Text Linking and Retrieval Experiments for Textbook Components

Gerard Salton*
Chris Buckley*
Zhongnan Zhao*

TR 90-1125
May 1990

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

*Department of Computer Science, Cornell University, Ithaca, NY 14853-7501. This study was supported in part by the National Science Foundation under Grant IRI-87-02735.

Text Linking and Retrieval Experiments for Textbook Components

Gerard Salton, Chris Buckley and Zhongnan Zhao*

May 10, 1990

Abstract

Experiments are described designed to retrieve individual paragraphs of textbook material in answer to user-submitted queries. The retrieval strategies are based on the global comparison of paragraph texts, as well as on the local processing of text sentences. Furthermore, the retrieved items may be freely chosen, or may alternatively be restricted to certain areas in a clustered arrangement of book paragraphs. The retrieval results indicate that high retrieval values are obtainable for the more refined retrieval strategies, ranging between 0.70 and 0.80 in search precision.

1 Introduction

In a recent report, retrieval strategies were described for linking and retrieving text excerpts extracted from conventional textbook materials in response to queries obtained from text readers. [1] The process consists in first selecting a few of the top documents (top docs) that most closely match a given search request. These top docs are then used in turn as search requests as a replacement for the initial queries, and additional text excerpts are retrieved, the process being repeated several times. The similarity between a previously available top document and a newly retrieved text excerpt is obtained by performing global comparisons between the respective texts, and additionally by comparing sentence pairs included in the respective paragraphs. The basic process is outlined in Table 1.

Although the process of Table 1 provides satisfactory retrieval output in many cases, various refinements may be introduced in an attempt to render

*Department of Computer Science, Cornell University, Ithaca, NY 14853-7501. This study was supported in part by the National Science Foundation under grant IRI 87-02735.

the text linking and retrieval process more secure. The basic aim consists in providing an environment where a particular text excerpt is linked to a new text only when the respective subject areas are closely related and the vocabulary is homogeneous in both texts. The current note is then designed to examine a number of retrieval parameters, leading to an appropriate choice of retrieval strategies. The following questions are investigated:

a) The number of top documents to be retrieved initially in order to obtain satisfactory retrieval effectiveness.

b) The importance of local sentence matching in addition to the global paragraph comparisons.

c) The choice of matching functions to be used in comparing paragraph or sentence texts.

d) The retrieval threshold to be used, that is, the number of documents to be retrieved in each pass of the procedure.

e) The actual search strategy to be used, including, in particular, the implementation of clustered search strategies where retrieval activities are restricted to items located in particular text clusters.

Six basic retrieval strategies are examined in the remainder of this note based on the use of one or two initial top documents, the optional inclusion of sentence matches in addition to paragraph matches, the introduction of various text matching measures, and the addition of clustering restrictions for the retrieved items. When stringent search criteria are used, search precision results of up to 0.80 are obtainable, indicating that the vast majority of the retrieved text excerpts are properly chosen.

2 Experimental Design

The retrieval experiments described in this note reflect the average performance of ten user queries designed to retrieve excerpts from textbook material consisting of individual text paragraphs. The text corpus consists of the full text of a recent information science book which is broken down into 1,137 local documents (text paragraphs), and approximately 4,500 text sentences.[2] Six main retrieval runs are used experimentally, as shown in Table 2, and average performance results are given in each case over the ten sample queries.

The first two runs in Table 2, labelled 15 and 37, respectively, use a standard high-performance term weighting system, in which the terms attached to the document and query texts are weighted directly by the occurrence frequency of a term in a document, and inversely by the number of different documents to which the term is assigned (known as the 'term frequency times inverse document frequency' ($tf \times idf$) weight). The final product is normalized for document length to insure that all documents have an equal chance of being retrieved.[3] The normalized ($tf \times idf$) term weighting system is known to provide a high standard of performance.

