Representation Quality in Text Classification: An Introduction and Experiment

David D. Lewis

Computer and Information Science Dept.
University of Massachusetts
Amherst, MA 01003
lewis@cs.umass.edu

ABSTRACT

The way in which text is represented has a strong impact on the performance of text classification (retrieval and categorization) systems. We discuss the operation of text classification systems, introduce a theoretical model of how text representation impacts their performance, and describe how the performance of text classification systems is evaluated. We then present the results of an experiment on improving text representation quality, as well as an analysis of the results and the directions they suggest for future research.

1 The Task of Text Classification

Text-based systems can be broadly classified into *classification* systems and *comprehension* systems. Text classification systems include traditional information *retrieval* (IR) systems, which retrieve texts in response to a user query, as well as *categorization* systems, which assign texts to one or more of a fixed set of categories. Text comprehension systems go beyond classification to transform text in some way, such as producing summaries, answering questions, or extracting data.

Text classification systems can be viewed as computing a function from documents to one or more class values. Most commercial text retrieval systems require users to enter such a function directly in the form of a *boolean query*. For example, the query

(*language* OR *speech*) AND *AU* = *Smith*

specifies a 1-ary 2-valued (boolean) function that takes on the value TRUE for documents that are authored by Smith and contain the word *language* or the word *speech*. In statistical IR systems, which have long been investigated by researchers and are beginning to reach the marketplace, the user typically enters a natural language query, such as

Show me uses of speech recognition.

The assumption is made that the attributes (content words, in this case) used in the query will be strongly associated with documents that should be retrieved. A statistical IR system uses these attributes to construct a classification function, such as:

$$f(\mathrm{x}) = c_1 x_{show} + c_2 x_{uses} + c_3 x_{speech} + c_4 x_{recognition}$$

This function assumes that there is an attribute corresponding to each word, and that attribute takes on some value for each document, such as the number of occurrences of the word in the document. The coefficients c_i indicate the weight given to each attribute. The function produces a numeric score for each document, and these scores can be used to determine which documents to retrieve or, more usefully, to display documents to the user in ranked order:

Speech Recognition *Applications*	0.88
Jones Gives **Speech** *at Trade* **Show**	0.65
Speech *and* **Speech** *Based Systems*	0.57
⋮	

Most methods for deriving classification functions from natural language queries use statistics of word occurrences to set the coefficients of a linear discriminant function [5,20]. The best results are obtained when supervised machine learning, in the guise of *relevance feedback*, is used [21,6].

Text categorization systems can also be viewed as computing a function defined over documents, in this case a k-ary function, where k is the number of categories into which documents can be sorted. Rather than deriving this function from a natural language query, it is typically constructed directly by experts [28], perhaps using a complex pattern matching language [12]. Alternately, the function may be induced by machine learning techniques from large numbers of previously categorized documents [17,11,2].

1.1 Text Representation and The Concept Learning Model

Any text classification function assumes a particular representation of documents. With the exception of a few experimental knowledge-based IR systems [15], these text representations map documents into vectors of attribute values, usually boolean or numeric. For example, the document title "*Speech and Speech Based Systems*" might be represented as

(F, F, F, T, F, F, T, F, T, F, F, F ...)

in a system which uses boolean attribute values and omits common function words (such as *and*) from the text representation. The *T's* correspond to the words *speech*, *based*, and *systems*. The same title might be represented as

(0, 0, 0, 1.0, 0, 0, 0.5, 0, 0.5, 0, 0, 0 ...)

in a statistical retrieval systems where each attribute is given a weight equal to the number of occurrences of the word in the document, divided by the number of occurrences of the most frequent word in the document.

Information retrieval researchers have experimented with a wide range of text representations, including variations on words from the original text, manually assigned keywords, citation and publication information, and structures produced by NLP analysis [15]. Besides this empirical work, there have also been a few attempts to theoretically characterize the properties of different representations and relate them to retrieval system performance. The most notable of these attempts is Salton's *term discrimination model* [19] which says that a good text attribute is one that increases the average distance between all pairs of document vectors.

