Chapter 1

COMBINING APPROACHES TO INFORMATION RETRIEVAL

W. Bruce Croft
Department of Computer Science
University of Massachusetts, Amherst
croft@cs.umass.edu

Abstract The combination of different text representations and search strategies has become a standard technique for improving the effectiveness of information retrieval. Combination, for example, has been studied extensively in the TREC evaluations and is the basis of the "meta-search" engines used on the Web. This paper examines the development of this technique, including both experimental results and the retrieval models that have been proposed as formal frameworks for combination. We show that combining approaches for information retrieval can be modeled as combining the outputs of multiple classifiers based on one or more representations, and that this simple model can provide explanations for many of the experimental results. We also show that this view of combination is very similar to the inference net model, and that a new approach to retrieval based on language models supports combination and can be integrated with the inference net model.

1 INTRODUCTION

Information retrieval (IR) systems are based, either directly or indirectly, on models of the retrieval process. These retrieval models specify how representations of text documents and information needs should be compared in order to estimate the likelihood that a document will be judged relevant. The estimates of the relevance of documents to a given query are the basis for the document rankings that are now a familiar part of IR systems. Examples of simple models include the probabilistic or Bayes classifier model (Robertson and Sparck Jones, 1976; Van Rijsbergen, 1979) and the vector space model (Salton et al., 1975). Many others have been proposed and are being used (Van Rijsbergen, 1986; Deerwester et al., 1990; Fuhr, 1992; Turtle and Croft, 1992).

As these retrieval models were being developed, many experiments were carried out to test the effectiveness of these approaches. Quite early in these experiments, it was observed that different retrieval models, or alternatively, variations on ranking algorithms, had surprisingly low overlap in the relevant documents that were found, even when the overall effectiveness of the algorithms was similar (e.g. McGill et al., 1979; Croft and Harper, 1979). Similar studies showed that the practice of searching on multiple document representations such as title and abstract or free text and manually assigned index terms was more effective than searching on a single representation (e.g. Fisher and Elchesen, 1972; McGill et al., 1979; Katzer et al., 1982). These, and other studies, suggested that finding all the relevant documents for a given query was beyond the capability of a single simple retrieval model or representation.

The lack of overlap between the relevant documents found by different ranking algorithms and document representations led to two distinct approaches to the development of IR systems and retrieval models. One approach has been to create retrieval models that can explicitly describe and combine multiple sources of evidence about relevance. These models are typically probabilistic and are motivated by the Probability Ranking Principle (Robertson, 1977), which states that optimal retrieval effectiveness is achieved by ranking documents in decreasing order of probability of relevance and that "probabilities are estimated as accurately as possible on the basis of whatever data has been made available to the system". The INQUERY system, for example, is based on a probabilistic model that is explicitly designed to combine evidence from multiple representations of documents and information needs (Turtle and Croft, 1991; Callan et al., 1995a). The other approach has been to design systems that can effectively combine the results of multiple searches based on different retrieval models. This combination can be done in a single system architecture (e.g. Croft and Thompson, 1987; Fox and France, 1987) or in a distributed, heterogeneous environment (e.g. Lee, 1995; Voorhees et al., 1995; Callan et al., 1995b). Combining multiple, heterogeneous searches is the basis of the "metasearch" engines on the Web (e.g., MetaCrawler[1]) and has become increasingly important in multimedia databases (e.g. Fagin, 1996).

The motivation for both these approaches is to improve retrieval effectiveness by combining evidence. Apart from the empirical results, theoretical justification for evidence combination is provided by a Bayesian probabilistic framework (e.g. Pearl, 1988). In this framework, we can describe how our belief in a hypothesis H is incrementally affected by a new piece of evidence e. Specifically, using log-odds:

$$\log O(H|E,e) = \log O(H|E) + \log L(e|H)$$

[1] `http://www.metacrawler.com`

where E is all the evidence seen prior to e,
$O(H|E) = \frac{P(H|E)}{P(\neg H|E)}$ is the *posterior odds* on H given evidence E,
$O(H|E, e)$ is the *odds* on H given the new evidence e, and
$L(e|H) = \frac{P(e|H)}{P(e|\neg H)}$ is the *likelihood ratio* of evidence e.

This formulation makes it clear that each additional piece of positive evidence (i.e. with likelihood > 1) increases the odds of the hypothesis being true. A piece of evidence with very strong likelihood can have a substantial impact on the odds. In addition, the effect of a large error in the estimation of the likelihood for one piece of evidence can be reduced by additional evidence with smaller errors. In other words, the average error can be smaller with more evidence.

This analysis assumes that the evidence is conditionally independent and, therefore, $P(e|E, H) = P(e|H)$. If, however, the new evidence is correlated with the previous evidence, the impact of that new evidence will be reduced. If the new evidence can be directly inferred from the previous evidence, $P(e|E, H) = 1$ and the probability of the hypothesis being true does not change. In retrieval models, the hypothesis of *relevance* (R) is based on the observation of (or evidence about) *document* (D) contents and a specific *query* (Q). Estimating $P(R|D, Q)$ could be viewed, then, as accumulating pieces of evidence from the representations of documents and queries, such as additional words or index terms. Accumulating more pieces of evidence should result in more accurate estimates of the probability of relevance, if the evidence is uncorrelated. As we will see, retrieval models often introduce intermediate concepts that make the relationship between observations and hypothesis less direct, but this simple model supports the basic intuition of evidence combination.

Similarly, combining the output of ranking algorithms or search systems can be modeled as a combination of classifiers, which has been shown to reduce classification error (Tumer and Ghosh, 1999). A search system can be viewed as a classifier for the classes *relevant* and *nonrelevant*. For a given document, the search system's output corresponds to a probability of that document belonging to the *relevant* class. In this framework, classification errors reduce retrieval effectiveness. Misclassifying a relevant document reduces recall and misclassifying a nonrelevant document reduces precision. The amount of error reduction from combination depends on the correlation of the classifier outputs, with uncorrelated systems achieving the maximum reduction. We will show that this model provides an explanation for many of the phenomena observed in combination experiments (e.g., Vogt and Cottrell, 1998), such as the increased probability of relevance for a document ranked highly by different systems. It also provides a simple prescription of the conditions for optimum combination.

Despite the popularity of the combination approach (sometimes called "fusion"), what is known about it is scattered among many papers covering different areas of IR. One of the main goals of this paper is to summarize the research in this area. The summary will show how combination has been applied to many aspects of IR systems, and will discuss the successes and limitations of this research. The most obvious limitation is that there is no clear description of how representations, retrieval algorithms, and search systems should be combined for optimum effectiveness. By comparing and analyzing previous research using the terminology of classifier combination and inference nets, we hope to improve this situation. We will also describe how a new approach to probabilistic retrieval based on *language models* (Ponte and Croft, 1998; Miller et al., 1999; Berger and Lafferty, 1999) provides mechanisms for representing and combining sources of evidence for IR, and that this approach can be integrated with the inference net model to provide an improved framework for combination.

Although our focus in this paper is primarily on combination techniques for improving *retrieval* effectiveness, combination has been applied to a number of related tasks, such as *filtering* (Hull et al., 1996) and *categorization* (Lewis and Hayes, 1994; Larkey and Croft, 1996), and has been studied in other fields such as *machine learning* (Mitchell, 1997). We will refer to work in these areas in a number of sections of the paper.

In the following sections, we describe the research that has been done on combination applied to different levels of an IR system. Section 2 describes how different representations of text and documents can be generated and how they have been combined. Section 3 describes research related to combining different representations of information needs (queries). Section 4 describes how ranking algorithms can be combined and the results of that combination. Section 5 describes how the output from different search systems can be combined and the effectiveness of such combinations. In Section 6, we describe some of the retrieval models that have been proposed for combining all the evidence about relevance in a single framework. Finally, in section 7, we describe the language model approach to retrieval and show how it can be used to support combination.

In discussions of retrieval effectiveness in this paper, we assume familiarity with the standard recall and precision measures used for evaluations of information retrieval techniques (Van Rijsbergen, 1979). Although specific performance improvements are discussed for some experiments, it is in general difficult to compare the results from multiple studies because of the variations in the baselines and test collections that are used. For example, a combination technique that produces a 20% improvement in average precision in one study may not yield any improvement in another study that uses a more effective search as the baseline. For this reason, we focus on general summaries of the research rather than detailed comparisons.

2 COMBINING REPRESENTATIONS

2.1 MANUAL AND AUTOMATIC INDEXING

The use of multiple representations of document content in a single search appears to have started with intermediaries searching bibliographic databases. These databases typically contain the titles, abstracts, and authors of scientific and technical documents, along with other bibliographic information and manually assigned *index terms*. Index terms are selected from a *controlled vocabulary* of terms by *indexers* based on their reading of the abstract. The query languages supported by typical bibliographic search systems allow the searcher to specify a Boolean combination of words, possibly restricted by location, and index terms as the retrieval criterion. An example of this would be

> (DRUGS in TI) and (DRUG near4 MEMORY) and (SIDE_EFFECTS_DRUG in DE)

This query is designed to find documents about the side effects of drugs related to memory. It specifies that the word "drugs" should be in the title of the document, and that the text of the document should contain the word "drug" within 4 words of the word "memory", and that the controlled vocabulary term "side_effects_drug" has been used to index the document (i.e. it is present in the descriptor or DE field).

