X. A Relevance Feedback System Employing a Dynamically Evolving Document Space

M. C. Davis, M. D. Linsky, M. V. Zelkowitz

Abstract

Methods for improving precision and recall in information retrieval have been based mainly on query modification or temporary document transformations. The present study investigates results obtained when, in addition to modifying the query, the document collection is considered as a dynamically changing space which is continually improved to reflect, more accurately, the contents of the documents it contains. This alteration is achieved by reclustering the documents based on relevance feedback, so that future queries can benefit from the results of processing previous queries.

1. Introduction

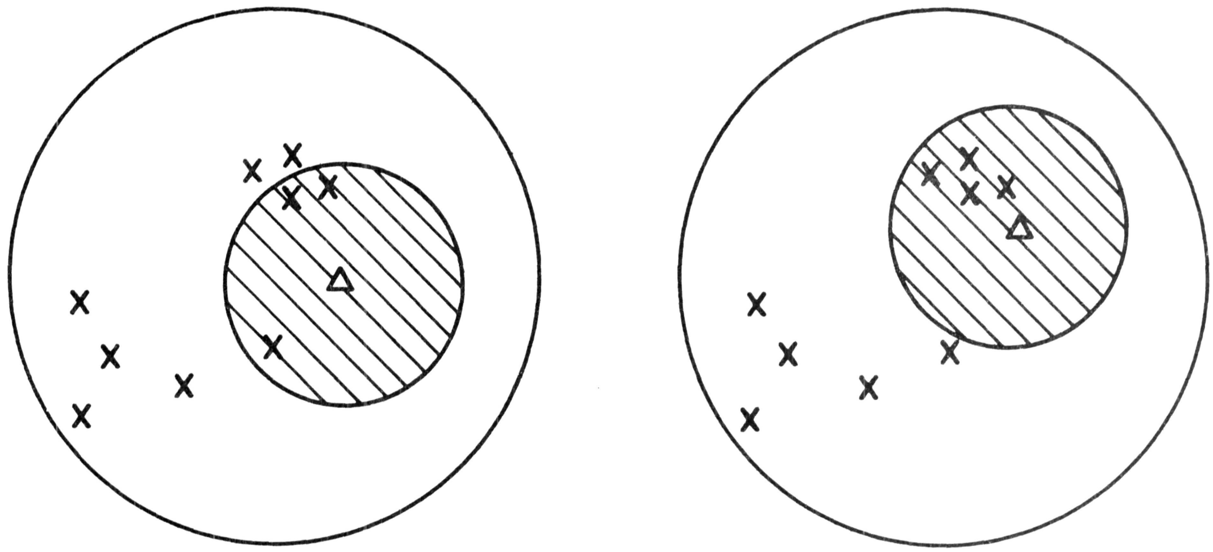
Information retrieval is fundamentally concerned with the selective retrieval of information which is pertinent to an inquiry from a large source of data. A comprehensive manual search covering even a small portion of available information is clearly impossible when dealing with a large library containing several million volumes. Current card catalogue oriented systems have proved to be useful tools towards the realization of more efficient, exhaustive scanning of information files, but intrinsic difficulties resulting from requirements imposed upon such systems by the alphabetical nature of these files either renders them unwieldy or incomplete. Innovations in the computing field have led to the notion that the retrieval problem can pragmatically be coped with only by using computing devices programmed to simulate personal inspection of possible relevant information. The SMART document retrieval programs are designed to accomplish such a simulation.

Basic SMART retrieval involves two major procedures. Initially, syntactic and semantic considerations are employed to automatically construct a concept vector with numerical components which will essentially act as the query itself when future reference is made to that query. When a query is processed, its concept vector is compared with the concept vectors of all of the documents in the collection (these document vectors are derived in a manner analogous to that for the concept vector for queries) and the cosine correlation, a measure of the similarity among queries and documents, is obtained. It is assumed that the probability of relevance of a given document to the query at hand is greatest for those documents whose correlation with this query is highest. Thus, the user will be presented with identification numbers of the documents with the highest correlations.

Due to the syntactic and semantic impreciseness of the English language as well as the user's possible uncertainty pertaining to the exact information which he is seeking, standard means of condensing or reducing documents by automatic procedures such as, for example, statistical term associations and frequency counts of particular words and phrases, are not definitive enough to produce a space which is an exact image of the original documents and queries. Thus, it has been hypothesized that systematic alteration of these indexing products based on user relevance feedback, can be employed to rectify the effects of any misinterpretations of user intent or document emphasis, to produce an "improved" or "refined" document space. For instance, studies have been made to evaluate systems which require that the user return judgments indicating which of the retrieved documents are of value to him. Based on these personal relevance judgments, the system then processes a modified query which reflects the feedback indications. That is, more emphasis is placed on documents which bear a marked similarity to the documents previously found to be relevant. The expected improved results, which have been demonstrated in [1], indicate that more relevant documents can thus be retrieved.

