCH AND 3 FEEDBACK ITERS. THE 5 RELEVANT DOCUMENTS BEING 90 91 93 94 95

INITIAL			FEEDBACK ITERATIONS						RECALL				PRECISION			
			1		2		3									
RANK	DOC	CORR	DOC	CORR	DOC	CORR	DOC	CORR	INIT	1	2	3	INIT	1	2	3
1	109	.2224	94R	.9904	94R	.9348	94R	.8954	0	.2000	.2000	.2000	0	1.0000	1.0000	1.0000
2	60	.1928	81	.4909	95R	.6190	95R	.6675	0	.2000	.4000	.4000	0	.5000	1.0000	1.0000
3	121	.1570	195	.4802	90R	.5449	90R	.5881	0	.2000	.6000	.6000	0	.3333	1.0000	1.0000
4	192	.1388	123	.4233	195	.5023	195	.4929	0	.2000	.6000	.6000	0	.2500	.7500	.7500
5	193	.1359	80	.4144	81	.4494	91R	.4632	0	.2000	.6000	.8000	0	.2000	.6000	.8000
6	119	.1226	114	.3781	80	.3969	81	.4199	0	.2000	.6000	.8000	0	.1667	.5000	.6667
7	82	.1212	90R	.3742	114	.3850	80	.3808	0	.4000	.6000	.8000	0	.2857	.4286	.5714
8	24	.1177	193	.3685	193	.3795	111	.3787	0	.4000	.6000	.8000	0	.2500	.3750	.5000
9	86	.1151	122	.3394	123	.3778	114	.3758	0	.4000	.6000	.8000	0	.2222	.3333	.4444
10	123	.1149	95R	.3371	111	.3776	193	.3735	0	.6000	.6000	.8000	0	.3000	.3000	.4000
11	100	.1115	111	.3309	91R	.3557	93R	.3541	0	.6000	.8000	1.0000	0	.2727	.3636	.4545
12	146	.1044	64	.3201	109	.3378	123	.3538	0	.6000	.8000	1.0000	0	.2500	.3333	.4167
13	18	.0992	102	.3181	159	.3348	192	.3369	0	.6000	.8000	1.0000	0	.2308	.3077	.3846
14	94R	.0992	109	.3177	103	.3314	109	.3368	.2000	.6000	.8000	1.0000	.0714	.2143	.2857	.3571
15	167	.0930	82	.3077	192	.3281	159	.3329	.2000	.6000	.8000	1.0000	.0667	.2000	.2667	.3333
16	125	.0906	103	.3033	82	.3260	155	.3287	.2000	.6000	.8000	1.0000	.0625	.1875	.2500	.3125
17	163	.0900	78	.2936	93R	.3143	103	.3262	.2000	.6000	1.0000	1.0000	.0588	.1765	.2941	.2941
18	114	.0848	125	.2933	78	.3104	82	.3196	.2000	.6000	1.0000	1.0000	.0556	.1667	.2778	.2778
19	65	.0848	20	.2851	122	.3069	78	.3014	.2000	.6000	1.0000	1.0000	.0526	.1579	.2632	.2632
20	177	.0844	192	.2736	102	.2992	110	.2891	.2000	.6000	1.0000	1.0000	.0500	.1500	.2500	.2500
21	93R	.0836	124	.2714	64	.2986	122	.2884	.4000	.6000	1.0000	1.0000	.0952	.1429	.2381	.2381
22	90R	.0836	159	.2711	155	.2969	153	.2874	.6000	.6000	1.0000	1.0000	.1364	.1364	.2273	.2273