Runs 15 and 37 differ from each other in that the two best documents, producing the highest matching coefficients with the query statements, are used as initial top documents for run 15, whereas only one top document is needed for run 37. In each case, a variable retrieval threshold is used in each pass, designed to retrieve more documents in each subsequent pass than in the immediately preceding pass. The retrieval criterion in each case is a global (scalar product) match between the weighted term vectors attached to queries and documents, respectively. [1, 3]

The distinction between runs 30 and 39 is the same as that between runs 15 and 37, that is, the number of initial top documents is 2 and 1, respectively. The first run of each pair thus represents a broader search than the second, because two different documents are used to start a search rather than only one.

The output of runs 30 and 39 differs from that of runs 15 and 37 in that an acceptable sentence match is needed for retrieval in the former case, in addition to the global match between the document texts. In particular, for each paragraph pair that exhibits a sufficiently high global text similarity, the individual sentences are indexed using the terms appearing in common in the corresponding paragraph pair. A sentence match is then carried out, and a new document is retrieved only if it contains at least one matching sentence in common with the original document. Requiring a sentence match between two documents in addition to the global match between document texts increases the likelihood that the corresponding documents are semantically related and cover homogeneous subject areas.

The last two runs of Table 2 are related to run 37, in the sense that only global document text comparisons are involved without separate sentence comparisons, and a single top document is used as a starting point for the search. However, different document similarity measures are used: in run 38, the ($tf \times idf$) similarity function is applied without the usual length normalization. This favors the retrieval of longer documents that may be expected to carry more matching terms than shorter documents. Run 40 uses a term-frequency weighting system that lacks any collection frequency component; in that case, documents are retrieved when they include a number of high-frequency matching terms. In earlier term weighting studies the normalized $tf \times idf$ term weighting system was preferred over other competing weighting measures.[3]

The six indexing and retrieval strategies outlined in Table 2 can be used to retrieve text excerpts directly from an unstructured collection of texts. Alternatively, a structure can be imposed on the text collection, for example, by building a cluster structure in which the text paragraphs are collected into individual <u>text clusters</u> in such a way that texts exhibiting a sufficiently high pairwise similarity are included in a common class, or cluster.[4] When a clustered document collection is used, improved search results may be obtainable by confining the search activities to certain document clusters only.

Two types of collection searches are therefore evaluated in the present study: all retrieval runs are first performed for the unclustered case where documents are freely retrievable. In addition, a clustered collection search is used for which the 1,137 text paragraphs are grouped into 71 high-level clusters using a complete-link clustering system, and each top document is restricted to retrieving items from within its own top-level cluster only.[4] This effectively restricts the search to document clusters containing one of the two top documents identified in each run.

In the remainder of this note, retrieval results are presented for clustered and unclustered collection searches, using a variety of different search parameters.

3 Experimental Output

Table 3 contains the summary output for the six main experimental runs. An overall retrieval precision figure is given for each run, computed as the sum of the relevant retrieved items over the 10 queries divided by the sum of the retrieved items for the 10 queries. Average precision figures are also given for each of the three passes performed during retrieval operations. The Table shows that the strategies using the normalized ($tf \times idf$) term weighting system are preferred, in particular the two alternatives of runs 38 and 40 appear to be less effective on average. This confirms the earlier results previously obtained in other retrieval experiments.

The best global results are produced by run 15, using a broad search with two initial top documents but no sentence comparisons. When enough initial information is used, a global text match appears to suffice. On the other hand, when only one top document is available, as in run 39, it is best to add a sentence matching system to the global text comparison process. Run 39 is second best, but the additional sentence matching requirement limits the scope of the output: only 121 items in all are retrieved by run 39 for the 10 queries compared with 161 items for run 15.

The output for the individual passes shows that, as expected, the best precision results are obtained in the first pass. Subsequent passes generate additional output but the search precision drops from one pass to the next. When only a small number of items is wanted, and precision is important, only one retrieval pass should be used. In that case, a precision of 0.60 is attainable, retrieving 40 documents in all for the 10 queries.