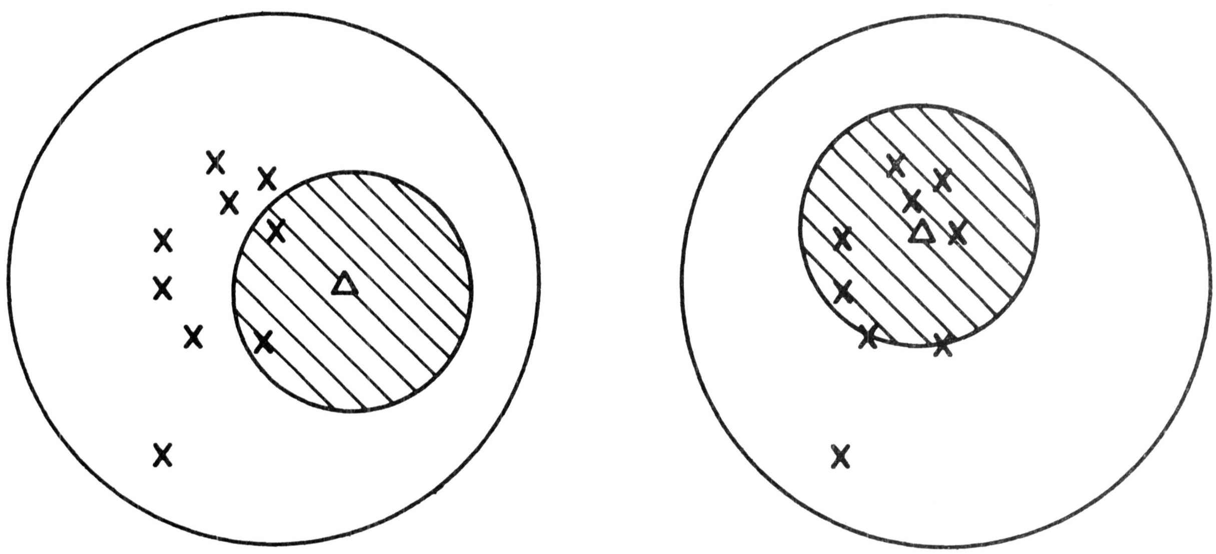
There exists a mathematical justification to support the expectation that such query modification will result in more effective retrieval. Since documents and queries are vectors with numerical components, they can be considered as points in a vector space. SMART normally retrieves all documents in the vector space which lie "close" to the query (see Fig. 1(a)). Hopefully, modifications based on relevance feedback can be used to move the query to a new position in the space. Ideally, a greater density of relevant documents will be centered about this portion of the space.

Though such a query modification does rectify to some extent the imperfections in the concept vectors corresponding to the user's queries, it has no effect on the document space itself. Any inadequacies in the original document space will exist throughout the life of the collection. It is, therefore, contended that the document space must also be altered if optimal results are to be obtained.



a) Typical SMART Retrieval

b) Typical SMART Retrieval with Relevance Feedback



c) Typical Retrieval from Modified Document Space

d) Typical Retrieval from Modified Document Space with Relevance Feedback

Retrieval Illustrations

Fig. 1

Implementation of a SMART-like system with the inclusion of the above query alteration technique will be notably unsuccessful in handling situations where relevant documents are clustered about distinct points which are distant from one another as in the illustration of Fig. 1. The query, when altered, will be moved to a position which is close to or within one cluster, but far from the second cluster of relevant documents. This second group of documents will, therefore, be totally ignored. The assumption that distinct documents which are found to be relevant to a given inquiry are in fact interrelated, leads to the contention that a reclustering of the document space based on relevance feedback data, is highly desirable. If this is done, determination of a single relevant document will easily facilitate the recovery of others. Whereas query modification involves a temporary change, since it is unlikely that a given query will exactly duplicate another one submitted at a future date, the proposed document space revisions are permanent in the sense that each updated document space is a refined version of the current space, not the original one determined by SMART.

This report is concerned with the reclustering of documents within a dynamic document space and the corresponding effects of these modifications on retrieval results. Control cases are examined to provide a basis for the evaluation of the effectiveness of the proposed method.

2. Proposed Study

The concept of a dynamic document space is not in itself novel. The work of Friedman, Maceyak, and Weiss also involved the clustering of relevant documents [1]. However, unlike the presently proposed scheme which retains the effects of the document space modifications during the processing of future queries, the document space treated in [1] reverts to its original form before the processing of the next query.

With the proposed system, the methods which control the successive alterations of the document space are based upon the following assumptions:

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.

It is difficult to determine an adequate criterion for deciding which concepts are, in fact, significant to the relevance of a particular document. A discrimination factor, d_i, can be calculated from the quantities r_i and n_i, where d_i, r_i, and n_i, are defined by equations (1), (2), and (3).

$$r_i = \frac{1}{I} \sum_{k \varepsilon R} c_{k,i} \quad ; \quad I = \text{no. of elements } \varepsilon \ R \qquad (1)$$

$$n_i = \frac{1}{J} \sum_{k \varepsilon N} c_{k,i} \quad ; \quad J = \text{no. of elements } \varepsilon \text{ N} \qquad (2)$$

$$d_i = (r_i - n_i)/(r_i + n_i) \qquad (3)$$

$c_{k,i}$ is the weight of concept i in the kth document.

R is the set of relevant retrieved documents.

N is the set of nonrelevant retrieved documents.