23	19	.0811	194	.2709	110	.2894	11	.2874	.6000	.6000	1.0000	1.0000	.1304	.1304	.2174	.2174
24	153	.0794	196	.2659	11	.2878	76	.2859	.6000	.6000	1.0000	1.0000	.1250	.1250	.2083	.2083
25	181	.0778	86	.2634	153	.2822	64	.2820	.6000	.6000	1.0000	1.0000	.1200	.1200	.2000	.2000
26	58	.0775	63	.2578	76	.2813	92	.2818	.6000	.6000	1.0000	1.0000	.1154	.1154	.1923	.1923
27	22	.0772	66	.2524	196	.2677	102	.2804	.6000	.6000	1.0000	1.0000	.1111	.1111	.1852	.1852
28	172	.0771	91R	.2496	20	.2668	152	.2783	.6000	.8000	1.0000	1.0000	.1071	.1429	.1786	.1786
29	200	.0726	110	.2485	194	.2657	161	.2670	.6000	.8000	1.0000	1.0000	.1034	.1379	.1724	.1724
30	64	.0726	93R	.2457	132	.2648	132	.2644	.6000	1.0000	1.0000	1.0000	.1000	.1667	.1667	.1667
31	3	.0715	11	.2426	152	.2637	196	.2601	.6000	1.0000	1.0000	1.0000	.0968	.1613	.1613	.1613
32	195	.0713	61	.2376	92	.2623	96	.2584	.6000	1.0000	1.0000	1.0000	.0938	.1563	.1563	.1563
33	144	.0696	77	.2362	125	.2618	29	.2568	.6000	1.0000	1.0000	1.0000	.0909	.1515	.1515	.1515
34	122	.0693	76	.2245	124	.2590	133	.2553	.6000	1.0000	1.0000	1.0000	.0882	.1471	.1471	.1471
35	63	.0671	132	.2236	86	.2585	194	.2537	.6000	1.0000	1.0000	1.0000	.0857	.1429	.1429	.1429
36	184	.0663	104	.2235	161	.2505	20	.2523	.6000	1.0000	1.0000	1.0000	.0833	.1389	.1389	.1389
37	34	.0657	153	.2194	61	.2491	86	.2489	.6000	1.0000	1.0000	1.0000	.0811	.1351	.1351	.1351
38	74	.0652	74	.2178	133	.2482	104	.2469	.6000	1.0000	1.0000	1.0000	.0789	.1316	.1316	.1316
39	113	.0639	69	.2055	29	.2480	61	.2436	.6000	1.0000	1.0000	1.0000	.0769	.1282	.1282	.1282
40	17	.0635	177	.2030	104	.2474	125	.2429	.6000	1.0000	1.0000	1.0000	.0750	.1250	.1250	.1250
41	95R	.0626	144	.2028	63	.2448	176	.2424	.8000	1.0000	1.0000	1.0000	.0976	.1220	.1220	.1220
42	75	.0626	67	.2004	96	.2401	124	.2413	.8000	1.0000	1.0000	1.0000	.0952	.1190	.1190	.1190
43	67	.0600	29	.1988	83	.2384	121	.2410	.8000	1.0000	1.0000	1.0000	.0930	.1163	.1163	.1163
44	140	.0599	60	.1976	77	.2347	83	.2374	.8000	1.0000	1.0000	1.0000	.0909	.1136	.1136	.1136
76	91R	.0389	19	.1436	160	.1745	160	.1793	1.0000	1.0000	1.0000	1.0000	.0658	.0658	.0658	.0658