Table 4 contains an expanded view of the global results previously included in Table 3. In Table 4, the runs are listed in decreasing order of desirability: that is, the preferred run is 15, followed by runs 39 and 30 in order. The basic runs from Table 3 are listed again in Table 4; in addition, output labelled DD, and DD + SS, is shown. In the latter cases, the variable retrieval threshold normally used to control the number of retrieved items in each pass is replaced by a fixed retrieval threshold. The DD label identifies retrieval strategies using a fixed similarity threshold of 0.350 for the document-document comparisons. That is, each top document retrieves new items only when the pairwise global text similarity is not smaller than 0.350. As expected, such a stringent fixed threshold substantially reduces the total number of retrieved items, but produces substantially enhanced precision values.

The DD + SS output uses two fixed retrieval thresholds, including the previously mentioned global threshold of 0.350 for the document- document (DD) similarities, and a fixed threshold of at least 5.0 for the sentence-sentence (SS) similarities. (When sentences are compared, as opposed to documents, a similarity measure based, directly on the number of matching terms is used, as opposed to the normalized ($tf \times idf$) measure [4]). For runs 39 and 30 where sentence matches are automatically involved, the 39 DD and 30 DD runs are equivalent to 39 DD + SS and 30 DD + SS, respectively.

The output of Table 4 shows that when various fixed thresholds are imposed, the output performance improves considerably, in part because many nonrelevant items are no longer retrieved. This is seen most directly in the output of run 39, where the output size decreases from the original 121 to 68 when the DD + SS thresholds are used; at the same time, the global precision increases from 0.4628 to 0.5147. The comparable figures for pass one alone are 0.5833 for 24 retrieved documents, and 0.8889 for only 9 retrieved items. When only one pass is used in the retrieval process, the distinctions between the various retrieval strategies disappear, and the normally inferior similarity measures used in runs 40 and 38 prove quite acceptable. Overall, the simple search of run 15, based on a broad initial search that lacks sentence comparisons, furnishes large output sets and adequate precision ratios.

Table 5 contains a detailed analysis of the performance for each of the 20 sample queries. The performance is inadequate only for the last two queries (numbers 9 and 10). As expected, run 15 is preferred for many of the queries, but the less favored approaches using term frequency weighting (run 40) and unnormalized ($tf \times idf$) weighting (run 38) may be effective in some instances for queries 7 and 6, respectively). When short query titles are used for the initial text comparisons, as in the case of the sample queries of Table 5, query formulations that match the text vocabulary produce better output than less precise, more descriptive, query statements. This may explain why excellent results are obtained for query 1 ("electronic mail and messages") but more uncertain results for query 10 ("natural language interface to information systems").

Tables 6 and 7 cover the cluster-based retrieval results corresponding to the global searches used earlier in Tables 3 and 4. In this case, a complete-link clustering process was used to subdivide the paragraph collection into affinity classes, and the retrieval activities were confined to clusters containing the initially available top documents only. Specifically, depending on the retrieval strategy in use, one or two top documents are identified by the initial query-document comparison operations as before. In the cluster retrieval strategy, each top document is then restricted to operate within its own top-level cluster only.

For the collection of text paragraphs under study, the cluster search extends over a maximum of two clusters per query for cases where the two top documents are in fact located in different clusters. A total of 17 different clusters are involved in the cluster searches over the ten queries (out of 71 maximal top-level clusters produced by the complete-link clustering system for the sample collection of 1137 text paragraphs). In a complete-link clustering system, a maximal cluster is one whose cluster similarity with any other top-level cluster is zero; in that case, no further merging between clusters is possible.

The output of Tables 6 and 7 shows that the scope of the cluster searches is much more restricted than the scope of the original searches illustrated in Tables 3 and 4. The total number of retrieved elements ranges from 20 to 50 items for the clustered searches compared with 70 to 200 retrieved items for the global searches. Correspondingly, the cluster searches produce much better precision than the global searches.