Early studies showed the potential effectiveness of this search strategy. For example, Fisher and Elchesen, 1972, showed that searching title words in combination with index terms was better than searching either representation alone. Svenonius, 1986, in a review of research related to controlled vocabulary and text indexing, makes it clear that the two representations have long been thought of, and used, as complementary. A number of major studies, such as the Cranfield tests (Cleverdon, 1967), the SMART experiments (Salton, 1971), and the Cambridge experiments (Sparck Jones, 1974), also used multiple representations of documents but focused on establishing the relative effectiveness of each representation, rather than on the effectiveness of combinations of representations. The Cranfield tests considered 33 different representations and a number of these were combinations of simpler representations. The major classes of representations, however, were considered separately. Specifically, there were representations based on single words from the text of the documents ("free text"), representations based on controlled index terms, and representations based on "concepts". Some single word representations were combinations of other single word representations, and similarly for index term representations, but there were no representations that were combinations of single words and index terms. The conclusion of the Cranfield study was that single word representations appeared to perform somewhat better than index term and concept representations, but no mention was made of combining them.

The first large-scale evaluation of combining representations was reported in Katzer et al., 1982. This study was based on a previous study (McGill et al., 1979) that found low overlap in the documents retrieved by different representations for the same queries. Katzer et al considered representations based on free text and controlled vocabularies. They found that different representations retrieve quite different sets of documents in a Boolean search system. There was low overlap between the relevant document sets retrieved by the representations (about 28% on average) and even lower overlap between all the documents in the retrieved sets (about 13% on average). Despite this, there was little difference in retrieval effectiveness for the representations. In addition, documents with a high probability of relevance had the highest overlap in the retrieved sets and each representation retrieved some unique relevant documents. Using the same data, Rajashekar and Croft, 1995, showed that significant effectiveness improvements could be obtained by combining free text and controlled vocabulary indexing in a probabilistic retrieval system. Turtle, 1990, obtained a similar result with a different set of data. Fox, 1983, carried out combination experiments with controlled vocabulary using a retrieval system based on the vector space model and was also able to improve effectiveness. In each of the experiments using search systems based on ranking, the best results were obtained when the controlled vocabulary representation was treated as weaker evidence than the free text representations. The methods of weighting evidence and choosing weights are important aspects of the overall framework for combining evidence. These frameworks are discussed further in section 6.

Controlled vocabulary terms are only one of the alternative representations of documents that have been studied in combination experiments. *Citations, passages, phrases, names* and *multimedia objects* have all been considered as sources of evidence about relevance. We will describe each of these here. In order to use some of these representations in a retrieval system, extended representations of the information need (i.e. the query) will also be needed. For example, controlled vocabulary terms could be manually included in a query formulation, as shown in the example query above. Other techniques for extending the query with alternate representations are possible, such as relevance feedback (e.g., Salton et al., 1983), and these techniques will be discussed further in section 3. Query extension is not, however, required for all representations. Controlled vocabulary terms can be matched directly with free text queries, for example, and retrieval results can be altered by relationships, such as citations, between documents.

2.2 CITATIONS

Citations have long been recognized as an alternative document representation (Salton, 1968; Small, 1973). A number of studies have established that there is low overlap in the documents found using citation representations compared to those found using word or index term representations (e.g., Pao and Worthen, 1989). Retrieval experiments that combine citations with other representations have established that significant effectiveness benefits can be obtained (Salton, 1974; Fox et al., 1988; Croft et al., 1989; Turtle, 1990). The best results in these studies, which used relatively small data sets, improved average precision by 5-10%. It was also consistently found that the evidence for relevance provided by citations was weaker than that provided by the word-based representation. The citation approach was extended to include hypertext links (Frisse and Cousins, 1989; Croft and Turtle, 1989) and is now being used as the basis for some Web search engines (e.g., Google[2]). Detailed evaluations of representations based on Web "citations" are not available, but qualitatively the techniques appear to scale well to these very large databases.

2.3 PASSAGES

The basic premise behind passage retrieval is that some parts (or passages) of a document may be more relevant to a query than other parts. By representing a document as a collection of passages rather than a monolithic block of text, more accurate retrieval may be possible (O'Connor, 1975; O'Connor, 1980). A number of definitions of document passages are possible (Callan, 1994). *Discourse* passages are based on textual discourse units such as sentences, paragraphs and sections (e.g., Salton et al., 1993; Wilkinson, 1994). *Semantic* passages are based on similarities of the subject or content of the text (e.g., Hearst and Plaunt, 1993; Mittendorf and Schauble, 1994). *Window* passages are based upon a fixed number of words (Callan, 1994). Almost all of the research related to passages involves retrieval experiments where passage-level representations are combined with global (whole document) representations. For example, Salton et al., 1993, refine rankings based on global similarity using sentence, paragraph and section similarities. Mittendorf and Schauble, 1994, combine a probabilistic model of relevant text passages with a model of text in general. Callan, 1994, combines global document evidence and window-based passage evidence in a probabilistic framework. Recent research by Kaszkiel and Zobel, 1997, confirms Callan's earlier result that fixed-length, overlapping passages produce the best effectiveness, and that the best window size is between 150-250 words. More than 20% improvements in average precision were obtained in some experiments.

[2] `http://www.google.com`

2.4 PHRASES AND PROPER NOUNS

Simple noun phrases are an extremely important part of the searcher's vocabulary. Many of the queries submitted to current Web search engines consist of 2-3 words (about 50% in Jansen et al., 1998, and many of those queries are phrases. Phrase representations of documents were used in the earliest bibliographic systems and were evaluated in early studies (Cleverdon, 1967). Salton and Lesk, 1968, reported retrieval experiments that incorporated *statistical* phrases based on word co-occurrence into the document representation as additional index terms. Fagan, 1989, also studied statistical phrases but ranked documents by a weighted average of the scores from a word representation and a phrase representation. In both of these studies, the effectiveness improvements obtained were mostly small but varied considerably depending on the document collections being used (from -1.8% to 20% improvement in average precision). Fagan's experiments with *syntactic* methods of recognizing phrases were less successful (Fagan, 1987). Croft et al., 1991, and Callan et al., 1995a, introduced a phrase model that explicitly represents phrasal or proximity representations as additional evidence in a probabilistic framework. This approach yielded results that were somewhat more effective than simple statistical phrases, but not consistently. Bartell et al., 1994, also demonstrated improvements from combining word-based and phrase-based searches.

Another representation that has been treated as additional index terms are the so-called named entities found using information extraction (MUC-6, 1995). These are special classes of proper nouns mentioned in document text such as people, companies, organizations, and locations. Callan and Croft, 1993, described how these entities could be incorporated into the retrieval process. For the queries used in this study, the impact on effectiveness was not significant.

2.5 MULTIMEDIA

Documents, in general, can be complex, multimedia objects. We have described some of the representations that can be derived from the text and links associated with the document. Other media such as speech, images, and video may also be used in queries and documents, and should be considered alternative representations. A number of people have described how multimedia objects can be retrieved using associated text, such as captions, surrounding text, or linked text (e.g., Croft and Turtle, 1992; Harmandas et al., 1997). This is the primary representation used by image searching engines on the Web. Increasingly, however, image and video search systems are making use of image processing techniques that help to categorize pictures (e.g., Frankel et al., 1996) or compare images directly (Flickner et al., 1995; Ravela and Manmatha, 1997). These image-based techniques typically use very different data structures and algorithms compared to text-based techniques. As a result, combining the evidence about relevance from text and image representations (and potentially other representations) involves combining the rankings from multiple subsystems. This has been a concern of both the IR community (e.g., Croft et al., 1990; Fuhr, 1990; Callan et al., 1995b) and the multimedia database community (e.g., Fagin, 1996), and will be discussed further in sections 4 through 6.

3 COMBINING QUERIES

We have described various representations of documents that could be used as evidence for relevance. Experiments with combinations of these representations show that, in general, using more than one representation improves retrieval effectiveness. They also show that when one source of evidence is weaker (less predictive of relevance) than the others, this must be reflected in the process of accumulating evidence or effectiveness will suffer. These observations are consistent with the simple probabilistic framework mentioned in the first section.

Estimating relevance, however, involves more than document representations. Queries, which are representations of the searcher's information need, are an important part of the process of calculating $P(R|D, Q)$. Each additional piece of evidence that the query contains about the true information need can make a substantial difference to the retrieval effectiveness. This has long been recognized and is the basis of techniques such as relevance feedback, where user judgments of relevance from an initial ranked list are used to modify the initial query (Salton and McGill, 1983), and query expansion, which involves the automatic addition of new terms to the query (Xu and Croft, 1996; Mitra et al., 1998). Relevance feedback and query expansion can also be viewed as techniques for creating alternative representations of the information need. Traditional query formulation tools, such as the thesaurus, can be viewed the same way. Even in the earliest retrieval experiments with a thesaurus (Salton and Lesk, 1968) and automatic query expansion using term clustering (Sparck Jones, 1971), the thesaurus classes or term clusters were treated as alternative representations that were combined with word-based queries.