1 03/22/67 25.12.41

Recall-Precision Tables for Q147
Showing Improvements in the Rankings
of the Relevant Documents

Fig. 19



Averaged Recall-Precision Plot for Relevance Feedback Process
(averages over 42 search requests -
200 documents in aerodynamics)

Fig. 20

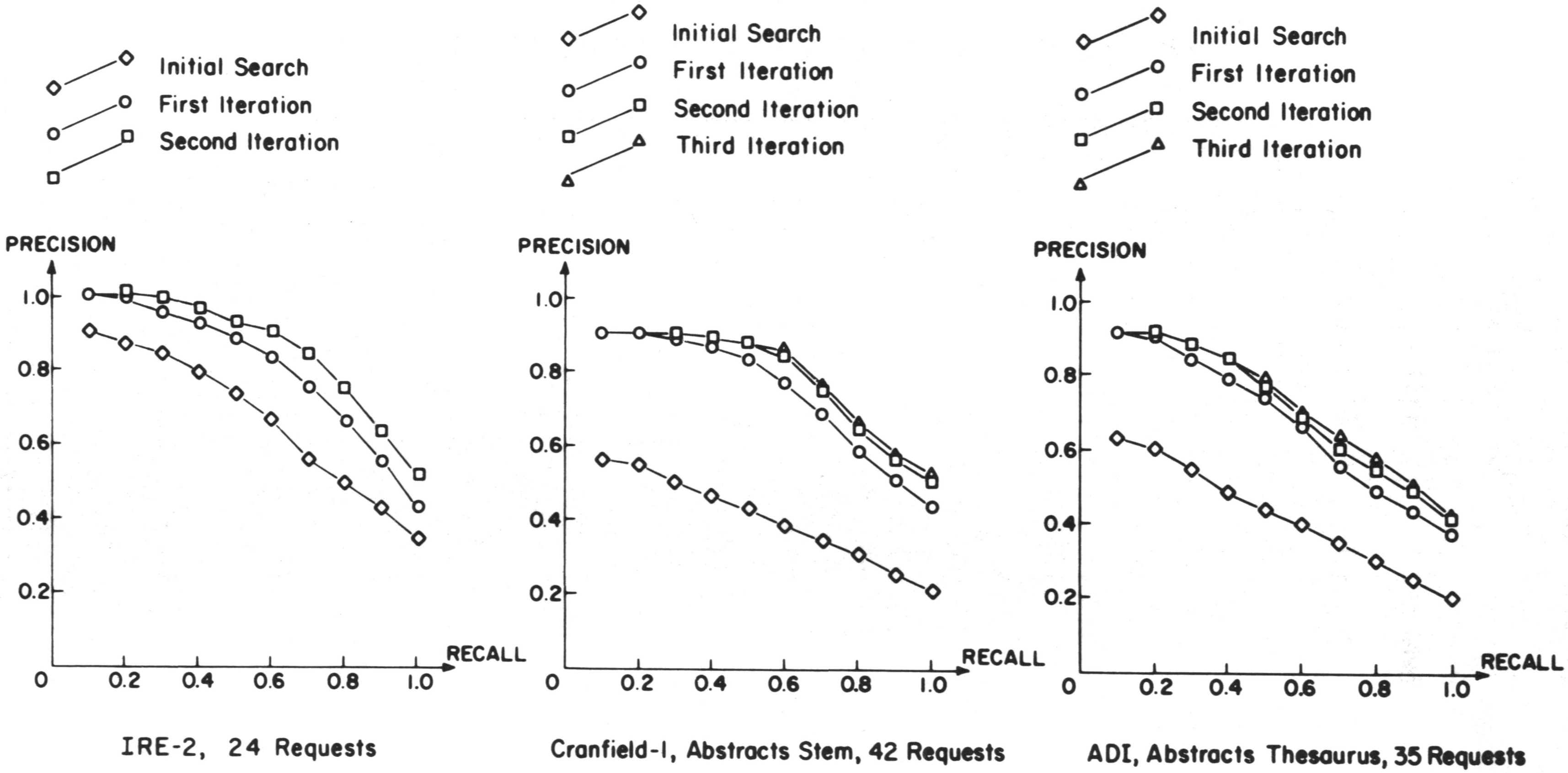
The output shown in Figs. 20 and 21 is produced with a single feedback strategy. Many of the changes suggested by the variable parameters of equation (2) still remain to be tested. Procedures must also be devised to cover the case where the user finds no relevant material to be returned, or where he finds only nonrelevant items. Finally, requests may have to be handled which cover several distinct subject areas. In that case, the feedback algorithm may not perform satisfactorily, since it is not then possible to approach a well-specified subject area in an optimal way.

4. Adaptive User-Controlled Multi-level Search

In a real-time environment, the two search strategies discussed in this report may be combined into a single overall search scheme based on cluster searches for fast turnaround, and on relevance feedback for the optimization of retrieval effectiveness. A possible systems design is suggested in Fig. 22. [19]

An attempt is first made to perform a request cluster search for each incoming search request, since this type of search may be expected to require the smallest number of comparison operations. If the request cluster process reveals relevant items, the relevance feedback process is used next. If no relevant items are found, however, a document cluster search is tried next, followed again by the relevance feedback method. Eventually, a full search may be tried, assuming that a high recall need exists, and that the two cluster searches are not successful in retrieving relevant material.