The output of Table 7 shows that the ranking of the various search strategies in order of effectiveness is approximately the same for clustered as for unclustered searches. Once again runs 38 and 40 perform less well than the others, and runs 15 and 39 perform best. In the clustered search environment, run 39 is, however, slightly better than run 15, indicating that the more rigorous sentence-based matching strategy is actually beneficial. The overall search precision reaches 0.80 for the rigorous search strategy used in run 39 DD + SS. This particular strategy does, however, retrieve only 20 documents over the 10 queries, of which 18 prove relevant. The output data for the individual passes on the right side of Table 7 indicate once again that the first pass is the most effective. Correspondingly lower precision values are obtained for the second and third passes.

An attempt is made in Table 8 to evaluate the efficiency, as well as the effectiveness, of the 17 clusters actually used in the cluster searches. The cluster efficiency, reflecting the proportion of clustered documents actually retrieved from the 17 clusters, is low in all cases. Thus, whereas a total of 265 documents are contained in the 17 clusters being searched, only 44 of these are retrieved in run 15, and only 32 out of 265 in run 39. The cluster effectiveness, giving the proportions of relevant documents retrieved from those clusters is, however, quite high. 31 of the 44 retrieved items prove relevant for run 15, producing a cluster effectiveness (or cluster precision) of 0.7045. The effectiveness increases to 0.7188 for run 39.

In summary, methods are covered in this note for linking, and or retrieving, text excerpts based on similarities between global texts, or between particular text components, such as sentences. Depending on the strictness of the matching criteria used, the search precision ranges from a high of 0.80 when about 20 documents are retrieved by strict cluster search strategies, to a low of about 0.40 when over 200 documents are retrieved from an unstructured collection by global text comparisons.

It is hoped that such relatively high performance standards can be maintained in more heterogeneous text-environments involving texts from many different sources.

REFERENCES

1. Gerard Salton and Chris Buckley, Approaches to Text Retrieval for Structured Documents, Technical Report 90-1083, Department of Computer Science, Cornell University, Ithaca, NY., January 1990.

2. G. Salton, *Automatic Text Processing - The Transformation, Analysis, and Retrieval of Information by Computer,* Addison-Wesley Publishing Co., Reading, MA, 1989.

3. G. Salton and C. Buckley, Term Weighting Approaches in Automatic Text Retrieval, *Information Processing and Management,* 24:5, 1988, 513-523

4. G. Salton and C. Buckley, Approaches to Global Text Analysis, Technical Report No. 90-1113 , Department of Computer Science, Cornell University, Ithaca, NY, April 1990.

1. Compare incoming query with all existing document (paragraph) descriptions.
2. Retrieve one or two of the best matching documents (paragraphs); call these topdocs.
3. Use the topdocs in lieu of original query and compare them with all other documents (paragraphs). Retrieve additional new documents based on global pairwise document match plus optionally on pairwise sentence match. Use the newly retrieved items as topdocs and repeat step 3. Normally perform 3 passes.

Table 1: Basic Paragraph Retrieval Process

Run Number	Number of Initial Topdocs	Document-Document Similarity Measure	Retrieval Threshold	Sentence Indexing	Sentence-Sentence Similarity	Sentence Threshold
15	2	$tf \times idf$ normalized	Variable (set to retrieve more items than in previous pass)	–	–	–
37	1	(same)	variable (set to retrieve 3 more items than earlier)	–	–	–
30	2	$tf \times idf$ normalized	Use 10 top items for each topdoc for sentence indexing	Use all common terms in each document pair	Innermin	Variable (set to retrieve 3 more items than earlier)
39	1	(same)	(same)	(same)	(same)	(same)
38	1	$tf \times idf$ (no normalization)	variable	–	–	–
40	1	tf weights similarity depends on number of matching terms	variable	–	–	–

Table 2: Description of Experimental Runs with More Parameter Settings

Run Number	Total Relevant over Total Retrieved (10 queries)	Overall Precision (10 queries)	Average Precision and Total Number Retrieved in each Pass		
			Pass 1	Pass 2	Pass 3
15 (best)	83/161	0.5155	0.5946 (37)	0.5313 (64)	0.4500 (60)
37 (3rd)	69/151	0.4570	0.5750 (40)	0.4717 (53)	0.3621 (58)
30	72/169	0.4260	0.5366 (41)	0.4444 (54)	0.3514 (74)
39 (2nd)	56/121	0.4628	0.5833 (24)	0.4750 (40)	0.4035 (57)
38	70/171	0.4094	0.6000 (40)	0.4545 (55)	0.2763 (76)
40	80/201	0.3980	0.6304 (46)	0.4688 (64)	0.2308 (91)