Thus r_i is the average weight of concept i in the retrieved relevant documents; n_i is the average weight of concept i in the retrieved non-relevant documents. The difference, $r_i - n_i$, if positive, is then a measure of how much more important the ith concept is in describing the nature of the relevant documents than in describing the nature of the nonrelevant documents. This measure, when normalized by dividing by the factor, $r_i + n_i$, becomes the desired discrimination factor, d_i. A positive value for d_i indicates that the concept occurs more frequently in the retrieved relevant documents than in the retrieved nonrelevant and therefore is of some significance. Clearly, the larger the value of d_i, the more significant the concept is as an indicator of document relevance. A concept is deemed "significant" if and only if

$$d_i > \delta \qquad (4)$$

where δ is an appropriately chosen constant which specifies the minimum value of d_i, which demonstrates the "importance" of the concept.

A reasonable approach to determining the proper magnitudes of the ensuing alterations is to define the increment as a function of d_i. All documents within the entire document space are then modified by the formula:

$$c_{k,i} = c_{k,i}(1+\gamma d_i) \text{ for appropriately chosen } \gamma. \qquad (5)$$

It is evident that if $c_{k,i}$ is originally 0, equation (5) will not affect its value. However, consistent with assumption b) above, for documents deemed relevant, the absence of concept i will result in an alteration specified by equation (6) as follows:

$$c_{k,i} = \varepsilon \text{ for } K \varepsilon R \qquad (6)$$

Since concept weights can never decrease with this scheme, they would grow unmanageably large over a long period of time if no provisions were made to check this growth. Consequently, the documents are all normalized to a Euclidean length of 1000. This normalization process serves an additional purpose. Concepts which are never significant, i.e., have corresponding d_i's which are always negative or negligible positive quantities, are reduced in magnitude due to the increase in the weights of the relevant concepts. In a sense, negative feedback is thereby achieved.

The retrieval process can now be specified by the following algorithm:

a) Retrieve the top 15 documents (based upon the cosine correlation with the query).

b) Obtain relevance feedback judgments from the user concerning these 15 retrieved documents (Our experiments relied on a priori knowledge of the relevant documents to simulate this feedback procedure.)

c) Compute r_i, n_i, and d_i, from (1), (2), and (3).

d) Set $d_i = 0$ if $d_i < 0$ (7)

e) Process the collection and perform the transformation specified by (5).

f) Repeat step a) with the modified collection and the same or different query depending on input specification.

3. Experimental Results

The basis experiment consists in applying algorithm (7), programmed on the IBM 360/65, to the Cranfield collection of 200 documents and 42 queries.

In performing the experiments summarized below, two general conditions may obtain during the alteration of the document collection:

a) The queries in each group have similar sets of relevant documents, (illustrated in Fig. 4);

b) The queries in each group have different sets of relevant documents, (illustrated in Figs. 2 and 3).

In almost all instances, several queries are "batch processed," before the collection is refined. Condition b) is probably more representative of a real situation since many queries would normally enter a retrieval system before the collection is updated.

The recall and precision indicated in Figs. 2 through 5 are typical of the results obtainable with the proposed system. The collection was modified on the basis of prior searches using query 34 for Fig. 2, queries 12, 15, 16, 17, 38, and 41 for Fig. 3, and queries 7, 15, and 17

Rank	Document	Correlation
1	104	.2307
2	102 R	.2091
3	199	.1501
4	96	.1484
5	18	.1451
6	191	.1409
7	200	.1407
8	99	.1394
9	193 R	.1304
10	109	.1223
11	83 R	.1169
12	98	.1146
13	91	.1114
14	90	.1102
15	64	.0960

a) SMART Standard Retrieval

Rank	Document	Correlation
1	102 R	.2273
2	104	.2262
3	199	.1447
4	18	.1420
5	191	.1409
6	193 R	.1398
7	96	.1363
8	83 R	.1349
9	200	.1328
10	99	.1264
11	109	.1192
12	64	.1124
13	90	.1095
14	91	.1091
15	98	.1064

b) Iteration Using Query 34
$\gamma = 0.1$, $\delta = 0.9$, $\varepsilon = 0$

Rank	Document	Correlation
1	102 R	.2290
2	104	.2276
3	193 R	.1543
4	199	.1444
5	18	.1440
6	191	.1418
7	96	.1359
8	83 R	.1339
9	200	.1313
10	99	.1282
11	109	.1182
12	64	.1165
13	91	.1091
14	90	.1083
15	98	.1057