As mentioned in the last section, in order to make use of some alternative document representations, the query must at least be partially described using the same representations. Salton et al., 1983, used relevance feedback to add citations to the initial query, and consequently was able to use citations in the document representations to improve the ranking. Crouch et al., 1990, also used feedback to add controlled vocabulary terms to the query. Xu and Croft, 1996, used an automatic query expansion technique to construct a phrase-based representation of the query that was combined with the initial word-based representation using weighted averaging. Callan et al., 1995a, describe a number of other strategies for automatic construction of alternative query representations.

The idea that there are alternative queries only makes sense with the assumption that there is an underlying information need associated with the searcher. A given query is a noisy and incomplete representation of that information need. By constructing multiple queries, we are able to capture more pieces of evidence about relevance. The best source of information about the information need is, of course, the searcher. A number of studies have looked at or observed the effect of capturing multiple queries from a single searcher or from multiple searchers given the same specification of an information need. McGill et al., 1979, carried out a study of factors affecting ranking algorithms, and noticed that there was surprisingly little overlap between the documents retrieved by different search intermediaries (people who are experts in the use of a particular search system) when they were assigned the same information need as a starting point. Saracevic and Kantor, 1988, also found that when different intermediaries constructed Boolean search formulations based on the same descriptions of the information need, there was little overlap in the retrieved sets. In addition, they observed that the odds of a document being judged relevant was proportional to the number of times it was in a retrieved set.

Based on these studies, Turtle and Croft, 1991, proposed a retrieval model that explicitly incorporated the notion of multiple representations of the information need. They report the results of experiments that combined word-based and Boolean queries to improve retrieval effectiveness. Rajashekar and Croft, 1995, extended this work by combining word-based queries with two other queries based on different types of manual indexing. Combining pairs of query representations produced consistent performance improvements, and a weighted combination of all three achieved the best retrieval effectiveness. Belkin et al., 1993, carried out a more systematic study of the effect of query combination in the same probabilistic framework. They verified that retrieval effectiveness could be substantially improved by query combination, but that the effectiveness of the combination depends on the effectiveness of the individual queries. In other words, queries that provided less evidence about relevance had to have lower weights in the combination to improve performance. Bad query representations could, in fact, reduce effectiveness when combined with better representations. In a subsequent, larger study, Belkin et al., 1995, obtained essentially the same results and compared query combination to the strategy of combining the output of different systems (which they called data fusion). The latter strategy is discussed in the next two sections.

4 COMBINING RANKING ALGORITHMS

The next two sections discuss techniques for combining the output of several ranking algorithms. The ranking algorithms can be implemented within the same general framework, such as probabilistic retrieval or the vector space approach, or they can be implemented in different frameworks. The ranking algorithms can also be operating on the same databases, overlapping databases, or totally disjoint databases. In this section, we focus on the combination of the output of ranking algorithms implemented in the same framework and operating on the same data.

Croft and Harper, 1979, noted that a cluster-based ranking algorithm retrieved different relevant documents than a word-based probabilistic algorithm, even though their average performance was very similar. They proposed using clustering as an alternate search strategy when word-based ranking failed. Similar observations were made about the performance of other ranking algorithms, particularly in the TREC evaluations (Harman, 1995). The approach of providing alternative ranking algorithms was incorporated into the design of some experimental retrieval systems, most notably I^3R (Croft and Thompson, 1987) and CODER (Fox and France, 1987). Attempts were made to select the best algorithm for a given query using an adaptive network (Croft and Thompson, 1984), and combine the results of multiple ranking algorithms using a plausible inference network (Croft et al., 1989). Turtle and Croft, 1991, showed how a nearest neighbor cluster search could be described and combined with a word-based search in a probabilistic framework.

As mentioned in section 1, combining the output of ranking algorithms can be modeled as combining the output of multiple classifiers (Tumer and Ghosh, 1999). A ranking algorithm defines a classifier for each query, where the classes are associated with relevance and non-relevance (Van Rijsbergen, 1979). These classifiers can be trained using relevance feedback, but typically the only information that is available about the relevant class comes from the query. In fact, the approach of combining multiple representations of the information need, discussed in the last section, is more properly viewed as constructing multiple classifiers (one for each query representation) and combining their output. From this point of view, experiments such as Rajashekar and Croft, 1995, and Belkin et al., 1995, can be viewed as validation of the effectiveness of combining multiple classifiers for IR.

Combining classifiers has been extensively studied in neural network research and the machine learning area in general. Tumer and Ghosh, 1999, provide a good overview of the literature in this area. They point out that given limited training data and a large, noisy "pattern space" (possible document descriptions), variations in weighting, initialization conditions, and the internal structure of the classifier produce different outputs. This is exactly what IR researchers have observed, even to the extent that variations in the "*tf.idf*" weighting functions[3] can retrieve substantially different documents (Lee, 1995).

Tumer and Ghosh observe that simply averaging the output of the classifiers is the most common combining strategy, although more complex strategies such as learning appropriate weighted averages have been evaluated. They analyze the averaging strategy and show that for unbiased, independent classifiers, the "added error" above the Bayes error will be reduced by a factor of N for N classifiers. They model the output of a classifier for a given input (a document) as a combination of the probability distribution for each class and a noise distribution (the error). Reducing the error corresponds to reducing the variance of the noise. Since classification errors correspond to not retrieving relevant documents or retrieving non-relevant documents, reducing this error will improve retrieval effectiveness (Van Rijsbergen, 1979). Tumer and Ghosh also mention that the simple combining strategies are best suited for situations where the classifiers all perform the same task (which is the case for IR), and have comparable success. Simple combination strategies can fail when even one of the classifiers being combined has very poor performance or very uneven performance.

There is some evidence that simple combination strategies such as summing, averaging or weighted averaging may be adequate for IR. For example, most of the experiments described in sections 2 and 3 used these strategies. Weighted averaging was required in cases where one of the classifiers were based on a poor document description (controlled vocabulary terms). Bartell et al., 1994, describe an approach to learning a weighted, linear combination of classifiers for IR. Fox and Shaw, 1994, conducted an evaluation of combination strategies using different retrieval algorithms in a vector space model. They found the best combination strategy consisted of summing the outputs of the retrieval algorithms, which is equivalent to averaging in terms of the final ranking. Hull et al., 1996, compared simple and complex combinations of classifiers for the document filtering problem, which has substantially more training data than is the case for IR. They found that the best improvement in performance came from the simple averaging strategy.

IR classifiers will often not be independent, since they typically use the same document and query representations. Some of the best results in terms of improving retrieval effectiveness have come from combining classifiers based on very different representations, such as the citation experiments described in section 2. Combining classifiers that are very similar, such as those based on minor differences in the *tf.idf* weights, usually does not improve perfor-

[3] The *tf.idf* weight is a combination of weights derived from within-document term frequency (*tf*) and the inverse of the number of documents in the database that contain the term (*idf*). There are many variations of this weight discussed in the literature (Salton and Buckley, 1988).

mance. Lee, 1995, conducted an extensive study of combining retrieval output based on different weighting schemes in the vector space model. He combined these outputs by averaging the normalized scores. His experiments showed that combining classifiers ("retrieval runs" in his paper) based on similar weighting schemes had little impact on performance. Combining classifiers based on substantially different weighting schemes, however, produced significant improvements. Specifically, he found that combining rankings based on cosine normalization of the *tf.idf* weight with rankings based on other normalization schemes was effective (about 15% improvement in average precision). Hull et al., 1996, found that the gains in performance from combination were limited by the correlation between the classifiers they used. The correlation was caused primarily by using the same training data and was strongest for classifiers that used the same document representations.

The discussion of classifier combination in Tumer and Ghosh assumes that the classifiers have comparable output in that they are trying to make the same decision within the same framework. For probabilistic systems, this means they are all attempting to estimate $P(R|D,Q)$ and we can combine these estimates using simple strategies. The lack of knowledge of prior probabilities and the lack of training data, however, make the accurate estimation of these probabilities difficult and can make the outputs of the classifiers less compatible. Combining a cluster-based retrieval algorithm with a word-based algorithm, for example, can be quite difficult because the numbers produced by these algorithms for ranking may have little relationship to the probabilities of relevance. With sufficient training data, the relationship between these numbers and the probabilities can be learned but this data is usually not available. Incompatibility of classifier output also occurs in the vector space model, as discussed in Lee, 1995. In that paper, the scores produced by different retrieval runs were normalized by the maximum scores for each run in order to improve compatibility. This problem is particularly acute for an approach that combines the output of completely different search systems, with little idea of how the numbers output by those systems are calculated. This situation is discussed in the next section.

5 COMBINING SEARCH SYSTEMS

The idea of combining the output of different search systems was introduced during the DARPA TIPSTER project (Harman, 1992) and the associated TREC evaluations (Harman, 1995). These evaluations involve many search systems running the same queries on the same, large databases. The results of these searches are made available for research and a number of studies have been done of combination strategies. Belkin et al., 1995, combined the results of searches from a probabilistic system and a vector space system and showed performance improvements. Lee, 1997, combined the results from six selected retrieval systems from a TREC evaluation. He investigated the combination strategies used in Fox and Shaw, 1994, and Lee, 1995. Scores from the different retrieval systems were normalized using the maximum and minimum scores according to the formula:

$$normalized\_score = \frac{unnormalized\_score - min\_score}{max\_score - min\_score}$$

Lee's results showed that the most effective combinations (up to approximately 30% improvement relative to a single search) were between systems that retrieve similar sets of relevant documents, but different sets of nonrelevant documents. This is related to the observation in Lee, 1995, that retrieval algorithms with low overlap in the retrieved sets will, given similar overall performance, produce the best results in combination. Vogt and Cottrell, 1998, in a study of the factors that predict good combination performance, looked at pairwise combinations of all systems (61 of them) from another TREC evaluation. They were able to verify Lee's observation that the best combinations were between systems that retrieve similar sets of relevant documents and dissimilar sets of nonrelevant documents.