However, none of the proposed models of text representation quality addresses the following anomaly: since most text representations have very high dimensionality (large number of attributes), there is usually a legal classification function that will produce *any* desired partition of the document collection. This means that essentially all proposed text representations have the same upper bound performance. Therefore, in order to understand why one text representation is better than another, we need to take into consideration the limited ability of both humans and machine learning algorithms to produce classification functions.

The *concept learning model* of text classification [14] assumes that both machine production of classification functions (as in translation of natural language queries and relevance feedback) and human production of classification functions (as in user querying or expert construction of categorization systems) can usefully be viewed as machine learning. Whether this is a useful model of human production of classification functions is a question for experiment. If so, useful view (which remains to be determined), a wide range of theoretical results and practical techniques from machine learning, pattern recognition, and statistics will take on new significance for text classification systems.

We survey a variety of representations from the standpoint of the concept learning model in [15]. We are currently conducting several experiments to test the predictions of the model [14]. One such experiment is described in Section 2 of this paper. First, however, we discuss how text classification systems are evaluated.

1.2 Evaluation of Text Classification Systems

We have refered several times to the "performance" of text classification systems, so we should say something about how performance is measured. Retrieval systems are typically evaluated using *test collections* [24]. A test collection consists of, at minimum, a set of documents, a set of sample user queries, and a set of *relevance judgments*. The relevance judgments tell which documents are relevant (i.e. should be retrieved) for each query. The retrieval system can be applied to each query in turn, producing either a set of retrieved documents, or ranking all documents in the order in which they would be retrieved.

Two performance figures can be computed for a set of retrieved documents. *Recall* is the percentage of all relevant documents which show up in the retrieved set, while *precision* is the percentage of documents in the retrieved set which are actually relevant. Recall and precision figures can be averaged over the group of queries, or the recall precision pair for each query plotted on a scatterplot.

For systems which produce a ranking rather than a single retrieved set, there is a recall and precision figure corresponding to each point in the ranking. The average performance for a set of queries can be displayed in terms of average precision levels at various recall levels (as in Table 1) or the averages at various points can be graphed as a recall precision curve. Both methods display for a particular technique how much precision must be sacrificed to reach a particular recall level.

A single performance figure which is often used to compare systems is the average precision at 10 standard recall levels (again as in Table 1), which is an approximation to the area under the recall precision curve. A difference of 5% in these figures is traditionally called *noticeable* and 10% is considered *material* [22]. Other single figures of merit have also been proposed [27].

A large number of test collections have been used in IR research, with some being widely distributed and used by many researchers. The superiority of a new technique is not widely accepted until it has been demonstrated on several test collections. Test collections range in size from a few hundreds to a few tens of thousands of documents, with anywhere from 20 to a few hundred queries. Results on the smaller collections have often turned out to be unreliable, so the larger collections are preferred.

Evaluation is still a research issue in IR. The exhaustive relevance judgments assumed for traditional test collections are not possible with larger collections, nor when evaluating highly interactive retrieval systems [6]. For more on evaluation in IR, the reader is referred to Sparck Jones' excellent collection on the subject [25].

Evaluation of text categorization systems also needs more attention. One approach is to treat each category as a query and compute average recall and precision across categories [12], but other approaches are possible [2] and no standards have been arrived at.

2 An Experiment on Improving Text Representation

One method of improving text representation that has seen considerable recent attention is the use of syntactic parsing to create indexing phrases. These syntactic phrases are single attributes corresponding to pairs of words in one of several specified syntactic relationships in the original text (e.g. verb and head noun of subject, noun and modifying adjective, etc.). For instance, the document title

Jones Gives Speech at Trade Show

might be represented not just by the attributes

Jones, gives, speech, trade, show

but also by the attributes

<Jones gives>, <gives speech>, <speech show>, <trade show>.