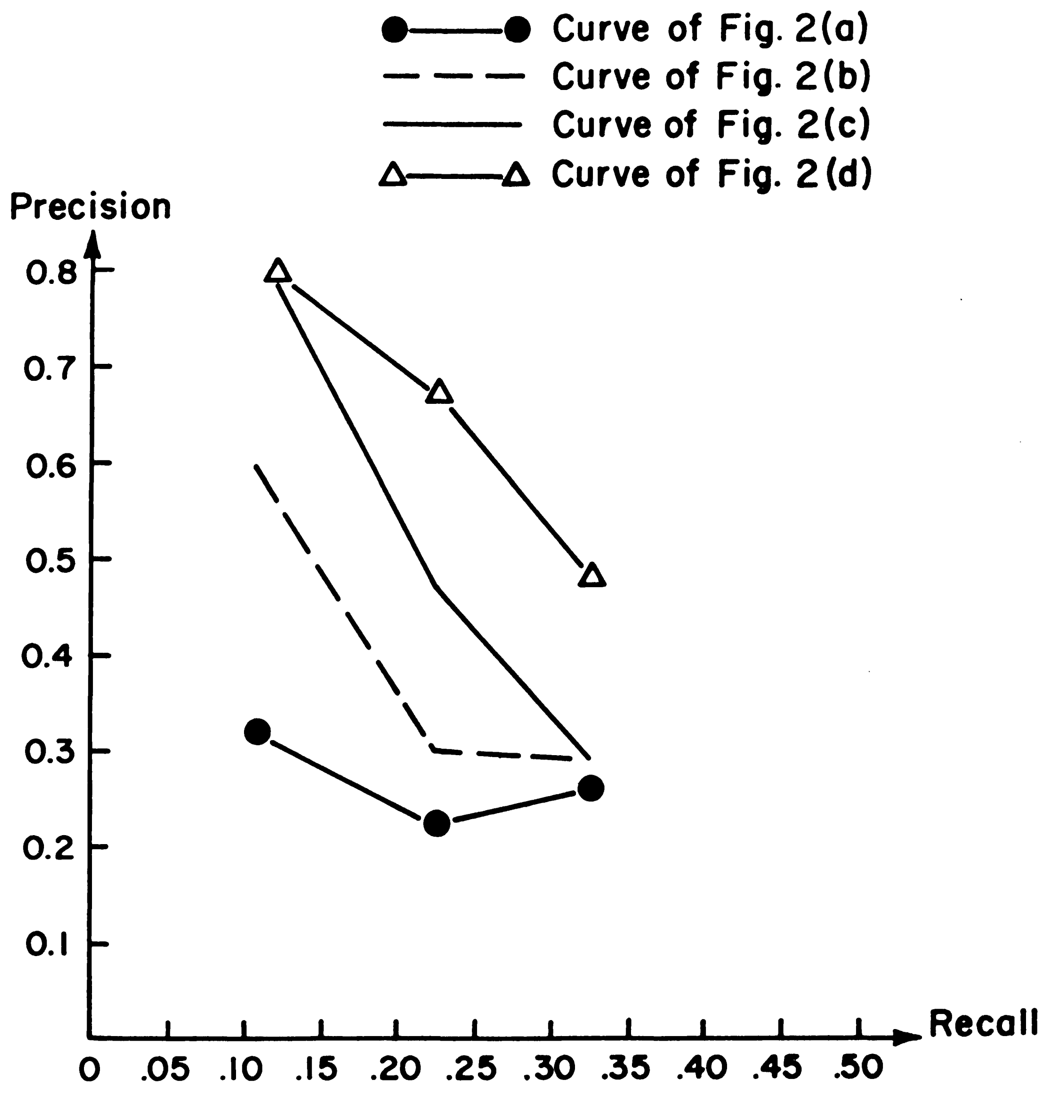
c) Iteration Using Queries 34 and 16
$\gamma = 0.1$, $\delta = 1.0$, $\varepsilon = 50$

Rank	Document	Correlation
1	102 R	.2380
2	104	.2295
3	193 R	.1455
4	83 R	.1438
5	191	.1427
6	199	.1420
7	18	.1387
8	96	.1352
9	200	.1287
10	64	.1237
11	99	.1215
12	109	.1173
13	90	.1090
14	91	.1088
15	17	.1076

d) Iteration Using Query 34
$\gamma = 0.2$, $\delta = 0.5$, $\varepsilon = 50$

Retrieval Results for Query 16

Fig. 2



e) Recall-Precision Curve

Fig. 2
(contd.)

Rank	Document	Correlation
1	41 R	.4762
2	100	.4280
3	90 R	.3859
4	111	.3151
5	11	.3123
6	45	.2896
7	110	.2750
8	127	.2688
9	104	.2637
10	192	.2610
11	71	.2601
12	159	.2576
13	42 R	.2572
14	76	.2481
15	133	.2480

a) SMART Standard Retrieval

Rank	Document	Correlation
1	41 R	.4784
2	100	.4312
3	90 R	.3799
4	111	.3177
5	11	.3095
6	45	.2910
7	127	.2737
8	110	.2699
9	104	.2662
10	42 R	.2620
11	159	.2582
12	76	.2560
13	71	.2506
14	133	.2503
15	185	.2461

b) Space Modification
$\gamma = 0.1$, $\delta = 0.5$, $\varepsilon = 50$

Rank	Document	Correlation
1	41 R	.4685
2	100	.4317
3	90 R	.3615
4	111	.3192
5	11	.3014
6	45	.2901
7	127	.2761
8	42 R	.2685
9	104	.2672
10	110	.2646
11	76	.2618
12	159	.2580
13	133	.2516
14	185	.2484
15	39	.2453