These results can be simply explained in terms of uncorrelated classifiers. The sets of documents that are being compared in these studies are the top 1000 documents retrieved by each system for each query. The number of relevant documents for a given query is typically not large (100-200). We would expect, therefore, that many of the relevant documents are retrieved by most systems. This is in fact shown by Lee's analysis of the correlation between the relevant retrieved document sets. Since there are large numbers of nonrelevant documents, we would expect that uncorrelated classifiers (search systems) would retrieve different sets of nonrelevant documents and this is what was observed. We would also expect that uncorrelated systems would produce different rankings of the relevant documents, even when the overlap in the sets of retrieved relevant documents is high. Vogt and Cottrell observed this difference in rankings for good combinations. They also observed, as did Lee, that the best combinations occur when both systems being combined have good performance, although it is possible to get improvement when only one of the systems has good performance. All of these observations are consistent with the statement that the combination with the lowest error occurs when the classifiers are independent and accurate.

Lee, 1997, presented two other results related to combination strategies for different search systems. The first of these results was that combining the outputs of search systems using the ranks rather than the scores of the documents was, in general, not as effective. The exception to that was when the search systems had very different characteristics in terms of the shape of the score-rank curve. This can be interpreted as evidence that the normalized score is usually a better estimator for the probability of relevance than the rank. Using the ranks is a more drastic form of smoothing that appears to increase error except when the systems being combined have very different scoring characteristics.

The second of Lee's results was that the best combination strategy was to sum the normalized scores and then multiply by the number of nonzero scores in the combination. This was better than simply summing the scores by a small but consistent margin. This form of combination heavily favors the documents retrieved by more than one system. A zero score for a document means that it was not retrieved in the top 1000 for that system rather than being a true estimate of the probability of relevance. Given that, the combined estimate of the probability for such a document is likely to have a much higher error than the estimates for documents which have only non-zero scores. A combination strategy that favors these documents could be interpreted, then, as favoring estimates with lower error.

The experiments mentioned previously combined the outputs of multiple search systems using the same database. The outputs of search systems using overlapping or disjoint databases could also be combined. This type of combination has been called *collection fusion, distributed IR*, or *meta-search*. Voorhees et al., 1995, report experiments on techniques for learning weights to associate with each of the systems in the combination. The weights are used with the ranked document sets to determine how the documents will be mixed for the final ranking. This approach has some similarity to the rank combination strategies used by Lee, 1997. Callan et al., 1995b, show that using scores weighted by an estimate of the value of the database for the query is substantially better than interleaving ranks. Both Voorhees and Callan used disjoint databases for their experiments. In many practical environments, such as meta-search on the web, the databases used by the search systems will be overlapping. This will result in a situation similar to that described by Lee, 1997, where documents would have a varying number of scores associated with them. Although there are no thorough evaluations of the combination of web search results, it appears that Lee's results may apply in this situation. This means that the best combination strategy may be to normalize the scores from each search engine, sum the normalized scores for each document, and multiply the sum by the number of search engines that returned that document (at a given cutoff). If the overlap between the databases used by the search engines is low (i.e. there are substantial differences in the amount of the web indexed), the last step would be less effective.

The situation of combining the outputs of multiple search systems also applies to multimedia retrieval (Croft et al., 1990; Fagin, 1996; Fagin, 1998). In this case, we are typically combining the output of a text search and the output of one or more ranking algorithms that compare image features such as color distributions or texture. The experimental results discussed above suggest that the scores from these image and text retrieval algorithms should be combined by normalizing and then summing, potentially taking into account the number of non-zero scores. There is, unfortunately, no current evidence that this is a good choice other than the theoretical argument about classifier combination. Fagin, 1996, develops an algorithm for combining scores in a multimedia database using the standard operators of fuzzy logic, namely *min* and *max*. Lee's experiments do provide evidence that these combination operators perform significantly worse than summing. Ciaccia et al., 1998, also discuss ranking in a multimedia database environment. They are concerned primarily with the efficiency of combination, as is Fagin, and present performance results for a range of combination operators.

6 COMBINING BELIEF

The previous sections have described the results of many different experiments with combining evidence to improve retrieval effectiveness. We have described these experiments in terms of combining evidence about relevance in a single classifier and then combining the outputs of multiple classifiers. Figure 1.1 (derived from Tumer and Ghosh, 1999) shows this overall conceptual view of combination. In this view, multiple representations (called *feature sets* in the classification literature) are constructed from the raw data in the documents. Both retrieval algorithms and search systems are regarded as classifiers that make use of these representations to calculate the probability of relevance of the documents. Some algorithms and systems combine representations in order to reduce the error of that calculation. The output of the retrieval algorithms or search systems can then be combined to further reduce the error and improve retrieval effectiveness.

This simple framework can be used to explain some of the basic results obtained with combination experiments, such as the increased probability of relevance for documents retrieved multiple times by alternative representations and search systems, and the low overlap between the outputs of the best combined systems. According to this view of combination, there are only two requirements for minimizing the classification error and obtaining the best retrieval performance. The first of these is that each individual classifier (retrieval algorithm or system) should be as accurate as possible. This means that each classifier should produce probabilities of relevance with low error. The second requirement is that the classifiers that are combined should be uncorrelated. This means that we do not want to combine classifiers that repeatedly produce the same or similar rankings for documents, regardless of whether those rankings are accurate or inaccurate. Classifiers that use different representations and retrieval algorithms are more likely to be independent.

A number of other frameworks have been proposed in the IR literature for providing a formal basis for the combination processes described in Figure 1.1.

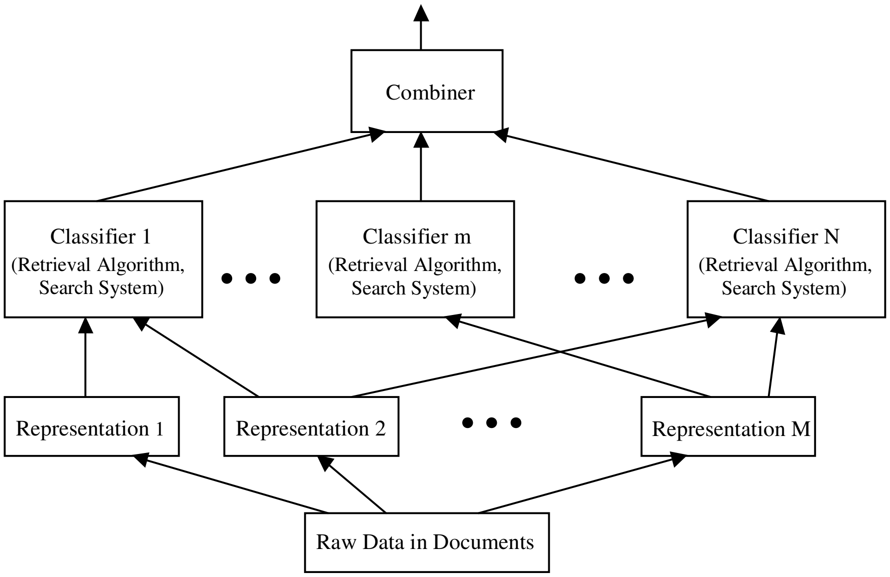

Figure 1.1 Combining strategies for retrieval

Some frameworks address the combination of representations for retrieval algorithms, some the combination of retrieval algorithms, and others the combination of search system output. We will discuss the frameworks using these categories.

6.1 FRAMEWORKS FOR COMBINING REPRESENTATIONS

The vector space model has been used as the basis for a number of combination experiments. In this model, documents and queries are characterized by vectors of weighted terms. Fox et al., 1988, proposed using subvectors to describe different "concept types" or representations derived from documents. An overall similarity between a document and a query, which is used to rank the documents, is computed as a linear combination of the similarities for each subvector. For example, if documents were represented using words, authors, and citations, the similarity function would be :

$$\begin{aligned} sim(Q,D) &= c_{word} \cdot sim(Q_{word}, D_{word}) + \\ &\quad c_{author} \cdot sim(Q_{author}, D_{author}) + c_{cite} \cdot sim(Q_{cite}, D_{cite}) \end{aligned}$$

where the c_i values are coefficients. This type of weighted linear combination is identical to that used in the experiments described in previous sections for combining the output of classifiers or search systems. This shows that there is little difference in this framework between combining representations and combining search output. Fox et al., 1988, used the similarity function to predict relevance and then performed a regression analysis with test collection data to determine the values of the coefficients.