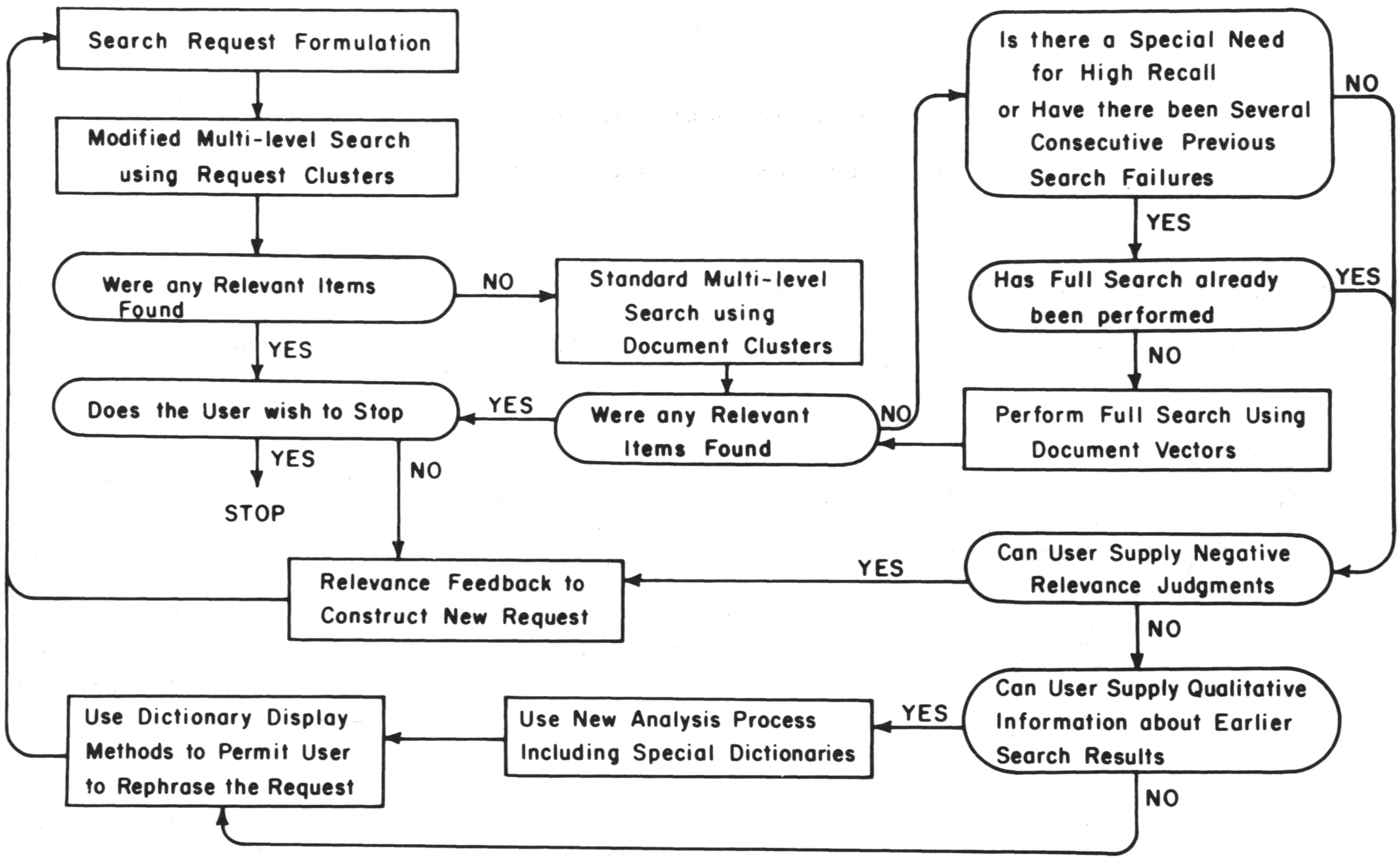
If only negative relevance judgments are available, a negative feedback algorithm may be used. Finally, if all else fails, qualitative