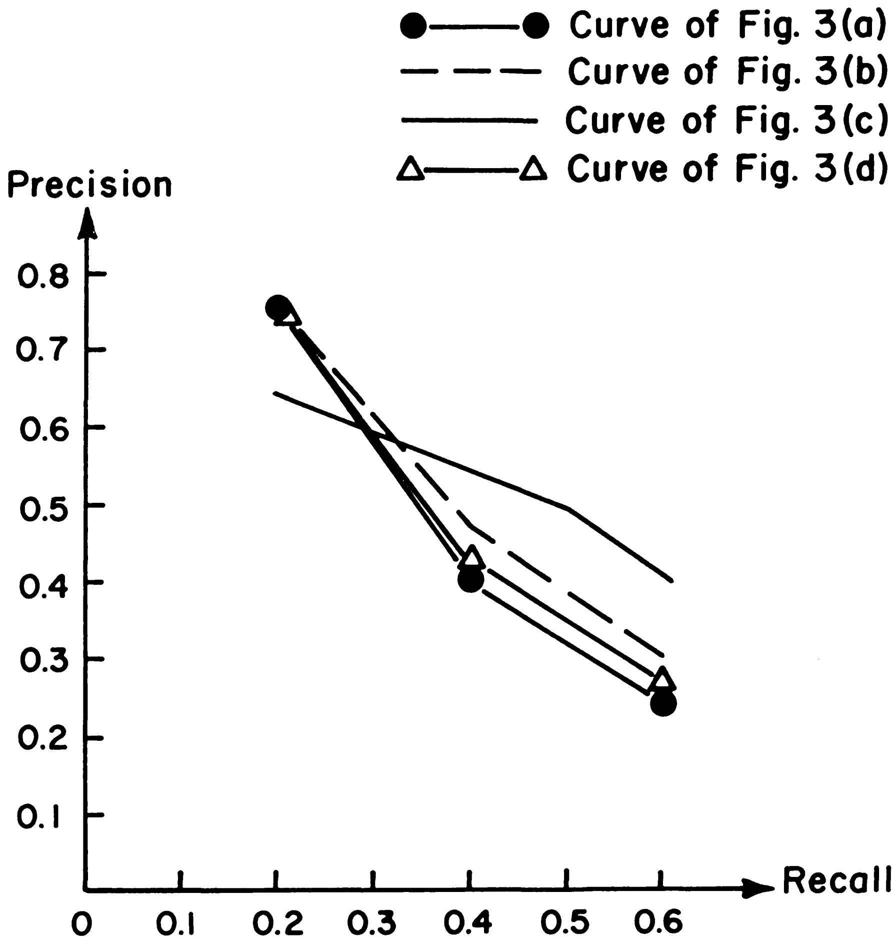
c) Space Modification
$\gamma = 0.2$, $\delta = 0.5$, $\varepsilon = 50$

Rank	Document	Correlation
1	41 R	.4450
2	100	.4183
3	111	.3152
4	90 R	.3085
5	42 R	.2895
6	45	.2731
7	127	.2708
8	76	.2678
9	11	.2643
10	39	.2641
11	104	.2610
12	188	.2492
13	156	.2492
14	185	.2489
15	159	.2478

d) Space Modification
$\gamma = 0.5$, $\delta = 0.2$, $\varepsilon = 50$

Retrieval Results for Query 7
(Results after iteration on Queries 12, 15, 16, 17, 38 and 41)

Fig. 3



e) Recall-Precision Curve

Fig. 3
(contd.)

Rank	Document	Correlation
1	80 R	.5190
2	81 R	.5021
3	102 R	.4696
4	66	.4537
5	82 R	.4218
6	69	.4036
7	83 R	.3966
8	88 R	.3831
9	125	.3807
10	193 R	.3660
11	114	.3578
12	94	.3533
13	111	.3369
14	124	.3341
15	11	.3213

a) SMART Standard Retrieval

Rank	Document	Correlation
1	80 R	.5009
2	81 R	.4889
3	66	.4661
4	102 R	.4460
5	69	.4192
6	82 R	.4073
7	125	.3935
8	83 R	.3828
9	114	.3703
10	94	.3611
11	88 R	.3589
12	193 R	.3443
13	111	.3404
14	124	.3361
15	11	.3296

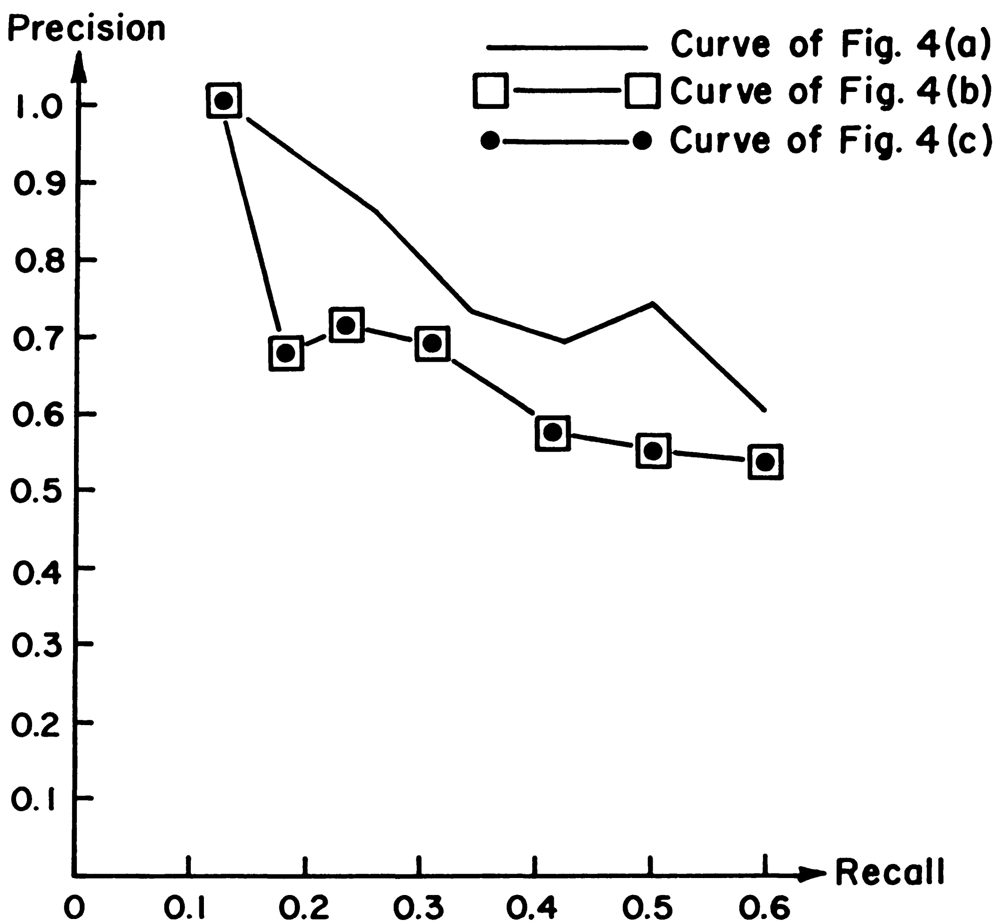
b) Space Modification Using Queries 7, 15 and 17
$\gamma = 0.1$, $\delta = 0.7$, $\varepsilon = 50$