Fuhr and Buckley, 1991, also used regression to learn effective combinations of document representations. This work was based on a probabilistic model that estimates $P(R|x(t,d))$, which is the probability of relevance given a "relevance description" x of the term and document characteristics, instead of $P(R|t,d)$. By using representations based on these characteristics rather than directly on words, more training data is available to estimate the probabilities in the model. They used a least-squared error criterion to compute coefficients of polynomial combinations of the term and document characteristics. Their results showed that a linear combination produced the best overall performance. The characteristics that were used in the relevance description included within-document frequency of terms, the maximum frequency of a term in a document, the number of documents in which a term occurs, the number of documents in the collection, the number of terms in a document, and whether a term occurs in the title or the body of a document.

Gey, 1994, developed a *logistic inference model* that used logistic regression, which is generally considered more appropriate for estimating probabilities, to compute the coefficients of a formula for the log of the odds of relevance given the presence of a term. The formula was a linear combination of term characteristics in the documents and the queries.

Greiff, 1998, described a probabilistic model developed using exploratory data analysis, which involves looking at large amounts of data about terms, documents and relevance to discover relationships. This approach has some similarity to the regression models described above, but does not make assumptions about the underlying distributions. In Greiff, 1999, he extends his approach to incorporate multiple sources of evidence (representations) based on the Maximum Entropy Principle, which is a way of determining an appropriate probabilistic model given known constraints. The model that results from this approach scores documents using a linear combination of a within-document frequency (*tf*) component and an inverse document frequency component (*idf*) for each term that matches the query. Although this is a relatively simple model in terms of the number of representations being combined, the framework he develops is sufficiently general to incorporate any of the representations mentioned previously. Incorporating a new representation involves studying retrieval data involving that representation, and using regression to determine coefficients of a formula that predicts relevance given the evidence provided by the new representation, conditioned on the evidence provided by the existing representations. This new formula can then be simply added to the linear combination of formulas for other representations.

The common characteristic for all frameworks that use training data and regression, of which Greiff's can be viewed as the most general, is that a retrieval algorithm or search system based on them will produce estimates of probabilities of relevance instead of just normalized similarity values. Other probabilistic systems, such as INQUERY (Callan et al., 1995a), assume that because the goal of the system is to rank documents, the parts of a probabilistic formula that are constant for a given query (such as prior probabilities) can be ignored. In addition, ad-hoc (but effective) formulas are used to calculate parts of the document scores. This means that the numbers produced by these systems are not probabilities. This is a significant disadvantage when it comes to combining the output of a system with the output of other systems. Systems with compatible outputs, and accurate probability estimates, will produce the best combinations assuming they are independent.

6.2 FRAMEWORKS FOR COMBINING RETRIEVAL ALGORITHMS

The inference network framework, developed by Turtle and Croft (Turtle and Croft, 1991; Turtle and Croft, 1992) and implemented as the INQUERY system (Callan et al., 1995a), was explicitly designed for combining multiple representations and retrieval algorithms into an overall estimate of the probability of relevance. This framework uses a Bayesian network (Pearl, 1988) to represent the propositions and dependencies in the probabilistic model (Figure 1.2). The network is divided into two parts: the document network and the query network. The nodes in the document network represent propositions about the observation of documents (D nodes), the contents of documents (T nodes), and representations of the contents (K nodes). Nodes in the query network represent propositions about the representations of queries (K nodes and Q nodes) and satisfaction of the information need (I node). This network model corresponds closely to the overview of combining classifiers in Figure 1.1. The parts of the network that model the raw data in documents, the features extracted from that data, the classifiers that use the features to predict relevance, and the overall combiner for the classifier outputs are labeled in Figure 1.2.

In this model, all nodes represent propositions that are binary variables with the values *true* or *false*, and the probability of these states for a node is determined by the states of the parent nodes. For node A, the probability that A is *true* is given by:

$$p(A) = \sum_{S \subseteq \{1,\ldots,n\}} \alpha_S \prod_{i \in S} p_i \prod_{i \notin S} (1 - p_i)$$

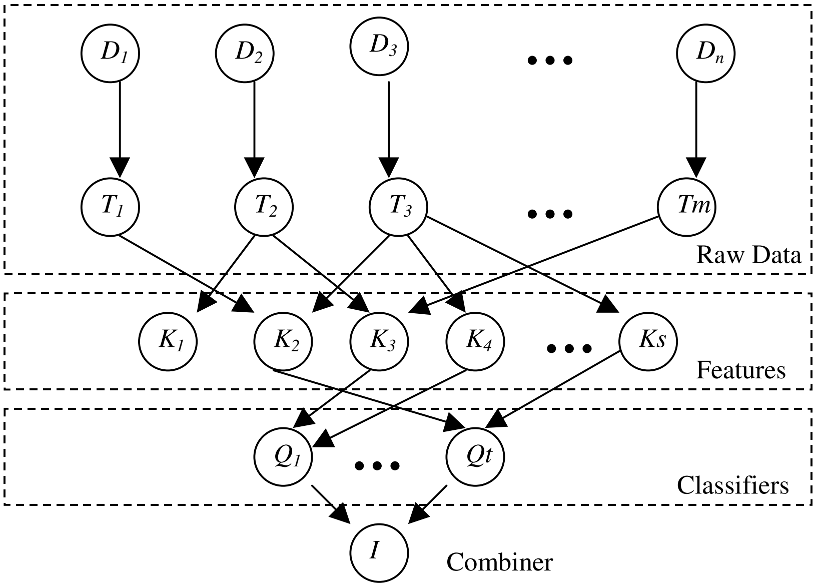

Figure 1.2 Bayesian net model of information retrieval

where α_S is a coefficient associated with a particular subset S of the n parent nodes having the state *true*, and p_i is the probability of parent i having the state *true*. Some coefficient settings result in very simple but effective combinations of the evidence from parent nodes. For example, if $\alpha_S = 0$ unless all parents have the state *true*, this corresponds to a Boolean *and*. In this case, $p(A) = \prod_{i=1}^{n} p_i$.

The most commonly used combination formulas in this framework are the average and the weighted average of the parent probabilities. These formulas are the same as those shown in other research to be the best combination strategies for classifiers and discussed earlier in the paper. The combination formula based on the average of the parent probabilities comes from a coefficient setting where the probability of A being *true* depends only on the number of parent nodes having the state *true*. The weighted average comes from a setting where the probability of A depends on the specific parents that are *true*. Parents with higher weight have more influence on the state of A. The INQUERY search system provides a number of these "canonical" combination formulas as query operators. The three described above are *#and, #sum,* and *#wsum*.

In the INQUERY system, different document representations are combined by constructing nodes corresponding to propositions about each representation (i.e. is this document represented by a particular term from a representation vocabulary) and constructing queries using those representation nodes. The queries for each representation are combined using operators such as *#wsum* (Rajashekar and Croft, 1995). For example, there may be nodes corresponding to word-based terms and nodes corresponding to controlled vocabulary terms. These nodes are connected to documents by the probabilities that those terms represent the contents of the documents. Query operators are then used to construct a query based on words (q_{word}) and a query based on controlled vocabulary ($q_{control}$). These queries may be complex combinations of the evidence in those terms. Each of the queries is, in fact, a classifier that could produce an individual ranking of the documents. The two representations are combined by combining the query nodes. If the searcher wanted to weight the word-based representation twice as much as the controlled vocabulary representation, the final INQUERY query would be $\#wsum(2.0\ q_{word}\ 1.0\ q_{control})$.

This example shows that the inference net framework can be used for most of the combination processes in Figure 1.1. Individual classifiers are built using representation nodes and combination operators, and the output of those classifiers is combined using other operators. It is possible to represent different retrieval algorithms in the inference net framework by using different combinations of representation nodes (i.e. new operators), new types of representation nodes, and different techniques for computing initial probabilities for representation nodes. The probabilities associated with the query node propositions are computed from the probabilities associated with representation nodes. The probabilities associated with representation nodes, however, can be computed from evidence in the raw data of the documents. For example, a *tf.idf* formula is used in INQUERY to compute the probability of a word-based representation node for a particular document. Turtle and Croft, 1991, describe how retrieval based on document clustering and hypertext links can be incorporated into the inference net framework by changing the probability estimates for nodes representing linked documents.

The advantage of the inference net is that it provides a probabilistic framework for rapidly constructing new classifiers based on different representations and retrieval algorithms and combining their output. Not every retrieval algorithm can be represented in this framework, however. Greiff et al., 1997, describes the class of combination operators that can be computed easily and how these can be used to model a well-known vector space ranking algorithm.

Although this effort was successful in achieving comparable or better effectiveness, it shows the difficulty of modeling even relatively simple retrieval algorithms that are not based on a probabilistic approach. A complex retrieval algorithm based on, for example, a neural net architecture, could not be modeled in an inference net.

Another problem with the inference net is that the output of the classifiers do not correspond well to real probabilities. This is due to the heuristic estimation formulas used (such as *tf.idf*), the lack of knowledge of prior probabilities, and the lack of training data. Haines and Croft, 1993, describe how relevance feedback can be used to modify the query network to produce more effective rankings, although their approach does not improve the correspondence between document scores and probabilities. Haines, 1996, discusses how the structure of the inference net could be changed to better accommodate learning, but his approach was difficult to implement in the INQUERY system. A similar comment applies to the general techniques for learning with Bayesian networks (Heckerman et al., 1994).

