The Cluster Hypothesis Revisited

Ellen M. Voorhees*

TR 85-658

January 1985

(Revised April 1985)

Department of Computer Science
Cornell University
Ithaca, New York 14853

* This study was supported in part by the National Science Foundation under grant IST 83-16166.

The Cluster Hypothesis Revisited

Ellen M. Voorhees[1]
Department of Computer Science
Cornell University
Ithaca, NY 14853

April 12, 1985

[1]This study was supported in part by the National Science Foundation under grant IST 83-16166.

Abstract

A new means of evaluating the cluster hypothesis is introduced and the results of such an evaluation are presented for four collections. The results of retrieval experiments comparing a sequential search, a cluster-based search, and a search of the clustered collection in which individual documents are scored against the query are also presented. These results indicate that while the absolute performance of a search on a particular collection is dependent on the pairwise similarity of the relevant documents, the relative effectiveness of clustered retrieval versus sequential retrieval is independent of this factor. However, retrieval of entire clusters in response to a query usually results in a poorer performance than retrieval of individual documents from clusters.

1 Introduction

Document clustering has been used in experimental information retrieval systems for many years [1,2,3,4]. The original goal of document clustering was to improve the efficiency of a search by reducing the number of documents that needed to be compared to the query. However, Jardine and van Rijsbergen reasoned that it should be possible to exploit the information inherent in a clustered collection, and thus that document clustering should be able to improve the effectiveness as well as the efficiency of retrieval searches [5]. Specifically, they stated the ***cluster hypothesis***: "the associations between documents convey information about the relevance of documents to requests", and proposed *cluster-based* retrieval on hierarchically clustered collections as the means by which the document relationships could be used [5,6].

Cluster-based retrieval retrieves one or more clusters in their entirety in response to a query. This is in contrast to most other cluster search methods which identify clusters that are likely to contain good documents and then compute the similarity between the query and each of the documents in the identified clusters. The rationale for using cluster-based retrieval is as follows: if documents that are similar to one another are relevant to the same queries, *i.e.* if the cluster hypothesis is true for a given collection, then clusters should contain mostly documents that are relevant to the same queries. Retrieving entire clusters should therefore be an effective strategy, provided that the proper clusters are chosen to be retrieved.

However, in her work in relevance feedback, Ide concluded that more than one feedback query should be created for each original query since relevant documents are frequently more similar to non-relevant documents than they are to some other relevant documents [7]. Since this intermingling of the relevant and non-relevant documents contradicts the cluster hypothesis, cluster-based retrieval would not be expected to work very well for a collection in which the intermingling occurred to a significant extent.

This paper reports on an investigation into how well the cluster hypotheses characterizes four test collections. Retrieval results for three different retrieval strategies are included. The next section describes the retrieval environment, including the characteristics of the collections. A new test for determining whether the cluster hypothesis holds for a given document collection is introduced and the results of this test are presented. We then investigate the cluster hypothesis in more detail by actually searching clustered and nonclustered document collections. Three retrieval strategies — a sequential search, a cluster-based search, and a search of the clustered collection in which individual documents are compared to the query — are described in the final section and the results of the retrieval experiments are summarized.

2 Retrieval Environment

The environment used in this study is similar to a vector processing environment. Each document is indexed automatically by a process that removes words found on a stop list, maps word variants into the same term, and assigns a weight to each term; that is, each document is represented by a set of weighted terms. The assigned weight is proportional to the number of times the term is used in the document and inversely proportional to the number of documents in which the term appears. The exact formulation of the weight can be found in [8].

Weighted extended Boolean queries as described in [9] are used. The similarity value computed between a query and a document is a function of the document weights, the query weights, and a parameter known as the ***p-value*** that controls the interpretation of the Boolean connectives. The p-value can take on real values between 1.0 and ∞. When the p-value is 1.0, the extended Boolean system reduces to a vector processing system; when the p-value is ∞ and the query weights are binary, the extended Boolean system reduces to the fuzzy set model.

For the queries used in this study, the weight of a term in a query is proportional to the inverse document frequency of the term in the document collection, and the weight of a clause is the mean of the weights of the components of the clause. As suggested in [9], all p-values are equal to 2.0.

2.1 Collection Statistics

The four test collections include

- MED, a biomedicine collection of 1033 documents and 30 queries,
- CACM, a computer science collection of 3204 documents and 52 queries,
- CISI, a documentation collection of 1460 documents and 35 queries, and
- INSPEC, an electrical engineering collection of 12684 documents and 77 queries.