Rank	Document	Correlation
1	80 R	.5091
2	81 R	.4964
3	66	.4661
4	102 R	.4505
5	69	.4193
6	82 R	.4104
7	125	.3935
8	83 R	.3883
9	114	.3704
10	94	.3611
11	88 R	.3606
12	193 R	.3503
13	111	.3404
14	124	.3362
15	11	.3296

c) Space Modification Using Queries 7, 15, and 17
$\gamma = 0.1$, $\delta = 0.9$, $\varepsilon = 0$

Retrieval Results for Query 15

Fig. 4



d) Recall-Precision Graph

Fig. 4
(contd.)

Initial Search			New Search Following Space Modification		Change in Recall and Precision
Figure	Queries	Relevant Documents	Queries	Relevant Documents	
2	34	10 127 128 129 130	16	67 80 81 82 83 84 85 102 193	Increase
3	12 15 16 17 38 41	46 47 48 49 50 51 52 53 54 55 67 80 81 82 83 84 85 86 87 88 90 91 92 93 94 95 102 109 120 178 192 193 194 195	7	41 42 72 90 95	Increase
4	7 15 17	41 42 72 80 81 82 83 84 85 86 87 88 90 91 93 94 95 102 109 193	15	80 81 82 83 84 85 86 87 88 102 109 193	Decrease

Summary of Results of Space Modification

Fig. 5

for Fig. 4. Following these space modifications, searches were performed using queries 16, 7, and 15 respectively.

One result is immediately obvious. Whenever the collection is modified using non-related queries (queries with different sets of relevant documents from those of the query now at hand), recall and precision increase. However, whenever the collection is first modified using queries with relevant document sets related to those of the query being examined, recall and precision decrease. The latter result may be explained by the fact that although the clustering of relevant documents is accomplished as desired, the centroid of the generated cluster moves away from the query (as in the example of Fig. 4).

Specifically, when a space modification is performed, there exists no a priori reason for expecting that the new document space should be such that the relevant documents are clustered around the present query, thus yielding improved precision and recall, since the scheme which controls the document space alteration is independent of the query. However, the relevant document cluster must be located at points in the space which are close to the query being processed if increased precision and recall are to be achieved. The addition of query modification to the system is therefore necessary to assure this closeness; the query will be moved towards the relevant document cluster, thus facilitating improved retrieval results.

In order to verify the fact that the relevant documents are grouped together by the space modification process, the document-document correlations of the original space and of the modified space are computed as in the example of Figs. 6 and 7. These results confirm the fact that the related documents are indeed grouped more closely together than they were originally. For example, in Fig. 6, Query 16 retrieves only relevant documents 83, 102, and 193. However, the intercorrelations among documents 80, 81, 83, 84, and 88 all increased markedly. While it is true that documents 67, 85, and 102 are essentially unaffected by the modification process, 5 relevant non-retrieved documents were clustered as desired. Fig. 7 offers another example of similarly successful clustering.

An additional experiment was conducted to demonstrate that the proposed system, when expanded to include query modification in addition to document space alteration, leads to the desired increase in precision and recall. Specifically, given the modified document space, relevance feedback results are used to modify the query in a fashion similar to that used by [2].

An updated query, Q', is determined from an original query, Q, using the following equation:

$$Q' = Q + \frac{\Sigma R_i}{|\Sigma R_i|} \tag{8}$$

where the R_i are the relevant documents.

The denominator is used to normalize the changes in the modification procedure so that the query is not altered too radically. If this normalization were not carried through, the incremented concepts would nullify the effects of any of the components not affected by the modification procedure.