6.3 FRAMEWORKS FOR COMBINING SEARCH SYSTEM OUTPUT

The strategies for combining the output of search systems described in Lee, 1997, are implemented in a simple, heuristic framework. The output of the systems are treated as similarity values that are normalized and combined using one of a variety of possible strategies. The specific normalization and combination strategies are selected based solely on empirical evidence, although some of the combination strategies, such as *min* and *max*, can be justified with formal arguments.

Hull et al., 1996, also experimented with various combination strategies, but in the context of a framework of combining classifier output. They derive a formula for the combination of the output of n classifiers $C_1, \ldots, C_n$ that are conditionally independent given relevance (R) and non relevance:

$$\log O(R|C_1, \ldots, C_n) = \sum_{i=1}^{n} \log O(R|C_i) - (n-1) \cdot \log O(R)$$

This formula is then used as justification for the strategy of summing the log-odds numbers derived from the classifier output. There are two problems with this framework. The first is that the conditional independence assumption is not warranted. The second is that $P(R|C_i)$ is not the same as the output of the classifier. The explanation for this is best done in a Bayesian network framework where the C_i are nodes representing the *decisions* of the document classifiers and R is a node representing the *combined decision*. The structure of this network is shown in Figure 1.3. Each classifier can be viewed as voting on the overall decision of the combined network. The output of classifier C_i for a given document is the probability associated with the state $C_i = true$. For this network,

$$P(R) = \sum P(R|C_1, \ldots, C_n)P(C_1, \ldots, C_n)$$

where $C_1, \ldots, C_n$ represents a particular configuration of nodes with values *true* and *false*, and the summation is over all possible configurations. In the Bayesian network framework, the evidence supporting each classifier's decision is assumed to be independent and

$$P(C_1, \ldots, C_n) = \prod_{i=1}^{n} P(C_i)$$

This is simply a reformulation of the function for node probability given in section 6.2. As mentioned previously, there are a number of possibilities for calculating $P(R|C_1, \ldots, C_n)$. If the probability of voting for relevance overall depends on the number of classifiers that vote for relevance, the resulting combining function is the average of the classifier probabilities. We could base the overall vote for relevance on the vote of the classifier with the maximum (or minimum) probability of voting for relevance. This would result in the combining function being *max* (or *min*). A number of other combining functions are possible, but they have the common characteristic that determining the probability of relevance involves looking at the collective vote of the classifiers. The overall vote and the probability of that vote cannot be determined by looking at each classifier's vote independently of the others. Thus the assumption of conditional independence is not appropriate.

Hull et al., 1996, also report the interesting result that the probability estimates obtained from the combined classifier were not as accurate as the best individual classifier, even though the combined rankings were significantly better. This is related to the problem of combining similarity scores mentioned by Lee, 1997. Referring to Figure 1.3, each of the classifiers is, in effect, voting on relevance. The probabilities associated with a particular classifier may be inaccurate relative to the true probability of relevance, but still be consistent with respect to ranking the "votes" for that classifier. This would produce an effective ranking overall, but inaccurate probabilities. Indeed, if one of the classifiers were producing very accurate probability estimates, the combined estimates would be worse. The best situation is, of course, when all classifiers are producing reasonably accurate estimates. In that case, the combined estimate, as well as the combined vote, should be more accurate (Tumer and Ghosh, 1999).

Fagin (Fagin, 1996; Fagin, 1998) proposed a framework based on fuzzy logic for combining the output of multiple search systems in a multimedia

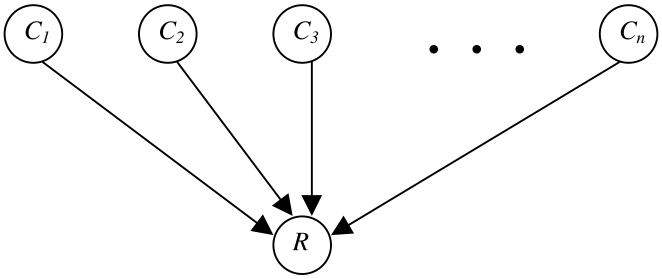

Figure 1.3 Combining the output of classifiers in a Bayesian net

database system. In this framework, a query is a Boolean combination of "atomic queries". Each atomic query is of the form $object\ attribute = value$ and the result of an atomic query is a "graded set" or a list of objects with their scores. Fagin combines the results of atomic queries using the *min* and *max* combining functions because they are unique in preserving logical equivalence of queries involving conjunction and disjunction. This property is important for the query optimization strategies he develops, but as we have discussed, *min* and *max* do not produce effective retrieval compared to averaging the output of the search systems. Query efficiency will continue to increase in importance, however, as multimedia systems are scaled up to accommodate the enormous volume of data being generated. The assumptions that have been used to build large distributed text retrieval systems may also be different in a multimedia environment, requiring changes in the query processing strategies. For these reasons, it is important to consider efficiency when implementing a combining strategy.

The inference network model has also been proposed as a framework for multimedia retrieval (Croft and Turtle, 1992). One of the main problems with incorporating the results of an image retrieval algorithm into a probabilistic combination framework is that the image techniques typically rank images based on distance or similarity scores that are not probabilities. As mentioned in section 5, the scores can be normalized, but this is a completely ad-hoc procedure that does not produce accurate probability estimates. There has recently been work done on object recognition using probabilistic models (Schneiderman and

Kanade, 1998). They develop a formula for $P(object|image)$, the probability of an object (such as a face) being present in an image. In an image retrieval setting, a query typically includes an image or part of an image and possibly some text. The task of the image retrieval component of the system is to find images that are "similar" to the query image. This task can be based on a probabilistic model, as in Schneiderman and Kanade, 1998. One such probabilistic model would be to compute $P(target\ image|database\ image)$. This probability could be calculated using a representation based on visual index terms, similar to the text models described above. Alternatively, a probability of generating the target image could be calculated. This approach corresponds to the use of language models for retrieval, as described in the next section.

7 LANGUAGE MODELS

Most probabilistic retrieval models attempt to describe the relationship between documents and index terms by estimating the probability that an index term is "correct" for that document (Fuhr, 1992). This is a difficult probability to explain, and as a result, heuristic *tf.idf* weights are used in the retrieval algorithms based on these models. In order to avoid these weights and the awkwardness of modeling the correctness of indexing, Ponte and Croft, 1998, proposed a *language modeling* approach to information retrieval. The phrase "language model" is used by the speech recognition community to refer to a probability distribution that captures the statistical regularities of the generation of language (Jelinek, 1997). Generally speaking, language models for speech attempt to predict the probability of the next word in an ordered sequence. For the purposes of document retrieval, Ponte and Croft modeled occurrences at the document level without regard to sequential effects, although Ponte, 1998, showed that it is possible to model local predictive effects for features such as phrases. Mittendorf and Schauble, 1994, used a similar approach to construct a generative model for retrieval based on document passages.

The approach to retrieval described in Ponte and Croft, 1998, is to infer a language model for each document and to estimate the probability of generating the query according to each of these models. Documents are then ranked according to these probabilities. In this approach, collection statistics such as term frequency, document length and document frequency are integral parts of the language model and do not have to be included in an ad hoc manner. The score for a document in the simple unigram model used in Ponte and Croft is given by:

$$P(Q|D) = \prod_{w \in Q} P(w|D) \prod_{w \notin Q} (1 - P(w|D)$$

where $P(Q|D)$ is the estimate of the probability that a query can be generated for a particular document, and $P(w|D)$ is the probability of generating a word given a particular document (the language model).

Much of the power of this simple model comes from the estimation techniques used for these probabilities, which combine both maximum likelihood estimates and background models. This part of the model benefits directly from the extensive research done on estimation of language models in fields such as speech recognition and machine translation (Manning and Schutze, 1999). More sophisticated models that make use of bigram and even trigram probabilities are described in Ponte, 1998, and are currently being investigated (Miller et al., 1999; Song and Croft, 1999). The Ponte and Croft model uses a relatively simple definition of relevance that is based on the probability of generating a query text. This definition does not easily describe some of the more complex phenomena involved with information retrieval. The language model approach can, however, be extended to incorporate more general notions of relevance. Berger and Lafferty, 1999, show how a language modeling approach based on machine translation provides a basis for handling synonymy and polysemy. Hofmann, 1999, describes how mixture models based on *latent classes* can represent documents and queries. The latent classes can be thought of as language models for important topics in a domain. The language model approach can also be integrated with the inference net model, as described later.

For this paper, the important issue is how the language model frameworks for retrieval deal with combination of evidence. In fact, it is ideally suited to the combination approach. The language model framework can readily incorporate new representations, it produces accurate probability estimates, and it can be incorporated into the general Bayesian net framework. Miller et al., 1999, point out that estimating the probability of query generation involves a mixture model that combines a variety of word generation mechanisms. They describe this combination using a Hidden Markov Model with states that represent a unigram language model ($P(w|D)$), a bigram language model ($P(w_n|w_{n-1}, D)$), and a model of general English ($P(w|English)$), and mentions other generation processes such as a synonym model and a topic model. Hofmann, 1999, and Berger and Lafferty, 1999 also describe the generation process using mixture models, but with different approaches to representation. Put simply, incorporating a new representation into the language model approach to retrieval involves estimating the language model (probability distribution) for the features of that representation and incorporating that new model into the overall mixture model. The standard technique for calculating the parameters of the mixture model is the EM (Expectation-Maximization) algorithm (McLachlan and Krishnan, 1997). This algorithm, like the regression techniques mentioned earlier, can be applied to training data that is pooled across queries and this, together with techniques for smoothing the maximum likelihood estimates, results in more accurate probability estimates than a system using *tf.idf* weights without training, such as INQUERY.

