Query-Based Sampling of Text Databases

Jamie Callan
Carnegie Mellon University
and
Margaret Connell
University of Massachusetts

The proliferation of searchable text databases on corporate networks and the Internet causes a database selection problem for many people. Algorithms such as *gGlOSS* and *CORI* can automatically select which text databases to search for a given information need, but only if given a set of *resource descriptions* that accurately represent the contents of each database. The existing techniques for acquiring resource descriptions have significant limitations when used in wide area networks controlled by many parties.

This paper presents *query-based sampling*, a new technique for acquiring accurate resource descriptions. Query-based sampling does not require the cooperation of resource providers nor does it require that resource providers use a particular search engine or representation technique. An extensive set of experimental results demonstrates that accurate resource descriptions are created, that computation and communication costs are reasonable, and that the resource descriptions do in fact enable accurate automatic database selection.

Categories and Subject Descriptors: H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing Methods*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Search process; Selection Process*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Distributed Systems; Information Networks*

General Terms: Algorithms, Design, Experimentation

Additional Key Words and Phrases: Distributed information retrieval, query-based sampling, resource ranking, resource selection, server selection

---

Callan's work was done in part while at the University of Massachusetts.
Name: Jamie Callan
Address: Language Technologies Institute, School of Computer Science, Carnegie Mellon University, 5000 Forbes Avenue, 4502 Newell Simon Hall, Pittsburgh, PA 15213-3890, USA; email: callan@cs.cmu.edu
Name: Margaret Connell
Address: Center for Intelligent Information Retrieval, Computer Science Department, University of Massachusetts, Amherst, MA 01003-4610, USA; email: connell@cs.umass.edu

1. INTRODUCTION

When many document databases are accessible, the first step of Information Retrieval is deciding where to search. Manual selection can be difficult when there are many databases from which to choose, so researchers have developed automatic *content-based* database selection algorithms. A content-based selection algorithm ranks a set of text databases by how well each database matches or satisfies the given query [15; 7; 36; 14; 38; 3; 42; 39; 11; 26; 19; 40; 10; 27; 12]. Content-based database selection has a number of desirable properties, among them reasonable accuracy, scalability, low computational costs, and ease of use.

Content-based database selection algorithms need information about what each database contains. This information, which we call a *resource description*, is simply assumed to be available in most prior research. However, in practice, accurate resource descriptions can be difficult to acquire in environments, such as the Internet, where resources are controlled by many parties with differing interests and capabilities. Our interest in the research described here was in studying how accurate resource descriptions can be acquired in multi-party environments.

Recent standardization efforts, such as the proposed STARTS extension to Z39.50 [13], illustrate the problem. STARTS requires every resource provider to provide accurate resource descriptions upon request. We call STARTS a *cooperative protocol*, because it only succeeds when each resource provider:

—is able to provide resource descriptions,

—chooses to provide resource descriptions,

—is able to represent database contents accurately, and

—chooses to represent database contents accurately.

Cooperative protocols are appropriate solutions when all resources are controlled by a single party that can mandate cooperation.

In multi-party environments such as the Internet or large corporate networks, complete cooperation is unlikely. Older database systems may be unable to cooperate, some services will refuse to cooperate because they have no incentive or are allied with competitors, and some services may misrepresent their contents, for example, to lure people to the site. All of these characteristics can be found today on the Internet; some of them also occur in large corporate networks.

One of the most serious problems with cooperative techniques is the great variety in how resource descriptions are created. Most of the prior research is based on descriptions consisting of term lists and term frequency or term weight information [15; 7; 14; 36; 26; 19]. However, differences in tokenizing, case conversion, stopword lists, stemming algorithms, proper name handling, and concept recognition are common, making it impossible to compare term frequency information produced by different parties, even if all parties are able and willing to cooperate.

For example, which database is best for the query 'Apple': A database that contains 2000 occurrences of 'appl', a database that contains 500 occurrences of 'apple', or a database that contains 50 occurrences of 'Apple'? The answer requires detailed knowledge about the stopping, stemming, case conversion, and proper name handling performed by each database. Each database could be required to reveal that information, too, but complying would be difficult. Researchers attempting to make identical indexing choices with two different IR systems often find it difficult to identify all of the deliberate choices, system quirks, and outright errors that lead to a particular term statistic. A resource selection algorithm that depended on that level of detail seems impractical.

Resource selection algorithms require accurate and consistent resource descriptions. However, the weaknesses of cooperative protocols make them an unsuitable solution for environments where resources are controlled by many parties. In these environments, a different solution is required.

Query-based sampling is a recently developed method of acquiring resource descriptions that does not require explicit cooperation from resource providers [5]. Instead, resource descriptions are created by running queries and examining the documents that are returned. Resource descriptions can be guaranteed to be compatible because they are created under the control of the sampling process, not each individual resource provider. Preliminary experiments suggested that query-based sampling is an effective and efficient method of acquiring resource descriptions.