Fig. 8 demonstrates the above assertion. Fig. 8(a) represents the original SMART retrieval with the original query; Fig. 8(b) represents retrieval results using the original query and the modified collection. The

Documents	67	80	81	82	83	84	85	86	87	88	102	109	193
67	1000	254 259	234 203	262 227	119 133	027 032	575 580	262 265	261 202	106 120	237 232	228 215	273 267
80		1000	703 700	419* 475	529 544	440* 495	249 254	360 377	296* 328	298* 373	492 475	234* 260	490 478
81			1000	502 515	541* 578	486* 532	253 240	442 457	358* 385	348* 406	613 580	311 316	476 472
82				1000	617 623	524* 548	132 127	206 217	284 286	366* 395	366 382	310 297	344 337
83					1000	779 786	132 129	333* 370	464* 552	420* 543	465* 541	220 223	418* 465
84						1000	063 053	272* 303	494* 592	531* 636	308 322	162 164	275 276
85							1000	293 291	219 163	176 172	269 257	136 140	252 256
86								1000	351 340	163* 197	429 440	305 309	407 410
87									1000	517* 646	333 296	136 101	245 237
88										1000	300 271	136 133	216 222
102											1000	341 339	470* 550
109												1000	343 324
193													1000

Document-Document Correlations for Documents Relevant to Query 16

(feedback using query 16 with $\gamma = 0.1$, $\delta = 1$; top number = original space; bottom number = modified space; *Significant increase in correlation)

Fig. 6

Document	8	13	58	59	60	200
8	1000	191* 212	186 177	233 219	000 014	183 163
13		1000	335 332	318 326	279 274	132 130
58			1000	456* 522	504* 526	384 389
59				1000	235* 304	446 450
60					1000	086 108
200						1000

Document-Document Correlations for Documents
Relevant to Queries 4 and 5

(Space modification $\gamma = 0.1$, $\delta = 0.7$;
* Significant increase)

Fig. 7

Rank	Document	Correlation
1	165	.6559
2	162	.5773
3	164	.4740
4	58	.4772
5	163	.3741
6	60	.3721
7	110	.3615
8	150	.3570
9	92	.3497
10	167	.3381
11	127 R	.3181
12	198	.3162
13	10 R	.3058
14	185	.2971
15	128 R	.2888

a) Standard SMART Retrieval
R = 0.9364 P = 0.5960

Rank	Document	Correlation
1	165	.6485
2	162	.5713
3	164	.4675
4	58	.4337
5	60	.3670
6	163	.3642
7	110	.3581
8	150	.3475
9	92	.3451
10	167	.3353
11	198	.3122
12	185	.3002
13	127 R	.2921
14	10 R	.2757
15	129 R	.2682

b) Retrieval Using Modified Space and Fixed Query
(γ = 0.1, δ = 1.0)
R = 0.9887 P = 0.9044

Rank	Document	Correlation
1	165	.6434
2	10 R	.6377
3	129 R	.5872
4	162	.5609
5	127 R	.5502
6	164	.5422
7	58	.5013
8	128 R	.4581
9	163	.4386
10	130 R	.3840
11	167	.3819
12	14	.3792
13	150	.3788
14	92	.3776
15	110	.3676

c) Retrieval Using Relevance Feedback (modified query, fixed space)
R = 0.9867 P = 0.8621

Rank	Document	Correlation
1	10 R	.6521
2	165	.6331
3	129 R	.6114
4	127 R	.5801
5	162	.5688
6	164	.5461
7	58	.5067
8	128 R	.4592
9	163	.4345
10	130 R	.4014
11	92	.3918
12	150	.3911
13	14	.3786
14	110	.3721
15	60	.3587

d) Retrieval Using Modified Space and Modified Query
(γ = 0.1, δ = 1.0)
R = 0.9887 P = 0.9044

Retrieval Results for Various Combinations of Space and Query Modification

Fig. 8

results given in Fig. 8(b) are not as good as those in Fig. 8(a) since the document space has been reclustered, but not around the query vector. Fig. 8(c) represents the SMART retrieval with the modified query, whereas Fig. 8(d) represents retrieval results using the modified query and the modified collection. As expected, the results indicated in Fig. 8(d) are better than those in Fig. 8(c), the standard method of relevance feedback (using only query modification).

In analyzing the effects of the various chosen values for δ, γ, and ε, it appears that best results are obtained for small values of γ and large values of δ. For $\delta = 0.5$, too many significant d_i's are generated, not all of which were actually important to the relevance of the pertinent documents. Since as many as 18 to 20 concepts were present with corresponding d_i's of 1.0, large values of δ such as 0.8, 0.9, 0.98, yielded the best retrieval results. With $\gamma = 0.1$, retrieval results on successive iterations were almost identical to those of SMART. However, since the collections were being reclustered, impressive results could be obtained with query modification. The documents were greatly altered when large values for γ were chosen; the corrections to the concept weights adjusted these weights too drastically to be of value.

Concerning the modification to ε, the initial weight of new concepts entered into a document concept vector, it appears as if the addition of $\varepsilon \neq 0$ to relevant documents has some effect; however, the effect is not appreciable, unless, of course, ε is set to some unreasonably large value.

4. Results and Conclusions

The results of these experiments indicate that by use of the discrimination factor, d_i, to guide the redefinition of the document space, documents are reclustered, and are subsequently brought closer together. However, the reclustering does not necessarily take place around the original query. This explains the result that initial space modification without query modification is not necessarily as useful as an original SMART search would be. As soon as relevance feedback is further employed to modify also the query, the relocation of the query leads, however, to an improved retrieval when the next search is accomplished. The next step to be taken is clearly the general incorporation of document space and query modification into a system such as SMART.