Previous experiments have shown only small retrieval performance improvements from the use of syntactic phrases. Syntactic phrases are desirable text attributes since they are less ambiguous than words and have narrower meanings. On the other hand, their statistical properties are inferior to those of words. In particular, the large number of different phrases and the low frequency of occurrence of individual phrases makes it hard to estimate the relative frequency of occurrence of phrases, as is necessary for statistical retrieval methods. Furthermore, a syntactic phrase representation is highly redundant (there are large numbers of phrases with essentially the same meaning), and noisy (since redundant phrases are not assigned to the same set of documents).

2.1 Clustering of Syntactic Phrases

The concept learning model predicts that if the statistical properties of syntactic phrases could be corrected, without degrading their desirable semantic properties, then the quality of this form of representation will be improved. A number of dimensionality reduction techniques from pattern recognition potentially would have this effect [13]. One approach is to use cluster analysis [1] to recognize groups of redundant attributes and replace them with a single attribute.

We recently conducted a preliminary experiment testing this approach.[1] The titles and abstracts of the 3204 documents in the CACM-3204 test collection [9] were syntactically parsed and phrases extracted. Each phrase corresponded to a pair of content words in a direct grammatical relation. Words were stemmed [18] and the original relationship between the words was not stored. (The words are unordered in the phrases.)

Phrases were clustered using a nearest neighbor clustering technique, with similarity between phrases based on their tendency to occur in documents assigned to the same *Computing Reviews* categories.[2] Each of the 6922 phrases which occurred in two or more documents was used as the seed for a cluster, so 6922 clusters were formed. A variety of thresholds on cluster size and minimum similarity were explored. Document scores were computed using the formulae for word and phrase weights used in Fagan's study of phrasal indexing [8] and Crouch's work on cluster indexing [7].

Precision figures at 10 recall levels are shown in Table 1 for words, phrases combined with words, and clusters combined with words. While phrase clusters did improve performance, as is not always the case with clusters of individual words, the hypothesis that phrase clusters would be better identifiers than individual phrases was not supported. A number of variations on the criteria for membership in a cluster were tried, but none were found to give significantly better results. In the next section we discuss a number of possible causes for the observed performance levels.

2.2 Analysis

Can we conclude from Table 1 that clustering of syntactic phrases is not a useful technique for information retrieval? No—the generation of performance figures is only the beginning of the analysis of a text classification technique. Controlling for all variables that might affect performance is usually impossible due to the complexity of the techniques used and the richness and variety of the texts which might be input to these systems. Further analysis, and usually further experiment, is necessary before strong conclusions can be reached.

In this section we examine a range of possible reasons for the failure of syntactic phrase clusters to significantly improve retrieval performance. Our goal is to discover what the most significant influences were on the performance of syntactic phrase clusters, and so suggest what direction this research should take in the future.

2.2.1 Document Scoring Method

The first possibility to consider is that there is nothing wrong with the clusters themselves, but only with how we used them. In other words, the coefficients of the classification functions derived from queries, or the numeric values assigned to the cluster attributes, might have been inappropriate. There is some merit in this suggestion, since the cluster and phrase weighting methods currently used are heuristic, and are based on experiments on relatively few collections. More theoretically sound methods of phrase and cluster weighting are being investigated [6,26].

On the other hand, scoring is unlikely to be the only problem. Simply examining a random selection of clusters (the seed member for each is underlined)

[1] Full details are found in [16].

[2] Only 1425 of the 3204 CACM documents had *Computing Reviews* categories assigned, so only phrases that appeared in these documents were clustered.