The preliminary experiments studied how closely a resource description created by sampling (a *learned* resource description) matched the *actual* resource description for a text database [5]. The results were encouraging but inconclusive, in part due to a flawed experimental methodology. This paper reproduces the earlier experiments using an improved experimental methodology. It also extends the prior research by investigating the effects of learned resource descriptions on a resource selection task. The result is a comprehensive study of the efficiency, effectiveness, and robustness of query-based sampling under a variety of conditions.

The next section reviews prior research on resource selection and distributed information retrieval, emphasizing the information requirements of several representative algorithms. Section 3 describes query-based sampling in more detail. Sections 4, 5 and 6 describe experiments that test basic hypotheses about query-based sampling, its sensitivity to parameter settings, and the efficacy of the resulting resource descriptions for resource selection and multi-database retrieval. Section 7 discusses the use of query-based sampling for summarizing database contents. Section 8 discusses other uses for query-based sampling, and Section 9 concludes.

2. PRIOR RESEARCH

Automatic selection among text databases has been studied since at least the early 1980s, when the EXPERT CONIT system was developed [25]. CONIT used a rule-based system to select among a small set of databases, but few details were published about how how database contents were represented or matched to queries.

A variety of different approaches to database selection were developed beginning in the mid 1990s. The most common approach is exemplified by *gGlOSS* [15; 14], *CORI* [7; 39], *Cue Validity Variance* (CVV) [41], and Xu's *language modeling* approach [40]. This family of algorithms represents database contents by the words contained in the database, and by statistics computed from word frequencies. These algorithms can be viewed as adapting document ranking representations and algorithms to the database ranking task. They are easily scaled to large numbers of databases, they are computationally efficient, and no manual effort is required to create or update database descriptions. The main problem in applying this family of algorithms is obtaining accurate descriptions of each database.

Query clustering and *RDD* [36] are algorithms that rank databases using information about the distribution of relevant documents for similar queries seen in the past. These two algorithms represent databases by their prior effectiveness for past queries, which makes it easy to control their behavior relatively precisely without knowing anything about the contents of the database. They also make it easy to integrate databases served by different search engines [37], because there is no need to compare representations or frequencies produced by different search engines. The main problem in applying these algorithms is that relevance judgements require manual effort, so it can be expensive to apply them when there are many databases or databases that are updated often.

FreeNet [8] is a peer-to-peer search algorithm that passes queries from node to node in a network until either a search horizon is reached or the query is satisfied. FreeNet nodes keep track of which other nodes have been successful in satisfying past queries, where 'satisfying' is defined as matching a Boolean query. That information is used later to route new queries. Although its architecture is rather different, the FreeNet search strategy is similar to the strategies of the RDD and Query Clustering algorithms under a weaker definition of relevance.

Query probing [19] is an algorithm that sends a two-word subset of a query (a 'probe query') to each database to discover how often the words occur and co-occur in each database. Query probing requires no advance knowledge of the contents of each database, so it is easy to apply in environments that change often. However, query probing requires a method of generating good probe queries, it requires that each database cooperate by providing the requested statistics, and it can entail significant communications costs in wide area networks or when large numbers of databases are available.

There are other database selection algorithms (e.g., [38; 3; 9], but they require similar information. Database selection algorithms may differ significantly in their architectures and assumptions, but they usually represent database contents in one of two ways: i) with information about queries satisfied in the past, or ii) with term frequency information. When using a database selection algorithm, one must have a strategy for obtaining the required information and keeping it current.

Our research interest is in database selection algorithms that represent database contents by term frequency information, because the algorithms are effective and easy to scale to large numbers of databases [14; 10; 31]. The principal problem in applying these algorithms is determining the contents of each database.

One solution is for databases to exchange term frequency information periodically [35]. This approach was formalized in STARTS [13], a standard protocol for describing database contents. A STARTS description lists the indexing terms used in a database, their frequencies, and information about word stemming, stopwords, and other indexing choices that affect frequency information. STARTS was designed to be applied on a large-scale, for example as part of the Z39.50 protocol for communicating with information retrieval systems [30]. However, as described above (Section 1), STARTS assumes that it is possible to compare frequency information provided by different parties, which is rarely true in practice. The STARTS protocol may be the preferred solution when all databases are controlled by a single party, but solutions such as query-based sampling are required when databases are controlled by many parties.

3. QUERY-BASED SAMPLING

Our goal is a method of acquiring resource descriptions that is not overly complex, that does not require special cooperation from resource providers, that can be applied to older ("legacy") systems, that is difficult (but not necessarily impossible) to deceive, and that is not sensitive to indexing choices made by resource providers.