Various statistics about these collections are given in Table 1.

2.2 Testing the Cluster Hypothesis

For experimental purposes, it is clearly desirable to know the extent to which the cluster hypothesis is true for a given test collection. To this end, Jardine and van Rijsbergen introduced the ***cluster hypothesis test*** [5,6]. The test involves plotting two frequency distributions and observing the separation between them. The distributions to be plotted are the frequency distribution of the distances between all pairs of documents such that both of the documents are relevant to the same query, and the frequency distribution of the distances between all pairs of documents such that one document is relevant to some query and the other document is not relevant to that query. A separation between the two distributions implies the cluster hypothesis may be true for the collection.

A test equivalent to this one was performed on each of the four collections described above. It differs only in that the frequency of the cosine similarities between documents (as opposed to the distances between them) was computed. These plots can be found in Figure 1. The separation between the frequency distributions in the MED, CACM, and INSPEC collections is substantial, while the separation in the CISI collection is quite small.

This test has been useful in explaining the widely varying effect changes to a retrieval system can have on different collections [10]. However, its appropriateness for testing the cluster hypothesis

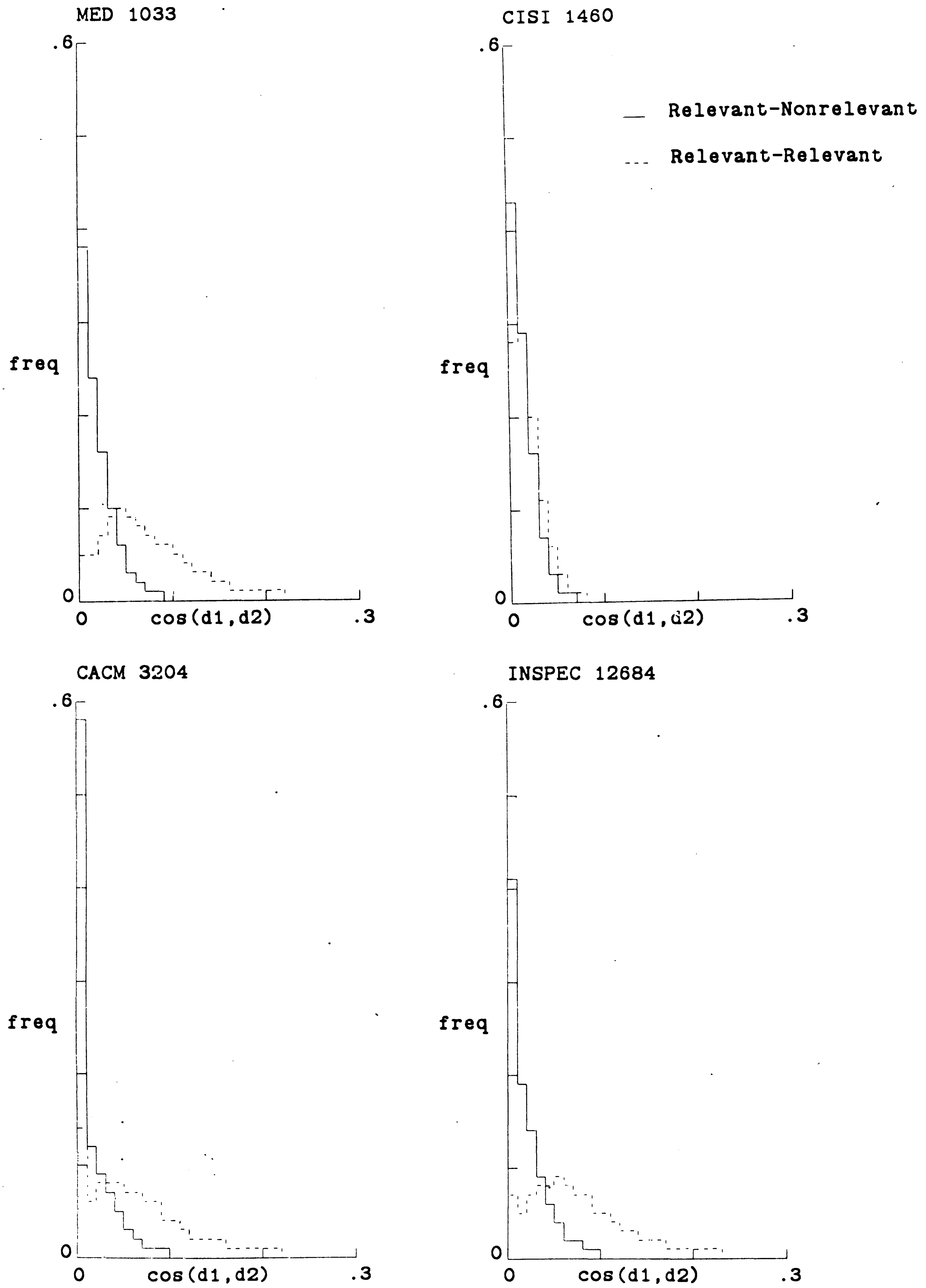

Figure 1: Cluster hypothesis test results

	MED	CACM	CISI	INSPEC
Number documents	1033	3204	1460	12684
Number terms	6927	8503	4941	14573
Mean terms per doc	51.6	22.5	43.9	32.5
Number queries	30	52	35	77
Mean terms per query	39.7	4.3	7.2	13.2
Mean relevant docs per query	23.2	15.3	49.8	33.0

Table 1: collection characteristics

is open to question. Since there are always many more relevant–non-relevant than relevant–relevant pairs, the relative frequency of very similar relevant–non-relevant pairs will be much less than the relative frequency of very similar relevant–relevant pairs even if the absolute number of pairs is the same. However, whether or not the cluster hypothesis is true for a collection will depend on the absolute number of non-relevant documents that are very similar to the relevant documents. Since the cluster hypothesis test does not give information at this level of detail, another test, the *nearest neighbor* test, was performed on the document collections.

The n nearest neighbors of a document d are the n documents that are the most similar to d. If the cluster hypothesis characterizes a collection, many of the nearest neighbors of a relevant document will also be relevant. The nearest neighbor test checks if this condition holds by computing the n nearest neighbors of a relevant document and recording the number of these documents that are also relevant. This process is repeated for each relevant document of each query that has more than one relevant document.