Combining representations to produce accurate classifier output is only part of the overall combination process. The need to combine the outputs of multiple classifiers still exists. The other classifiers might be based on alternate language modeling approaches or completely different retrieval models, as we have described in the last section. To accomplish this level of combination, the language modeling approach can be incorporated into the inference network framework described in section 6.2. Figure 1.4 shows the unigram language model approach represented using a simplified part of the network from Figure 1.2. The W nodes that represent the generation of words by the document language model replace the K nodes representing index terms describing the content of a document. The Q node represents the satisfaction of a particular query. In other words, the inference net computes the value of $P(Q\ is\ true)$. In the Ponte and Croft model, the query is simply a list of words. In that model, Q is *true* when the parent nodes representing words present in the query are *true* and the words not in the query are *false*. The document language model gives the probabilities of the *true* and *false* states for the W nodes.

More generally, however, we can regard the query as having an underlying language model, similar to documents. This language model is associated with the information need of the searcher and can be described by $P(W_1, \ldots, W_n|Q)$. This probability is directly related (by Bayes rule) to the probability $P(Q|W_1, \ldots, W_n)$ that is computed by the inference network. More complex query formulations, such as those used in the INQUERY system, and relevance feedback provide more information about the searcher's underlying language model. This information can be directly incorporated into the inference network version of the language model approach by adding more links between the Q node and the W nodes, and changing how the evidence from the W nodes is combined at the Q node. For example, if we learn from relevance feedback that W_2 is an important word for describing the user's language model, we can assign more weight to this word in calculating $P(Q|W_1, \ldots, W_n)$.

The inference network, therefore, provides a mechanism for comparing the document language model to the searcher's language model. The two language models could also be compared using the Kullback-Leibler divergence or a similar measure (Manning and Schutze, 1999). The advantages of using the inference net mechanism are that it provides a simple method of using the relatively limited information that is known about the searcher's language model in a typical retrieval environment and it allows the language model approach to be directly combined with other classifiers described in this framework.

In sections 6.2 and 6.3, we described how the inference net model could be used to represent the combination of different retrieval algorithms and search systems. This framework can now be extended to include the language model

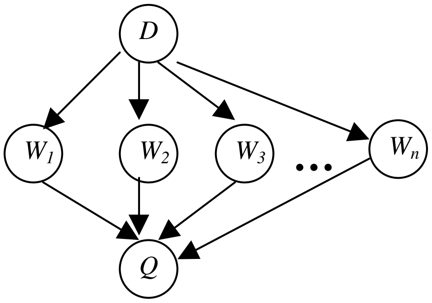

Figure 1.4 The language model approach represented in a Bayesian net

approach. The inference net incorporating language models will provide more accurate probability estimates than the inference net based on *tf.idf* weights. Language models also provide another view of the query formulation process that supports a more direct use of learning than was done in the INQUERY system.

8 CONCLUSION

It is clear from this survey of the experimental results published over the last twenty years that combination is a strategy that works for IR. Combining representations, retrieval algorithms, queries, and search systems produces, most of the time, better effectiveness than a single system. Sometimes the performance improvement is substantial. This approach to IR can be modeled as combining the output of classifiers. Given some assumptions, this model specifies that the best results will be achieved when the classifiers produce good probability estimates and are independent. Even when they are not independent, some improvement can still be expected from combination (Tumer and Ghosh, 1999). This simple prescription for good performance explains many of the results obtained from previous research, including those involving ad-hoc normalization and combination strategies for rankings based on similarity values.

The inference net framework was an attempt to provide a general mechanism for combination in a single search system. This framework is, in fact, an instantiation of the combination of classifiers model. Although it is a probabilistic model, the difficulty of estimating indexing probabilities led to ad-hoc *tf.idf* weights being used in the INQUERY implementation of the model. This, and the lack of training data, meant that the output of the system was not probabilities. The language model approach to retrieval can more easily be trained to produce accurate probabilities and can be integrated into the inference net framework. Other probabilistic approaches such as logistic regression or maximum entropy models could also be integrated into this framework.

Do we need to combine multiple approaches to retrieval and multiple search systems? Combination is, after all, expensive in terms of time, space and implementation effort. The combination of classifiers model implies that, given the extremely high-dimensional, noisy data contained in documents and the general lack of training data, many different classifiers for the retrieval problem could be built that are to some degree independent and would produce different, but equally effective, rankings. This means that combination is both inevitable and beneficial. Given that combination will need to be done, we should try to build search systems that use multiple representations to produce accurate output and we should provide a framework for those systems to be combined effectively. The inference net incorporating language models is a candidate for this framework. Providing such a framework will also be an important part of providing scalability for the immense information stores of the future.

Acknowledgments

This material is based on work supported in part by the National Science Foundation under cooperative agreement EEC-9209623. It is also supported in part by the United States Patent and Trademark Office and the Defense Advanced Research Projects Agency/ITO under ARPA order D468, issued by ESC/AXS contract number F19628-95-C-0235, and by SPAWARSYSCEN contract N66001-99-1-8912. Any opinions, findings and conclusions or recommendations expressed in this material are the authors and do not necessarily reflect those of the sponsors.

References

Bartell, B., Cottrell, G., and Belew, R. (1994). Automatic combination of multiple ranked retrieval systems. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 173–181.

Belkin, N., Cool, C., Croft, W., and Callan, J. (1993). The effect of multiple query representations on information retrieval system performance. In *Proceedings of the 16th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 339–346.

Belkin, N., Kantor, P., Fox, E., and Shaw, J. (1995). Combining the evidence of multiple query representations for information retrieval. *Information Processing and Management*, 31(3):431–448.

Berger, A. and Lafferty, J. (1999). Information retrieval as statistical translation. In *Proceedings of the 22nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–229.

Callan, J. (1994). Passage-level evidence in document retrieval. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 302–310.

Callan, J. and Croft, W. (1993). An evaluation of query processing strategies using the TIPSTER collection. In *Proceedings of the 16th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 347–355.

Callan, J., Croft, W., and Broglio, J. (1995a). TREC and TIPSTER experiments with INQUERY. *Information Processing and Management*, 31(3):327–343.

Callan, J., Lu, Z., and Croft, W. (1995b). Searching distributed collections with inference networks. In *Proceedings of the 18th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–28.

Ciaccia, P., Patella, M., and Zezula, P. (1998). Processing complex similarity queries with distance-based access methods. In *Proceedings of the 6th International Conference on Extending Database Technology (EDBT)*, pages 9–23. Springer-Verlag.

Cleverdon, C. (1967). The Cranfield tests on index language devices. *Aslib Proceedings*, 19:173–192.

Croft, W. and Harper, D. (1979). Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*, 35:285–295.

Croft, W., Krovetz, R., and Turtle, H. (1990). Interactive retrieval of complex documents. *Information Processing and Management*, 26(5):593–613.

Croft, W., Lucia, T.J.and Cringean, J., and Willett, P. (1989). Retrieving documents by plausible inference: An experimental study. *Information Processing and Management*, 25(6):599–614.

Croft, W. and Thompson, R. (1984). The use of adaptive mechanisms for selection of search strategies in document retrieval systems. In *Proceedings of the 7th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 95–110. Cambridge University Press.

Croft, W. and Thompson, R. (1987). I^{3}R: A new approach to the design of document retrieval systems. *Journal of the American Society for Information Science*, 38(6):389–404.

Croft, W. and Turtle, H. (1989). A retrieval model incorporating hypertext links. In *Proceedings of ACM Hypertext Conference*, pages 213–224.

Croft, W. and Turtle, H. (1992). Retrieval of complex objects. In *Proceedings of the 3rd International Conference on Extending Database Technology (EDBT)*, pages 217–229. Springer-Verlag.

Croft, W., Turtle, H., and Lewis, D. (1991). The use of phrases and structured queries in information retrieval. In *Proceedings of the 14th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 32–45.

Crouch, C., Crouch, D., and Nareddy, K. (1990). The automatic generation of extended queries. In *Proceedings of the 13th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 369–383.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407.

Fagan, J. (1987). *Experiments in automatic phrase indexing for document retrieval: A comparison of syntactic and non-syntactic methods*. PhD thesis, Computer Science Department, Cornell University.

Fagan, J. (1989). The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, 40(2):115–132.

Fagin, R. (1996). Combining fuzzy information from multiple systems. In *Proceedings of the 15th ACM Conference on Principles of Database Systems (PODS)*, pages 216–226.

Fagin, R. (1998). Fuzzy queries in multimedia database systems. In *Proceedings of the 17th ACM Conference on Principles of Database Systems (PODS)*, pages 1–10.

Fisher, H. and Elchesen, D. (1972). Effectiveness of combining title words and index terms in machine retrieval searches. *Nature*, 238:109–110.

Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Lee, D., Petkovix, D., Steele, D., and Yanker, P. (1995). Query by image and video content: The QBIC system. *IEEE Computer Magazine*, 28(9):23–30.