It is well-known that the characteristics of a population can be determined to a desired degree of accuracy by random sampling. It is also well-known that word occurrence patterns in a corpus are very skewed. Zipf's Law states that a word's rank multipled by its frequency is approximately equal to a constant [43]. The skew described by Zipf's Law means that usually 75% of the unique words in a corpus occur 3 or fewer times [20]. A sampling technique might produce a very accurate indication of database contents even if fails to find a very large percentage of the words.

Heaps' Law provides further support for discovering database contents by sampling. Heaps' Law states that the size of a corpus vocabulary can be estimated by $V \approx KN^{\beta}$, where $K \approx 20$, N is the number of corpus word occurrences, and $0.4 \leq \beta \leq 0.6$ [20]. As a corpus is scanned, the vocabulary initially grows very rapidly, albeit exhibiting the frequency skew described by Zipf. As scanning continues, the vocabulary growth rate tapers off. Heap's Law suggests that it is not necessary to examine much of a corpus in order to discover most of its vocabulary.

Random selection is a cooperative method of of discovering database contents, because it depends upon *the provider* to select documents randomly from its database, which the provider might or might not do. Random selection is not a solution, but it suggests a solution.

Third parties can obtain *biased samples* of databases by running queries and examining the documents returned in response. We call this *query-based sampling*, to emphasize the biased nature of each sample. Query-based sampling satisfies all of the criteria outlined above, because it assumes only that database providers perform their usual service of running queries and returning documents.

Our central hypothesis is that a sufficiently unbiased sample of documents can be constructed from the union of biased samples obtained by query-based sampling.

Query-based sampling is implemented with a simple algorithm, outlined below.

(1) Select an initial query term.
(2) Run a one-term query on the database.
(3) Retrieve the top N documents returned by the database.
(4) Update the resource description based on the characteristics of the retrieved documents.
    (a) Extract words and frequencies from the top N documents returned by the database; and
    (b) Add the words and their frequencies to the learned resource description.
(5) If a stopping criterion has not yet been reached,
    (a) Select a new query term; and
    (b) Go to Step 2.

The algorithm involves several choices, for example how query terms are selected, how many documents to examine per query, and when to stop sampling. Discussion

Table 1. Test corpora.

Name	Size, in bytes	Size, in documents	Size, in unique terms	Size, in total terms	Variety
CACM	2MB	3,204	6,468	117,473	homogeneous
WSJ88	104MB	39,904	122,807	9,723,528	heterogeneous
TREC-123	3.2GB	1,078,166	1,134,099	274,198,901	very heterogenenous

of these choices is deferred to later sections of the paper.

How best to represent a large document database is an open problem. However, much of the prior research is based on simple resource descriptions consisting of term lists, term frequency or term weight information, and information about the number of documents [15; 14; 36] or number of words [7; 39; 40] contained in the resource. Zipf's Law and Heap's Law suggest that relatively accurate estimates of the first two pieces of information, term lists and the relative frequency of each term, can be acquired by sampling [20; 43].

It is not clear whether the size of a resource can be estimated with query-based sampling, but it is also not clear that this information is actually required for accurate database selection. We return to this point later in the paper.

The hypothesis motivating our work is that sufficiently accurate resource descriptions can be learned by sampling a text database with simple 'free-text' queries. This hypothesis can be tested in two ways:

(1) by comparing resource descriptions learned by sampling known databases (*'learned resource descriptions'*) with the *actual resource descriptions* for those databases, and

(2) by comparing resource selection accuracy using learned resource descriptions with resource selection using actual resource descriptions.

Both types of experiments were conducted and are discussed below.

4. EXPERIMENTAL RESULTS: DESCRIPTION ACCURACY

The first set of experiments investigated the accuracy of learned resource descriptions as a function of the number of documents examined. The experimental method was based on comparing learned resource descriptions for known databases with the actual resource descriptions for those databases.

The goals of the experiments were to determine whether query-based sampling learns accurate resource descriptions, and if so, what combination of parameters produce the fastest or most accurate learning. A secondary goal was to study the sensitivity of query-based sampling to parameter settings.

The following sections describe the data, the type of resource description used, the metrics, parameter settings, and finally, experimental results.

4.1 Data

Three full-text databases were used:

CACM: a small, homogeneous set of titles and abstracts of scientific articles from the *Communications of the ACM*;

WSJ88: the 1988 *Wall Street Journal*, a medium-sized corpus of American newspaper articles;[1] and

TREC-123: a large, heterogeneous database consisting of TREC CDs 1, 2, and 3, which contains newspaper articles, magazine articles, scientific abstracts, and government documents [18].

These are standard test corpora used by many researchers. Their characteristics are summarized in Table 1.

4.2 Resource Descriptions

Experiments were conducted on resource descriptions consisting of index terms (usually words) and their document frequencies, *df* (the number of documents containing each term).