The results of the nearest neighbor test for each of the four collections can be found in Table 2. For each of the collections the value of n was (arbitrarily) set to five. The Table gives the percentage of relevant documents that have $0, \ldots, 5$ relevant nearest neighbors.

The two tests give rather different pictures of the document collections. The INSPEC collection has a reasonably good separation of the distributions in the cluster hypothesis test. In contrast, the nearest neighbor statistics show that for nearly half the relevant documents, there are no other relevant documents among the five nearest neighbors, and that for 70% of the relevant documents there is at most one relevant document among the five nearest neighbors. (The separation of the frequency distributions of the INSPEC collection probably results from the 9% of the relevant documents that have four or five relevant documents as nearest neighbors. Apparently most of the queries in this collection are either quite general, with the relevant documents spread throughout the collection, or are quite specific and thus have a more concentrated set of relevant documents.) The CACM collection, which also has a good separation in the cluster hypothesis test, has 57% of

# rels in nn set	MED	CACM	CISI	INSPEC
0	8	28	38	46
1	11	29	30	24
2	17	20	20	14
3	23	15	8	8
4	24	5	3	5
5	17	3	1	4

Table 2: Percentage of relevant documents with given number of relevant nearest neighbors

the relevant documents with at most one other relevant document among the five nearest neighbors. The distributions of the MED collection are similar to the distributions of the CACM and INSPEC collections, but MED has only 19% of the relevant documents with at most one other relevant document among the five nearest neighbors. The CISI collection has a small separation of the frequency distributions and correspondingly poor percentages of relevant documents with relevant nearest neighbors.

The percentage of relevant documents that have other relevant documents as nearest neighbors gives a more accurate description of how well the cluster hypothesis characterizes a collection than the separation of the frequency distributions of the distances between relevant–relevant and non-relevant–relevant documents does. The nearest neighbor test results suggest that the cluster hypothesis holds for the MED collection and does not hold for the CISI collection. It characterizes the other two collections to a limited extent.

3 Retrieval Experiments

Previous experiments have tested the effectiveness of cluster-based retrieval on hierarchically clustered collections [5,11,12,13]. These experiments provided evidence that cluster-based retrieval was about as effective as retrieval based on a total scan of the document collection, especially for precision-oriented searches. In particular, a bottom-up search of the hierarchy proved to be quite effective [13].