Fox, E. (1983). *Extending the Boolean and vector space models of information retrieval with p-norm queries and multiple concept types*. PhD thesis, Computer Science Department, Cornell University.

Fox, E. and France, R. (1987). Architecture of an expert system for composite document analysis, representation, and retrieval. *Journal of Approximate Reasoning*, 1:151–175.

Fox, E., Nunn, G., and Lee, W. (1988). Coefficients for combining concept classes in a collection. In *Proceedings of the 11th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 291–308.

Fox, E. and Shaw, J. (1994). Combination of multiple searches. In *Proceedings of the 2nd Text Retrieval Conference (TREC-2)*, pages 243–252. National Institute of Standards and Technology Special Publication 500-215.

Frankel, C., Swain, M., and Athitsos, V. (1996). WebSeer: An image search engine for the World Wide Web. Technical Report TR-96-14, University of Chicago Computer Science Department.

Frisse, M. and Cousins, S. (1989). Information retrieval from hypertext: Update on the dynamic medical handbook project. In *Proceedings of ACM Hypertext Conference*, pages 199–212.

Fuhr, N. (1990). A probabilistic framework for vague queries and imprecise information in databases. In *Proceedings of the Very Large Database Conference (VLDB)*, pages 696–707.

Fuhr, N. (1992). Probabilistic models in information retrieval. *Computer Journal*, 35(3):243–255.

Fuhr, N. and Buckley, C. (1991). A probabilistic learning approach for document indexing. *ACM Transactions on Information Systems*, 9(3):223–248.

Gey, F. (1994). Inferring probability of relevance using the method of logistic regression. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–231.

Greiff, W. (1998). A theory of term weighting based on exploratory data analysis. In *Proceedings of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 11–19.

Greiff, W. (1999). *Maximum entropy, weight of evidence, and information retrieval*. PhD thesis, Computer Science Department, University of Massachusetts.

Greiff, W., Croft, W., and Turtle, H. (1997). Computationally tractable probabilistic modeling of Boolean operators. In *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 119–128.

Haines, D. (1996). *Adaptive query modification in a probabilistic information retrieval model*. PhD thesis, Computer Science Department, University of Massachusetts.

Haines, D. and Croft, W. (1993). Relevance feedback and inference networks. In *Proceedings of the 16th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2–11.

Harman, D. (1992). The DARPA TIPSTER project. *ACM SIGIR Forum*, 26(2):26–28.

Harman, D. (1995). Overview of the second text retrieval conference (TREC-2). *Information Processing and Management*, 31(3):271–289.

Harmandas, V., Sanderson, M., and Dunlop, M. (1997). Image retrieval by hypertext links. In *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 296–303.

Hearst, M. and Plaunt, C. (1993). Subtopic structuring for full-length document access. In *Proceedings of the 16th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 59–68.

Heckerman, D., Geiger, D., and Chickering, D. (1994). Learning Bayesian networks: The combination of knowledge and statistical data. In *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*, pages 293–301. Morgan Kaufmann.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57.

Hull, D., Pedersen, J., and Schutze, H. (1996). Method combination for document filtering. In *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 279–287.

Jansen, B., Spink, A., and Saracevic, T. (1998). Real life information retrieval: A study of user queries on the Web. *SIGIR Forum*, 32(1):5–17.

Jelinek, F. (1997). *Statistical methods for speech recognition*. MIT Press, Cambridge.

Kaszkiel, M. and Zobel, J. (1997). Passage retrieval revisited. In *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 178–185.

Katzer, J., McGill, M., Tessier, J., Frakes, W., and DasGupta, P. (1982). A study of the overlap among document representations. *Information Technology: Research and Development*, 1(4):261–274.

Larkey, L. and Croft, W. (1996). Combining classifiers in text categorization. In *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 289–297.

Lee, J. (1995). Combining multiple evidence from different properties of weighting schemes. In *Proceedings of the 18th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 180–188.

Lee, J. (1997). Analyses of multiple evidence combination. In *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 267–276.

Lewis, D. and Hayes, P. (1994). Special issue on text categorization. *ACM Transactions on Information Systems*, 12(3).

Manning, C. and Schutze, H. (1999). *Foundations of statistical natural language processing*. MIT Press, Cambridge.

McGill, M., Koll, M., and Noreault, T. (1979). An evaluation of factors affecting document ranking by information retrieval systems. Final report for grant NSF-IST-78-10454 to the National Science Foundation, Syracuse University.

McLachlan, G. and Krishnan, T. (1997). *The EM algorithm and extensions*. Wiley, New York.

Miller, D., Leek, T., and Schwartz, R. (1999). A Hidden Markov Model information retrieval system. In *Proceedings of the 22nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 214–221.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill, New York.

Mitra, M., Singhal, A., and Buckley, C. (1998). Improving automatic query expansion. In *Proceedings of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 206–214.

Mittendorf, E. and Schauble, P. (1994). Document and passage retrieval based on Hidden Markov Models. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318–327.

MUC-6 (1995). *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, San Mateo.

O'Connor, J. (1975). Retrieval of answer-sentences and answer figures from papers by text searching. *Information Processing and Management*, 11(5/7):155–164.

O'Connor, J. (1980). Answer-passage retrieval by text searching. *Journal of the American Society for Information Science*, 31(4):227–239.

Pao, M. and Worthen, D. (1989). Retrieval effectiveness by semantic and citation searching. *Journal of the American Society for Information Science*, 40(4):226–235.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, San Mateo.

Ponte, J. (1998). *A Language Modeling Approach to Information Retrieval*. PhD thesis, Computer Science Department, University of Massachusetts.

Ponte, J. and Croft, W. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281.

Rajashekar, T. and Croft, W. (1995). Combining automatic and manual index representations in probabilistic retrieval. *Journal of the American Society for Information Science*, 46(4):272–283.

Ravela, C. and Manmatha, R. (1997). Image retrieval by appearance. In *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 278–285.

Robertson, S. (1977). The probability ranking principle in information retrieval. *Journal of Documentation*, 33:294–304.

Robertson, S. and Sparck Jones, K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146.

Salton, G. (1968). *Automatic information organization and retrieval*. McGraw-Hill, New York.

Salton, G. (1971). *The SMART retrieval system - Experiments in automatic document processing*. Prentice-Hall, Englewood Cliffs.

Salton, G. (1974). Automatic indexing using bibliographic citations. *Journal of Documentation*, 27:98–100.

Salton, G., Allan, J., and Buckley, C. (1993). Approaches to passage retrieval in full text information systems. In *Proceedings of the 17th ACM SIGIR*

Conference on Research and Development in Information Retrieval, pages 49–56.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24:513–523.

Salton, G., Fox, E., and Voorhees, E. (1983). Advanced feedback methods in information retrieval. *Journal of the American Society for Information Science*, 36(3):200–210.

Salton, G. and Lesk, M. (1968). Computer evaluation of indexing and text processing. *Journal of the ACM*, 15:8–36.

Salton, G. and McGill, M. (1983). *Introduction to modern information retrieval*. McGraw-Hill, New York.

Salton, G., Wong, A., and Yang, C. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620.

Saracevic, T. and Kantor, P. (1988). A study of information seeking and retrieving. Part III. Searchers, searches, overlap. *Journal of the American Society for Information Science*, 39(3):197–216.

Schneiderman, H. and Kanade, T. (1998). Probabilistic modeling of local appearance and spatial relationships for object recognition. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 45–51.

Small, H. (1973). Co-citation in scientific literature: A new measure of the relationship between two documents. *Journal of the American Society for Information Science*, 24:265–269.

Song, F. and Croft, W. (1999). A general language model for information retrieval. In *Proceedings of the Conference on Information and Knowledge Management (CIKM)*, pages 316–321.

Sparck Jones, K. (1971). *Automatic keyword classification for information retrieval*. Butterworths, London.

Sparck Jones, K. (1974). Automatic indexing. *Journal of Documentation*, 30(4):393–432.

Svenonius, E. (1986). Unanswered questions in the design of controlled vocabularies. *Journal of the American Society for Information Science*, 37(5):331–340.

Tumer, K. and Ghosh, J. (1999). Linear and order statistics combiners for pattern classification. In Sharkey, A., editor, *Combining Artificial Neural Networks*, pages 127–162. Springer-Verlag.

Turtle, H. (1990). *Inference networks for document retrieval*. PhD thesis, Computer Science Department, University of Massachusetts.

Turtle, H. and Croft, W. (1991). Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems*, 9(3):187–222.

Turtle, H. and Croft, W. (1992). A comparison of text retrieval models. *Computer Journal*, 35(3):279–290.

Van Rijsbergen, C. (1979). *Information Retrieval*. Butterworths, London.

Van Rijsbergen, C. (1986). A non-classical logic for information retrieval. *Computer Journal*, 29:481–485.

Vogt, C. and Cottrell, G. (1998). Predicting the performance of linearly combined IR systems. In *Proceedings of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 190–196.

Voorhees, E., Gupta, N., and Johnson-Laird, B. (1995). Learning collection fusion strategies. In *Proceedings of the 18th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 172–179.

Wilkinson, R. (1994). Effective retrieval of structured documents. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 311–317.

Xu, J. and Croft, W. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11.