Comparison of Initial Search with Iterated Search

Process Using Relevance Feedback

Fig. 21



Sample Search Strategy Using Multi-level Searches and Relevance Feedback

Fig. 22

information may be available from the user, suggesting the use of phrase procedures or hierarchical expansions of the type included in the SMART system to broaden or narrow the area covered by a given request. [7,8] Dictionary display methods may also be used to help the user in rephrasing his request if the automatic relevance feedback method does not produce the desired results. [16]

This proposed real-time search strategy and others like it remain to be tested under operational conditions.

References

[1] M. E. Stevens, Automatic Indexing: A State of the Art Report, National Bureau of Standards, Monograph 91, Washington, 1965.

[2] M. E. Stevens, V. E. Giuliano, and L. B. Heilprin, editors, Statistical Association Methods for Mechanized Documentation, National Bureau of Standards, Publication 269, 1965.

[3] M. M. Henderson, Bibliography on Evaluation of Information Systems, National Bureau of Standards, July 1965.

[4] C. W. Cleverdon, J. Mills, and M. Keen, Factors Determining the Performance of Indexing Systems, Vol. 1-Design, Vol. 2-Test Results, Aslib-Cranfield Research Project, Cranfield 1966.

[5] V. E. Giuliano, and P. E. Jones, Study and Test for a Methodology for Laboratory Evaluation of Message Retrieval Systems, Arthur D. Little Inc., Report ESD-TR-66-405, August 1966.

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

[7] M. E. Lesk and G. Salton, Design Criteria for Automatic Information Systems, Information Storage and Retrieval, Scientific Report No. ISR-11 to the National Science Foundation, Dept. of Computer Science, Cornell University, June 1966.

[8] G. Salton and M. E. Lesk, The SMART Automatic Document Retrieval System — An Illustration, Communications of the ACM, Vol. 8, No. 6, June 1965.

[9] G. Salton, et al., Information Storage and Retrieval, Reports to the National Science Foundation, Nos. ISR-7, ISR-8, ISR-9, ISR-11, Harvard University and Cornell University, 1964-1966.

[10] R. E. Bonner, On Some Clustering Techniques, IBM Journal of Research and Development, Vol. 8, No. 1, January 1964.

[11] H. Borko and M. D. Bernick, Automatic Document Classification, Journal of the ACM, Vol. 10, No. 2, April 1963.

[12] R. M. Needham and K. Sparck Jones, Keywords and Clumps, Journal of Documentation, Vol. 20, No. 1, March 1964.

[13] J. J. Rocchio Jr., Document Retrieval Systems — Optimization and Evaluation, Harvard University Doctoral Thesis, Report No. ISR-10 to the National Science Foundation, Chapter 4, Harvard Computation Laboratory, March 1966.

References (continued)

[14] V. R. Lesser, A Modified Two-Level Search Algorithm Using Request Clustering, Report No. ISR-11 to the National Science Foundation, Section VII, Dept. of Computer Science, Cornell University, June 1966.

[15] R. M. Curtice and V. Rosenberg, Optimizing Retrieval Results with Man-machine Interaction, Report, Center for the Information Sciences, Lehigh University, Bethlehem, 1965.

[16] J. J. Rocchio, Jr., and G. Salton, Information Search Optimization and Iterative Retrieval Techniques, Proceedings of the 1965 Fall Joint Computer Conference, Spartan Books, 1965, pp. 293-305.

[17] J. J. Rocchio, Jr., Document Retrieval Systems — Optimization and Evaluation, Report No. ISR-10 to the National Science Foundation, Section III, Harvard Computation Laboratory, March 1966.

[18] W. Riddle, T. Horwitz, and R. Dietz, Relevance Feedback in an Information Retrieval System, Report No. ISR-11 to the National Science Foundation, Section VI, Dept. of Computer Science, Cornell University, June 1966.

[19] E. M. Keen, Semi-Automatic User-Controlled Search Strategies, Fouth Annual Colloquium on Information Retrieval, Philadelphia, May 1967.