The single-link clustering method was used in the previous work and is also used in this study. The hierarchy constructed by this method is equivalent to a maximum spanning tree of the document collection where the weights of the tree edges are document similarities. The hierarchy itself is represented by a tree with the documents as the leaves. Associated with each interior node is a similarity value. Each subtree of the hierarchy corresponds to a single-link cluster of the collection at a threshold equal to the similarity value associated with the subtree's root. The clusters at the bottom of the hierarchy are small and consist of highly similar documents; clusters at higher levels of the hierarchy are much larger.

Each cluster is represented by a *centroid vector*. The centroid vectors are computed as follows:

- The sum of the within-document frequency of each of the terms in the cluster is computed. These frequencies are then ranked from largest to smallest.

- The top 100 terms are selected to be in the centroid vector. The weight of each of the terms in the centroid is the rank of the term in the sorted list – equal frequencies are assigned the "small enough" is defined as $size + NumRetrieved < NumWanted + 5$. If the cluster is not small enough, only the children of the node (as opposed to the entire cluster below the node) are considered.

```
perform inverted-index search of low-level centroids;
return top 10 centroids;
NumRetrieved ← 0;
while (NumRetrieved < NumWanted and there is another centroid) {
      if (the next cluster is "small enough") {
            Retrieve all documents in cluster;
            NumRetrieved ← NumRetrieved + sizeof(cluster);
      }
}
```

Figure 2: ENTIRE – a bottom-up search that retrieves entire clusters

same rank. (These are Murray's rank weight centroids [3].)

- Because the similarity function requires document weights between 0 and 1, the weights assigned in the preceding step are modified by the same procedure that computes document vector weights.

3.1 The Searches

Experiments using two bottom-up cluster searches are performed and the results compared with each other and with a sequential search. The somewhat simplified search algorithms are given in Figures 2 and 3.

Each of the cluster searches enforces a minimum and a maximum number of documents to be retrieved. These bounds are based on the number of documents that the user desires to retrieve (*NumWanted* in Figures 2 and 3). A larger value of *NumWanted* will generally produce better recall.

Both cluster searches begin by using an inverted index of the *low-level* centroids to find the best ten low-level clusters. (The low-level cluster of document *d* is the smallest cluster that contains document *d*. The set of low-level clusters of the collection consists of the low-level cluster of each document in the collection.) Using the ranked list of centroids returned by the inverted index search, the ENTIRE search retrieves entire clusters until at least *NumWanted* documents, but no more than $NumWanted + 5$, documents are retrieved. This search never examines individual documents. The other cluster search (INDIV) retrieves individual documents from the clusters identified by the inverted index search. The documents retrieved are those *NumWanted* documents in the highest ranking clusters that are the most similar to the query. INDIV and the sequential search (SEQ) will retrieve exactly *NumWanted* documents, but they will examine more than that number of documents in the searching process.

3.2 Evaluation

Evaluation of clustered retrieval must be done using document-level measures [1] since the cluster searches do not create a total ranking of the documents. Three evaluation measures are used here: recall, precision, and van Rijsbergen's E measure with a β value of 1. Given a set of retrieved

"small enough" is defined as $size < NumWanted + 5$. If the cluster is not small enough, only the children of the node (as opposed to the entire cluster below the node) are considered.
"results" is an array of length *NumWanted*.

```
perform inverted-index search of low-level centroids;
return top 10 centroids;
while (there is another centroid) {
    if (the next cluster is "small enough") {
        for (each document in cluster) {
            similarity ← sim(doc, query);
            if (similarity > MinSim) {
                replace document in results with similarity MinSim by doc;
                compute new MinSim;
            }
        }
    }
}
```

Figure 3: INDIV – a bottom-up search that retrieves individual documents

documents, recall is defined as the proportion of the relevant documents that are retrieved, and precision is defined as the proportion of the retrieved documents that are relevant. The E measure with parameter β is defined as

$$1 - \frac{1}{\frac{\alpha}{R} + \frac{1-\alpha}{P}}$$

where $\alpha = 1/(\beta^2 + 1)$ and P and R are the precision and recall of the retrieved set [5]. The parameter β indicates the relative importance of precision to recall; when $\beta = 1$, equal importance is attached to them. Note that for the E measure a small value indicates effective retrieval.

3.3 Retrieval Results

The evaluation output for the three searches and two values of *NumWanted* is given in Figures 4–7.