	Precision					
Recall	Clusters + Terms				Phrases	
Level	Size 2	Size 4	Size 8	Size 12	+ Terms	Terms
0.10	55.5	55.5	57.9	57.1	58.1	56.3
0.20	43.2	42.0	42.2	41.9	45.4	41.0
0.30	37.7	37.0	36.5	36.2	38.0	35.7
0.40	31.1	30.5	30.8	30.0	30.2	29.6
0.50	23.3	23.3	22.2	22.3	23.4	22.0
0.60	19.5	19.3	18.2	18.3	19.0	18.8
0.70	13.5	13.3	13.3	13.3	13.7	13.8
0.80	9.2	9.4	9.4	9.3	9.5	9.9
0.90	5.5	5.8	5.6	5.6	5.6	6.1
1.00	4.2	4.1	4.1	4.1	4.1	4.7
Avg. Prec.	24.3	24.0	24.0	23.8	24.7	23.8
Change	+2.1%	+0.8%	+0.8%	+0.0%	+3.8%	

Table 1: Performance Using Phrase Clusters, Individual Phrases, and Terms

Collection	Stemmed	
Frequency	Number of	Total Phrase
(in 1425 Docs)	Distinct Phrases	Occurrences
1	32470	34689
2	4056	8866
3	1284	4299
4	576	2584
5	309	1735
6	218	1503
7	108	855
8	90	814
9+	281	5176
Total	39392	60521

Table 2: Statistics on Phrase Generation for 1425 CACM Documents

{<*linear function*>, <*comput measur*>, <*produc result*>, <*log bound*> }

{<*princip featur*>, <*draw design*>, <*draw display*>, <*basi spline*>, <*system repres*> }

{<*error rule*>, <*explain techniqu*>, <*program involv*>, <*key data*> }

{<*substant increas*>, <*time respect*>, <*increase program*>, <*respect program*>}

shows they leave much to be desired as content indicators. We therefore need to consider reasons why the clusters formed were inappropriate.

2.2.2 Statistical Problems

The simplest explanation for the low quality of clusters is that not enough text was used in forming them. Table 2 gives considerable evidence that this is the case. The majority of occurrences of phrases were of phrases that occurred only once, and only 17.6% of distinct phrases occurred two or more times. We restricted cluster formation to phrases that occurred at least twice, and most of these phrases occurred exactly twice. This means that we were trying to group phrases based on the similarity of distributions estimated from very little data. Church [3] and others have stressed the need for large amounts of data in studying statistical properties of words, and this is even more necessary when studying phrases, with their lower frequency of occurrence.

Another statistical issue arises in the calculation of similarities between phrases. We associated with each phrase a vector of values of the form $n_{pc}/\sum n_{qc}$, where n_{pc} is the number of occurrences of phrase p in documents assigned to *Computing Reviews* category c, and the denominator is the total number of occurrences of all phrases in category c. This is the maximum likelihood estimator of the probability that a randomly selected phrase from documents in the category will be the given phrase. Similarity between phrases was computed by applying the cosine correlation [1] to these vectors. Problems with the maximum likelihood estimator for small samples are well known [10,4], so it is possible that clustering will be improved by the use of better estimators.

Another question is whether the clustering method used might be inappropriate. Previous research in IR has not found large differences between different methods for clustering words, and all clustering methods are likely to be affected by the other problems described in this section, so experimenting with different clustering methods probably deserves lower priority than addressing the other problems discussed.

A final issue is raised by the fact that using clusters and phrases together (see Table 3) produced performance superior to using either clusters or phrases alone. One way of interpreting this is that the seed phrase of a cluster is a better piece of evidence for the presence of the cluster than are the other cluster members. This raises the possibility that explicit clusters should not be formed at all, but rather that every phrase be considered good evidence for its own presence, and somewhat less good evidence for the presence of phrases with similar distributions.[3] Again, investigating this is not likely to

[3] Ken Church suggested this idea to us.