Table 3: Global Retrieval Evaluation Using Variable Retrieval Thresholds and Different Numbers of Initial Top Documents

Run Number	Total Relevant over Total Retrieved (10 queries)	Overall Precision (10 queries)	Average Precision and Total Number Retrieved in each Pass		
			Pass 1	Pass 2	Pass 3
15	83/161	0.5155	0.5946 (37)	0.5313 (64)	0.4500 (60)
15+DD	66/122	0.5410	0.7895 (19)	0.5185 (54)	0.4694 (49)
15+DD+SS	56/98	0.5714	0.7333 (15)	0.5682 (44)	0.5128 (39)
39	56/121	0.4628	0.5833 (24)	0.4750 (40)	0.4035 (57)
39+DD+SS	35/68	0.5147	0.8889 (09)	0.6316 (19)	0.3750 (40)
30	72/169	0.462	0.5366 (41)	0.4444 (54)	0.3514 (74)
30+DD+SS	40/80	0.5000	0.6364 (11)	0.5172 (29)	0.4500 (40)
37	69/151	0.4570	0.5750 (40)	0.4717 (53)	0.3621 (58)
37+DD	53/200	0.4818	0.8750 (16)	0.5227 (44)	0.3200 (50)
37+DD+SS	45/99	0.4545	0.8667 (15)	0.4872 (39)	0.2889 (45)
40	80/201	0.3980	0.6304 (46)	0.4688 (64)	0.2308 (91)
40+DD	42/93	0.4516	0.8571 (14)	0.5714 (28)	0.2745 (51)
40+DD+SS	40/85	0.4706	0.8461 (13)	0.6154 (26)	0.2826 (46)
38	70/171	0.4094	0.6000 (40)	0.4545 (55)	0.2763 (76)
38+DD	40/89	0.4494	0.8462 (13)	0.5429 (35)	0.2439 (41)
38+DD+SS	38/80	0.4750	0.8333 (12)	0.6129 (31)	0.2432 (37)

Table 4: Ranked Retrieval Output for Fixed and Variable Retrieval Thresholds

Query Number	Query Formulation	Performance
1	Electronic mail and message	Excellent performance (except run 40 with tf weights). Best overall is run 38 (precision 13/13 = 1.00). Very good run 15 (16/17), run 30 (17/18), runs 39, 37 (14/15)
2	Approaches to text generation	Fair performance. Sentence runs (30 and 39) do poorly. Best run is 38 (precision 9/21 = 0.4285); run 37 acceptable (7/18 = 0.3889)
3	Extended Boolean Systems	Excellent performance. Run 15 (13/15 = 0.8667) is best; run 39 (12/14 = 0.8571); run 30 (13/16 = 0.8125)
4	Multidimensional access structures	Good performance. Best is starting with two topdocs and using sentences (run 30: 12/19 = 0.6316, run 15: 7/12 = 0.5833, run 39: 10/17 = 0.8582) Run 38 poor
5	Relational database system	Excellent performance. Best is run 15 (13/13); run 39 (8/9 = 0.8889); run 37 (13/13); run 38 also good 13/14 = 0.9286). run 40 poor; run 30 (2 topdocs plus sentences) is very poor: 2/15 = 0.13333
6	Special purpose compression systems	Good performance. Run 15: 10/15= 0.6667;run 30: 9/14 = 0.6429. Run 38 also god (9/15 = 0.6000). Remainder poor.
7	Ciphers for computationally difficult problems	Good performance. Best here is tf weights, matching term similarity (run 40: 13/16 = 0.8125). Run 30 also good (9/12 = 0.7500). Other runs not so good (run 15: 9/26 0.3462).
8	Interactive graphics editing	Good performance. Run 15 is best (10/13 = 0.7692); next best is run 37 (7/11 = 0.6364). Sentence runs are not good.
9	Data security, integrity and recovery	Poor performance. Best is run 39 (one top doc and sentences) because this retrieves very few items: 1/2 = 0.5000.
10	Natural language interface to information systems	Poor performance across the board. Best is run 37 (one topdoc, no sentences): 2/17 = 0.1176.