The values given in these figures are the mean values obtained by the set of queries. When $NumWanted = 10$, the output is evaluated only after the first 10 documents are retrieved. When $NumWanted = 20$, the output is evaluated after the first 10 and 20 documents are retrieved.

As expected, the retrieval output demonstrates that the absolute performance obtainable on a collection is a function of how similar the relevant documents are to one another [10]. Both the recall and the precision of the MED searches are much better than any other collection. The largest differences in performance for a particular collection are when the SEQ search is compared to a cluster search after twenty documents have been retrieved. Note particularly the recall after twenty documents have been retrieved for the CACM collection (column R(20) in Figure 5). These results support the conclusion that cluster searches are precision-oriented [5].

A more surprising result is that the relative performance of a cluster search and a sequential search seems to be independent of how well the cluster hypothesis characterizes the collection. The INDIV search was more effective after ten documents were retrieved than the SEQ search for the MED and CISI collections, while neither cluster search worked as well as SEQ for the INSPEC and

	P(10)	R(10)	E($\beta = 1$;10)
SEQ	.5767	.2773	.6375
ENTIRE	.5467	.2544	.6634
INDIV	.5967	.2759	.6334

a) *NumWanted* = 10

	P(10)	P(20)	R(10)	R(20)	E($\beta = 1$;10)	E($\beta = 1$;20)
SEQ	.5767	.5183	.2773	.4791	.6375	.5185
ENTIRE	.5333	.4517	.2495	.4080	.6702	.5837
INDIV	.5933	.4650	.2739	.4211	.6359	.5711

b) *NumWanted* = 20

Figure 4: Comparison of retrieval strategies for the MED collection.

	P(10)	R(10)	E($\beta = 1$;10)
SEQ	.2538	.2177	.8094
ENTIRE	.1673	.0960	.8913
INDIV	.1923	.1456	.8634

a) *NumWanted* = 10

	P(10)	P(20)	R(10)	R(20)	E($\beta = 1$;10)	E($\beta = 1$;20)
SEQ	.2538	.2202	.2177	.3798	.8094	.7711
ENTIRE	.1654	.1346	.0912	.1572	.8941	.8702
INDIV	.1923	.1308	.1456	.1804	.8634	.8710

b) *NumWanted* = 20

Figure 5: Comparison of retrieval strategies for the CACM collection.

	P(10)	R(10)	E($\beta = 1$;10)
SEQ	.2543	.0527	.9157
ENTIRE	.2086	.0487	.9288
INDIV	.2657	.0597	.9100

a) *NumWanted* = 10

	P(10)	P(20)	R(10)	R(20)	E($\beta = 1$;10)	E($\beta = 1$;20)
SEQ	.2543	.2443	.0527	.1071	.9157	.8600
ENTIRE	.2086	.1543	.0487	.0665	.9288	.9173
INDIV	.2686	.1914	.0596	.0813	.9100	.8968

b) *NumWanted* = 20

Figure 6: Comparison of retrieval strategies for the CISI collection.

	P(10)	R(10)	E($\beta = 1$;10)
SEQ	.3481	.1384	.8298
ENTIRE	.2494	.0990	.8775
INDIV	.2844	.1057	.8648

a) *NumWanted* = 10

	P(10)	P(20)	R(10)	R(20)	E($\beta = 1$;10)	E($\beta = 1$;20)
SEQ	.3481	.2935	.1384	.2169	.8298	.7860
ENTIRE	.2481	.1812	.0977	.1281	.8788	.8706
INDIV	.2844	.1870	.1057	.1300	.8648	.8679

b) *NumWanted* = 20

Figure 7: Comparison of retrieval strategies for the INSPEC collection.

CACM collections. It was demonstrated earlier, however, that the cluster hypothesis is not true for the CISI collection while it holds for the other collections to varying extents.

In order to predict the relative performance of cluster and sequential searches, the similarities between documents and the query need to be taken into account in addition to the similarities between relevant–relevant and relevant–non-relevant document pairs. A cluster-based search such as ENTIRE can perform more effectively than a sequential search by either retrieving relevant documents that are similar to other relevant documents but are not very similar to the query, or by not retrieving non-relevant documents that are quite similar to the query but are not similar to relevant documents. A cluster search that retrieves individual documents from the clusters can improve upon the effectiveness of the sequential search only by not retrieving non-relevant documents that are similar to the query but are not in any cluster being examined. Whether or not these situations arise depends as much upon the query as it does upon the document-document similarities.