	Precision					
Recall	Clusters + Phrases + Terms				Phrases	
Level	Size 2	Size 4	Size 8	Size 12	+ Terms	Terms
0.10	57.4	60.0	59.3	58.5	58.5	56.3
0.20	46.4	46.4	46.1	45.0	45.4	41.0
0.30	38.8	39.5	38.9	37.7	38.0	35.7
0.40	31.3	31.1	31.1	30.8	30.2	29.6
0.50	23.0	23.1	23.1	23.1	23.4	22.0
0.60	19.3	19.5	19.5	19.5	19.0	18.8
0.70	13.9	13.9	13.8	13.7	13.7	13.8
0.80	9.6	9.8	9.7	9.6	9.5	9.9
0.90	5.7	5.7	5.7	5.7	5.6	6.1
1.00	4.2	4.2	4.2	4.2	4.1	4.7
Avg. Prec.	25.0	25.3	25.1	24.8	24.7	23.8
Change	+5.0%	+6.3%	+5.5%	+4.2%	+3.8%	

Table 3: Performance Using Clusters, Phrases, and Terms

be profitable until other problems are addressed.

2.2.3 Weaknesses in Syntactic Phrase Formation

Another set of factors potentially affecting the performance of phrase clustering is the phrases themselves. Our syntactic parsing is by no means perfect, and incorrectly produced phrases could both cause bad matches between queries and documents, and interfere with the distributional estimates that clustering is based on.

It is difficult to gauge directly the latter effect, but we can measure whether syntactically malformed phrases seem to be significantly worse content identifiers than syntactically correct ones. To determine this we found all matches between queries and relevant documents on syntactic phrases. We examined the original query text to see whether the phrase was correctly formed or whether it was the result of a parsing error, and did the same for the phrase occurrence in the document. We then gathered the same data for about 20% of the matches (randomly selected) between queries and *non*relevant documents.

The results are shown in Table 4. We see that for both relevant and nonrelevant documents, the majority of matches are on syntactically correct phrases. The proportion of invalid matches is somewhat higher for nonrelevant documents, but the relatively small difference suggests that syntactically malformed phrases are not a primary problem.

	Query / Relevant Document Matches (229 Pairs Total)	
	Correct Phrase in Doc	Flawed Phrase in Doc
Correct Phrase in Query	84.3% (193)	6.6% (15)
Flawed Phrase in Query	3.5% (8)	4.8% (11)

	Query / Nonrelevant Document Matches (424 Pairs in Random Sample)	
	Correct Phrase in Doc	Flawed Phrase in Doc
Correct Phrase in Query	77.6% (324)	13.0% (55)
Flawed Phrase in Query	4.5% (19)	5.0% (21)

Table 4: Syntactic Correctness of Query Phrases and their Occurrences in Documents

2.2.4 Correct Phrases with Poor Semantics

In proposing the clustering of syntactic phrases, we argued that the semantic properties of individual phrases were good, and only their statistical properties needed improving. This clearly was not completely true, since phrases such as *paper gives* (from sentences such as *This paper gives results on...*) are clearly very bad indicators of a document's content.

We believed, however, that such phrases would tend to cluster together, and none of the phrases in these clusters would match query phrases. Unfortunately, almost the opposite happened. While we did not gather statistics, it appeared that these bad phrases, with their relatively flat distribution, proved to be similar to many other phrases and so were included in many otherwise coherent clusters.

Some of the low quality phrases had fairly high frequency. Since IR research on clustering of individual words has shown omitting high frequency words from clusters to be useful, we experimented with omitting high frequency phrases from clustering. This actually degraded performance. Either frequency is less correlated with attribute quality for phrases than for words, or our sample was too small for reliable frequency estimates, or both.

Fagan, who did the most comprehensive study [8] of phrasal indexing to date, used a number of techniques to screen out low quality phrases. For instance, he only formed phrases which contained a head noun and one of its modifiers, while we formed phrases from all pairs of syntactically connected content words. Since many of our low quality phrases resulted from main verb / argument combinations, we will reconsider this choice.

Fagan also maintained a number of lists of semantically general content words that were to be omitted from phrases, and which triggered special purpose phrase formation rules. We chose not to replicate this technique, due to the modifications required to our phrase generator, and our misgivings about a technique that might require a separate list of exemption words for each corpus.