Table 5: Individual Query Performance

Run Number	Total Relevant over Total Retrieved (10 queries)	Overall Precision (10 queries)	Average Precision and Total Number Retrieved in each Pass		
			Pass 1	Pass 2	Pass 3
15 (2nd)	31/44	0.7045	0.6667 (21)	0.7647 (17)	0.6667 (6)
37	31/49	0.6327	0.6154 (26)	0.6923 (13)	0.6000 (10)
30 (3rd)	27/40	0.6750	0.6364 (22)	0.7000 (10)	0.7500 (8)
39 (best)	23/32	0.7188	0.6000 (15)	0.7000 (10)	1.0000 (7)
38	19/37	0.5135	0.5263 (19)	0.5000 (10)	0.5000 (8)
40	25/49	0.5102	0.6250 (24)	0.6667 (12)	0.1538 (13)

Table 6: Evaluation for Retrieval within Top Level Clusters for Variable Thresholds and Different Numbers of Initial Top Documents

Run Number	Total Relevant over Total Retrieved (10 queries)	Overall Precision (10 queries)	Average Precision and Total Number Retrieved in each Pass		
			Pass 1	Pass 2	Pass 3
39	23/32	0.7188	0.6000 (15)	0.7000 (10)	1.000 (7)
39+DD+SS	16/20	0.8000	0.8571 (7)	0.6250 (8)	1.000 (5)
15	31/44	0.7045	0.6667 (21)	0.7647 (17)	0.6667 (6)
15+DD	24/31	0.7742	0.8000 (10)	0.8000 (15)	0.6667 (6)
15+DD+SS	22/29	0.7586	0.8000 (10)	0.7692 (13)	0.6667 (6)
30	27/40	0.6750	0.6364 (22)	0.7000 (10)	0.7500 (8)
30+DD+SS	17/24	0.7083	0.7500 (8)	0.6667 (9)	0.7143 (7)
37	31/49	0.6327	0.6154 (26)	0.6923 (13)	0.6000 (10)
37+DD	22/31	0.7097	0.8182 (11)	0.7273 (11)	0.5556 (9)
37+DD+SS	19/27	0.7037	0.8182 (11)	0.6667 (9)	0.5714 (7)
38	19/37	0.5135	0.5263 (19)	0.5000 (10)	0.5000 (8)
38+DD	10/19	0.5263	0.7143 (7)	0.4286 (7)	0.4000 (5)
38+DD+SS	10/18	0.5556	0.7143 (7)	0.5000 (6)	0.4000 (5)
40	25/49	0.5102	0.6250 (24)	0.6667 (12)	0.1538 (13)
40+DD	15/26	0.5769	0.7778 (9)	0.6667 (9)	0.2500 (8)
40+DD+SS	14/25	0.5600	0.7778 (9)	0.6667 (9)	0.1429 (7)

Table 7: Ranked Retrieval Output for Retrieval Within Top Level Clusters for Fixed and Variable Retrieval Thresholds

Run Number	Average Cluster Efficiency (17 clusters for 10 queries)	Average Cluster Effectiveness (17 clusters for 10 queries)
15	0.1660 (44/265)	0.7045 (31/44)
37	0.1849 (49/265)	0.6327 (31.49)
30	0.1509 (40/265)	0.6750 (27/40)
39	0.1208 (32/265)	0.7188 (23/32)
38	0.1396 (37/265)	0.5135 (19/37)
40	0.1849 (49/265)	0.5102 (25/49)

Table 8: Average Efficiency and Effectiveness of Clusters (in Cluster Search Mode)