An aspect of the study which has not been fully investigated to date is the practicality of a system which centers around the frequent updating of a large document collection. Ideally, of course, the collection should be updated after each query is processed. However, this is certainly a very tedious process. In the experimental study of the 200 documents in the Cranfield collection, a full search on the 360/65 took about 15 seconds while updating the collection took about 10 seconds. By batching the queries in groups of three or four, these processing times are reduced to 6 and 9 seconds for the searching and collection refinement procedures, respectively. It may be possible to modify the collection while a full search is going on, thus reducing the processing times still further. That is, after computing the cosine correlations, the document is modified before writing it out again, thus eliminating one complete reading of the document collection per iteration.

The appendix describes the programming system written to carry out this study.

References

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

[2] E. Ide, User Interaction with an Automated Information Retrieval System, Scientific Report No. ISR-12 to the National Science Foundation, Section VIII, Department of Computer Science, Cornell University, June 1967.

[3] O. W. Riddle, T. Horwitz, and R. Dietz, Relevance Feedback in Information Retrieval Systems, Scientific Report No. ISR-11 to the National Science Foundation, Section VI, Department of Computer Science, Cornell University, June 1966.

[4] G. Salton, Class Notes in Information Storage and Retrieval.

Appendix

The Programming System

The programming system used consists of the five subroutines MAIN, SEARCH, GET, SET and UPDATE, which serve the following functions:

A) MAIN

1. Reads in the collections (documents and queries) and stores them on a disk.
2. Calls SET and then SEARCH.

   SET - Changes the PSW in the 360 so that exponent underflow messages do not appear in the output listings.

B) SEARCH

1. Reads in a card containing the next set of queries to "batch".
2. Reads in the actual query vectors into core from the disk storage.
3. Reads the document collection, and computes the cosine correlation for each document with all queries. The results of this search are sorted and printed.
4. Calls GET to read into core the relevant retrieved documents.
5. Calls UPDATE to compute the d_i's and update the collection.
6. Reads in the next batch of queries, if any, and repeats the search procedure. If there are no more queries, then control passes back to MAIN, which terminates execution.

   GET - reads into core the needed documents or queries based upon the a priori relevance feedback.

C) UPDATE

1. Computes and prints the discrimination factor d_i for each concept.
2. Updates the document collection.

The collection to be read in has been modified from the original SMART collection in order to be compatible with Cornell University's COOL system for the 360. Columns 1 to 72, only, are used on each card. The format of the collection is as follows:

Card 1 - columns	1-16	Title of collection
	17-20	NCOL - Number of cards in collection
	21-24	NTDOC - Number of documents in collection

Card 2 - columns	1-8	'NO MORE'

Next follows NTDOC blocks of cards, one for each document. The format for each block is:

Card 1 - columns	1-4	Document number
	9-12	Number of concept weight pairs
	13-16	Number of a priori relevant documents

Card 2 - Concept weight pairs, 9 per card, 4 columns for concept number, 4 columns for weight. Document vectors are normalized to an Euclidean length of 1000.

Last card - A priori relevance information, 4 columns each. (For the document collection, this card is blank).

The very last card in the set, immediately after the NTDOCth block, contains 'END' in columns 1 to 4. This is used as an error check whenever the collections are read.

The collections are organized so that 'A' format is used whenever the collections are read.

In order to run the program, the following data cards are used:

Card 1 - Columns	1-10	Maximum concept number in collection (I10 format)
	31-40	Value of GAMMA (F10.5)
	41-50	Value of DELTA (F10.5)
	51-60	Value of Epsilon (F10.5)
	61-70	Number of retrieved documents to sort and print (I10)
Card 2 - Column 1	0	Do not punch out updated collection at end of retrieval.
	1	Punch out collection (onto a Cornell Data Set)
Columns	2-72	Title card

Beginning with card 3 - Document collection, followed by query collection.

The remaining data cards are the actual search cards, one card per iteration. Each card has the format:

Columns	1-4	Number of queries to batch process
	5-8, 9-12, 13-16, ...	Query numbers, in ascending order.

At most 17 queries can be batched at once. (This arbitrary number is due to the size of the arrays set up in SEARCH. In order to batch more than 17 queries, the size of ARRAY and INDEXR must be increased).

The printed output consists on the top retrieved documents (as determined by the number in columns 61 to 70 from the first data card), including the a priori knowledge of the relevant documents. This is followed by a listing of all of the query-document correlations. After listing the results for all of the queries which were batched at one time, the d_i's are printed as the collection is updated.

In addition to the basic retrieval programs, three independent utility programs exist. The first lists the document collection. This routine is needed whenever a modified collection is punched onto cards and it is desired to see how the concept weights have been changed.

A second program computes the document-document correlations, given a document collection as input.

The third program modifies the query and computes the correlation of the modified query with the documents in the space given the original query and the a priori known relevant documents as input. This routine effectively performs relevance feedback.

At the present time, these three routines are independent of the main retrieval system; however, it would be relatively easy to incorporate them as subroutines of SEARCH in order to generate an effective retrieval system, which would also include the standard relevance feedback process.