We did, however, conduct a simpler experiment which suggests that distinguishing between phrases of varying qualities will be important. We had a student from our lab who was not working on the phrase clustering experiments identify for each CACM query a set of pairs of words he felt to be good content identifiers. We then treated these pairs of words just as if they had been the set of syntactic phrases produced from the query. This gave the results shown in Table 5. As can be seen, retrieval performance was considerably improved, even though the phrases assigned to documents and to clusters did not change. (More results on eliciting good identifiers from users are discussed in [6].)

Given this evidence that not all syntactic phrases were equally desirable identifiers, we tried one more experiment. We have mentioned that many poor phrases had relatively flat distributions across the *Computing Reviews* categories. Potentially this very flatness might be used to detect and screen out these low quality phrases. To test this belief, we ranked all phrases which occurred in 8 or more documents by the similarity of their *Computing Reviews* vectors to that of a hypothetical phrase with even distribution across all categories.

The top-ranked phrases, i.e. those with the flattest distributions, are found in Table 6. Unfortunately, while some of these phrases are bad identifiers, others are reasonably good. More apparent is a strong correlation between flatness of distribution and occurrence in a large number of documents. This suggests that once again we are being tripped up by small sample estimation problems, this time manifesting itself as disproportionately skewed distributions of low frequency phrases. The use of better estimators may help this technique, but once again a larger corpus is clearly needed.

3 Future Work

The fact that phrase clusters provided small improvements in performance is encouraging, but the most clear conclusion from the above analysis is that syntactic phrase clustering needs to be tried on much larger corpora. This fact poses some problems for evaluation, since the CACM collection is one of the larger of the currently available IR test collections. The need for larger IR test collections is widely recognized, and methods for their construction have been planned in detail [23], but financial support has not yet materialized.

	Similarity	
Phrase	to (1...1)	No. Docs
DESCRIB PAPER	.61	57
ALGORITHM PRESENT	.56	64
DESIGN SYSTEM	.55	54
COMPUT SYSTEM	.54	75
SYSTEM USE	.52	43
PAPER PRESENT	.51	47
LANGUAG PROGRAM	.48	71
DESCRIB SYSTEM	.47	26
REQUIR TIME	.47	22
DATA STRUCTUR	.47	38
PROCESS SYSTEM	.46	21
INFORM SYSTEM	.45	26
OPER SYSTEM	.44	59
PROGRAM USE	.44	27
MODEL SYSTEM	.44	26
EXECUT TIME	.43	28
PROBLEM SOLUT	.43	45
REQUIR STORAG	.42	24
TECHNIQU USE	.42	40
GENER SYSTEM	.41	22

Table 6: Syntactic Phrases with Least Skewed Distribution Across Computing Reviews Categories, and Number of Documents They Appear In

Until larger IR test collections are available, we are pursuing two other approaches for experimenting with phrase clustering. The first is to form clusters on a corpus different from the one on which the retrieval experiments are performed. If the content and style of the texts are similar enough, the clusters should still be usable. To this end, we have obtained a collection of approximately 167,000 MEDLINE records (including abstracts and titles, but no queries or relevance judgments) to be used in forming clusters. The clusters will be tested on two IR test collections which, while much smaller, are also based on MEDLINE records.

A second approach is to experiment with text categorization, rather than text retrieval, since large collections of categorized text are available. The same large MEDLINE subset described above can be used for this kind of experiment, and we have also obtained the training and test data (roughly 30,000 newswire stories) used in building the CONSTRUE text categorization system [12].

Besides the need for repeating the above experiments with more text, our analysis also suggests that some method of screening out low quality phrases is needed. We plan to experiment first with restricting phrases to nouns plus modifiers, as Fagan did, and with screening out phrases based on flatness of distribution, using more text and better small sample estimatiors. Improving the syntactic parsing method does not seem to be an immediate need.