The final result to be noted from the retrieval output is that the cluster search that retrieves individual documents almost always performed better than the search that retrieves entire clusters. The only exception is after twenty documents have been retrieved for the CACM collection. Not even in the case of the MED collection where the cluster hypothesis is clearly true did the relevant documents cluster into tight enough groups to make retrieving entire clusters worthwhile.

4 Conclusion

A new test for determining whether or not the cluster hypothesis characterizes a document collection was introduced. This test, computing the number of relevant documents that have relevant documents as nearest neighbors, was shown to be able to differentiate among collections better than the test introduced by Jardine and van Rijsbergen.

However, the extent to which the cluster hypothesis characterized a collection seemed to have little effect on how well cluster searching performed as compared to a sequential search of the collection. It should be noted that the collections for which the cluster search was better than the sequential search were smaller than the collections for which the opposite was true. The effect of a collection's size on the performance of a cluster search should be investigated more fully.

A direct comparison between retrieving entire clusters and retrieving individual documents from clusters was made. In these experiments, the search that retrieves individual documents was usually more effective than the search that retrieves entire clusters. A possible explanation of this can be found in the way the single-link hierarchy clusters documents. Even if the cluster hypothesis holds for a particular collection, clustering will not be beneficial unless the similar documents are close to one another in the cluster hierarchy. By definition, a document joins a single-link cluster if it is similar enough to any one of the other documents in the cluster. This can cause documents that are fairly similar to one another to be far apart in the hierarchy, and this in fact happens for all the collections discussed in this paper. There has been some recent experimental evidence which indicates that the single-link method may not be the best hierarchic clustering method to use for information retrieval [14]. Further study needs to be done to determine if retrieving entire clusters from other types of hierarchies is more effective.

References

[1] Salton, G., ed., (1971) *The* SMART *Retrieval System*. Prentice-Hall, Englewood Cliffs, N.J.

[2] Kerchner, M. D., (1971) Dynamic Document Processing in Clustered Collections. Ph.D. Thesis, Cornell University. Report ISR-19 to the National Science Foundation and to the National Library of Medicine.

[3] Murray, D. M., (1972) Document Retrieval Based on Clustered Files. Ph.D. Thesis, Cornell University. Report ISR-20 to the National Science Foundation and to the National Library of Medicine.

[4] Williamson, R.E., (1974) Real-time Document Retrieval. Ph.D. Thesis, Cornell University.

[5] Jardine, N. and van Rijsbergen, C. J., (1971) The Use of Hierarchic Clustering in Information Retrieval. *Inform. Stor. & Retr.*, **7**, 217–240.

[6] van Rijsbergen, C. J., (1979) *Information Retrieval*, 2nd edn. Butterworths, London.

[7] Ide, Eleanor Rose Cook, (1969) Relevance Feedback in an Automatic Document Retrieval System. Master Thesis, Cornell University. Report ISR-15 to the National Science Foundation.

[8] Fox, Edward A., (1983) Extending the Boolean and Vector Space Models of Information Retrieval with P-norm Queries and Multiple Concept Types. Ph.D. Thesis, Cornell University, pp. 44–46.

[9] Salton, G., Fox, E. A., and Wu, H., (1983) Extended Boolean Information Retrieval. *Communications of the ACM*, **26**, 1022–1036.

[10] van Rijsbergen, C. J. and Sparck Jones, K., (1973) A Test for the Separation of Relevant and Non-relevant Documents in Experimental Retrieval Collections. *Journal of Documentation*, **29**, 251–257.

[11] van Rijsbergen, C. J., (1974) Further Experiments with Hierarchic Clustering in Document Retrieval. *Inform. Stor. & Retr.*, **10**, 1–14.

[12] van Rijsbergen, C. J. and Croft, W. B., (1975) Document Clustering: An Evaluation of Some Experiments with the Cranfield 1400 Collection. *Inform. Proc. & Management*, **11**, 171–182.

[13] Croft, W. B., (1980) A Model of Cluster Searching Based on Classification. *Inform. Systems*, **5**, 189–195.

[14] Griffiths, Alan, Robinson, Lesley A., and Willett, Peter, (1984) Hierarchic Agglomerative Clustering Methods for Automatic Document Classification. *Journal of Documentation*, **40**, 175–205.