	Precision					
Recall	Clusters + Terms				Phrases	
Level	Size 2	Size 4	Size 8	Size 12	+ Terms	Terms
0.10	60.7	61.9	61.5	61.4	61.4	56.3
0.20	45.8	45.9	45.9	45.9	45.2	41.0
0.30	40.6	40.3	39.8	39.8	39.5	35.7
0.40	34.2	33.4	33.5	33.5	33.2	29.6
0.50	25.0	25.1	25.2	25.2	25.3	22.0
0.60	19.8	20.7	20.7	20.6	20.9	18.8
0.70	13.8	14.6	14.5	14.6	14.6	13.8
0.80	9.4	10.2	10.0	10.0	10.0	9.9
0.90	5.6	6.3	6.2	6.3	6.2	6.1
1.00	4.2	4.9	4.9	4.9	4.9	4.7
Avg. Prec.	25.9	26.3	26.2	26.2	26.1	23.8
Change	+8.8%	+10.5%	+10.1%	+10.1%	+9.7%	

Table 5: Performance With Human-Selected Query Phrases

4 Summary

Text-based systems of all kinds are an area of increasing research interest, as evidenced by the recent AAAI Symposium on Text-Based Intelligent Systems, by funding initiatives such as the TIPSTER portion of the DARPA Strategic Computing Program, and by an increase in the number of research papers proposing the application of various artificial intelligence techniques to text classification problems. This interest is driven by both an undeniable need to cope with large amounts of data in the form of online text, and by the resource that this text represents for intelligent systems.

In this paper we have discussed the nature of text classification, which is the central task of most current text-based systems, and which is an important component of most proposed text comprehension systems, as well. We introduced a theoretical model of how text representation impacts the performance of text classification systems, and described how the performance of these systems is typically evaluated.

We also summarized the results of our ongoing research on syntactic phrase clustering. Perhaps the most important point to stress about this work is the complexity of evaluating text classification systems, particularly those involving natural language processing or machine learning techniques, and the need to examine results carefully.

This should not discourage evaluation, however. If it is difficult to verify good text classification techniques through controlled experiments, it is impossible to do so purely through intuition or theoretical arguments. The history of IR is full of plausible techniques which experiment has shown to be ineffective. Only through careful evaluation will progress be likely.

Acknowledgements

The work reported here is part of my dissertation work under the supervision of W. Bruce Croft, whose guidance has been invaluable. We thank Longman Group, Ltd. for making available to us the online version of the Longman Dictionary of Contemporary English. This research was supported by the NSF under grant IRI-8814790, by AFOSR under grant AFOSR-90-0110, and by an NSF Graduate Fellowship. Anil Jain, Mike Sutherland, and Mel Janowicz provided advice on cluster analysis. The syntactic parser used was designed and built by John Brolio, who was also the principal designer of the syntactic grammar. Raj Das generated the hand-selected phrases for the CACM queries. Robert Krovetz and Howard Turtle made helpful comments on a draft of this paper.

References

[1] Michael R. Anderberg. *Cluster Analysis for Applications.* Academic Press, New York, 1973.

[2] Peter Biebricher, Norbert Fuhr, Gerhard Lustig, Michael Schwantner, and Gerhard Knorz. The automatic indexing system AIR/PHYS—from research to application. In *Eleventh International Conference on Research & Development in Information Retrieval*, pages 333–342, 1988.

[3] Kenneth Church, William Gale, Patrick Hanks, and Donald Hindle. Parsing, word associations, and typical predicate-argument relations. In *Second DARPA Speech and Natural Language Workshop*, pages 75–81, Cape Cod, MA, October 1990. Also appeared in Proceedings of the International Workshop on Parsing Technologies, CMU, 1989.

[4] Kenneth W. Church and William A. Gale. Enhanced Good-Turing and Cat-Cal: Two new methods for estimating probabilities of English bigrams. In *Second DARPA Speech and Natural Language Workshop*, pages 82–91, Cape Cod, MA, October 1990.

[5] W. B. Croft. Experiments with representation in a document retrieval system. *Information Technol-*

ogy: Research and Development, 2:1–21, 1983.

[6] W. Bruce Croft and Raj Das. Experiments with query acquisition and use in document retrieval systems. In *Thirteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1990. To appear.

[7] Carolyn J. Crouch. A cluster-based approach to thesaurus construction. In *Eleventh International Conference on Research & Development in Information Retrieval*, pages 309–320, 1988.

[8] Joel L. Fagan. *Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods.* PhD thesis, Department of Computer Science, Cornell University, September 1987.

[9] Edward A. Fox, Gary L. Nunn, and Whay C. Lee. Coefficients for combining concept classes in a collection. In *Eleventh International Conference on Research & Development in Information Retrieval*, pages 291–307, 1988.

[10] Norbert Fuhr and Hubert Huther. Optimum probability estimation from empirical distributions. *Information Processing and Management*, pages 493–507, 1989.

[11] Karen A. Hamill and Antonio Zamora. The use of titles for automatic document classification. *Journal of the American Society for Information Science*, pages 396–402, 1980.

[12] Philip J. Hayes, Laura E. Knecht, and Monica J. Cellio. A news story categorization system. In *Second Conference on Applied Natural Language Processing*, pages 9–17, 1988.

[13] J. Kittler. Feature selection and extraction. In Tzay Y. Young and King-Sun Fu, editors, *Handbook of Pattern Recognition and Image Processing*, pages 59–83. Academic Press, Orlando, 1986.

[14] David D. Lewis. *Representation and Learning in Information Retrieval.* PhD thesis, University of Massachusetts at Amherst, 1990. In preparation.

[15] David D. Lewis. Text representation for text classification. In P. S. Jacobs, editor, *Text-Based Intelligent Systems: Current Research in Text Analysis, Information Extraction, and Retrieval*, 1990. Selected papers from the AAAI Spring Symposium on Text-Based Intelligent Systems. Available as a technical report from General Electric Research & Development, Schenectady, NY, 12301.

[16] David D. Lewis and W. Bruce Croft. Term clustering of syntactic phrases. In *Thirteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1990. To appear.

[17] M. E. Maron. Automatic indexing: An experimental inquiry. *Journal of the Association for Computing Machinery*, 8:404–417, 1961.

[18] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, July 1980.

[19] G. Salton, C. S. Yang, and C. T. Yu. A theory of term importance in automatic text analysis. *Journal of the American Society for Information Science*, pages 33–44, January-February 1975.

[20] Gerard Salton. Another look at automatic text-retrieval systems. *Communications of the ACM*, 29(7):648–656, July 1986.

[21] Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.

[22] K. Sparck Jones and R. G. Bates. Research on automatic indexing 1974 - 1976 (2 volumes). Technical report, Computer Laboratory. University of Cambridge, 1977.

[23] K. Sparck Jones and C. J. van Rijsbergen. Report on the need for and provision of an 'ideal' information retrieval test collection. Technical report, University Computer Laboratory; University of Cambridge, 1975.

[24] K. Sparck Jones and C. J. van Rijsbergen. Information retrieval test collections. *Journal of Documentation*, 32(1):59–75, 1976.

[25] Karen Sparck Jones, editor. *Information Retrieval Experiment.* Butterworths, London, 1981.

[26] Howard Turtle and W. Bruce Croft. Inference networks for document retrieval. In *Thirteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1990.

[27] Cornelis J. van Rijsbergen. Retrieval effectiveness. In Karen Sparck Jones, editor, *Information Retrieval Experiment*, chapter 3. Butterworths, London, 1981.

[28] Natasha Vleduts-Stokolov. Concept recognition in an automatic text-processing system for the life sciences. *Journal of the American Society for Information Science*, 38:269–287, 1987.