Stopwords were not discarded when learned resource descriptions were constructed. However, during testing, learned and actual resource descriptions were compared only on words that appeared in the actual resource descriptions, which effectively discarded from the learned resource description any word that was considered a stopword by the database. The databases each used the default stopword list of the INQUERY IR system [34; 33; 6], which contained 418 very frequent and/or closed-class words.

Suffixes were not removed from words ('stemming') when resource descriptions were constructed. However, during controlled testing, suffixes were removed prior to comparison to the actual resource description, because the actual resource descriptions (the database indexes) were stemmed.

4.3 Metrics

Resource descriptions consisted of two types of information: a *vocabulary*, and *frequency information* for each vocabulary term. The correspondence between the learned and actual vocabularies was measured with a metric called *ctf ratio*. The correspondence between the learned and actual frequency information was measured with the *Spearman Rank Correlation Coefficient*. Each metric is described below.

4.3.1 *Measuring Vocabulary Correspondence:* Ctf *Ratio.* The terms in a learned resource description are necessarily a subset of the terms in the actual description. One could measure how many of the database terms are found during learning, but such a metric is skewed by the many terms occurring just once or twice in a collection [43; 20]. We desired a metric that gave more emphasis to the frequent and moderately-frequent terms, which we believe convey the most information about the contents of a database.

Ctf ratio is the proportion of term occurrences in the database that are covered by terms in the learned resource description. For a learned vocabulary V' and an actual vocabulary V, *ctf* ratio is:

$$\frac{\sum_{i \in V'} ctf_i}{\sum_{i \in V} ctf_i} \tag{1}$$

[1]The 1988 Wall Street Journal data (WSJ88) is included on TREC CD 1. WSJ88 is about 10% of the text on TREC CD 1.

Table 2. *ctf* ratio example.

Actual Resource Description	
Vocabulary	*ctf*
apple	4
bear	1
cat	3
dog	2

Learned Resource Descriptions		
	Vocabulary	*ctf* ratio
LRD 1	apple	40%
LRD 2	bear	10%
LRD 3	apple, cat	70%

where ctf_i is the number of times term i occurs in the database (collection term frequency, or *ctf*). A *ctf* ratio of 80% means that the learned resource description contains the terms that account for 80% of the term occurrences in the database.

For example, suppose a database consists of 4 occurrences of "apple", 1 occurrence of "bear", 3 occurrence of "cat", and 2 occurrences of "dog" (Table 2). If the learned resource description contains only the word "apple" (25% of the actual vocabulary terms), the *ctf* ratio is 4 / 10 = 40%, because the word "apple" accounts for 40% of the word occurrences in the database. If the learned resource description contains both "apple" and "cat", the *ctf* ratio is 70%. *ctf* ratio measures the degree to which the learned resource description contains the words that are frequent in the actual resource description.

Note that the *ctf* ratios reported in this paper are not artificially inflated by finding stopwords, because *ctf* ratio was always computed *after* stopwords were removed.

4.3.2 *Spearman Rank Correlation Coefficient.* The second component of a resource description is document frequency information (*df*), which indicates the relative importance of each term in describing the database. The accuracy of frequency information can be determined either by comparison of learned and actual *df* values after appropriate scaling, or by comparison of the frequency-based term rankings produced by learned and actual *df* values. The two measurement methods emphasize different characteristics of the frequency information.

Direct comparison of *df* values has the undesirable characteristic that the comparison is biased in favor of estimates based on larger amounts of information, because estimates based on 10^n documents enable only n digits of accuracy in scaled values. This characteristic was a concern because even relatively noisy *df* estimates based on small numbers of documents might be sufficient to enable accurate resource selection.

Term rankings produced by learned and actual *df* values can be compared by the Spearman Rank Correlation Coefficient, an accepted metric for comparing two orderings. The Spearman Rank Correlation Coefficient is defined [32] as:

$$R = \frac{1 - \frac{6}{n^3-n}(\Sigma d_i^2 + \frac{1}{12}\Sigma(f_k^3 - f_k) + \frac{1}{12}\Sigma(g_m^3 - g_m))}{\sqrt{(}1 - \frac{\Sigma(f_k^3-f_k)}{n^3-n})\sqrt{(}1 - \frac{\Sigma(g_m^3-g_m)}{n^3-n})} \quad (2)$$

where d_i is the rank difference of common term i, n is the number of terms, f_k is the number of ties in the kth group of ties in the learned resource description, and g_m is the number of ties in the mth group of ties in the actual resource description. Two orderings are identical when the rank correlation coefficient is 1. They are uncorrelated when the coefficient is 0, and they are in reverse order when the coefficient is -1.

The complexity of this variant of the Spearman Rank Correlation Coefficient may surprise some readers. Simpler versions are more common (e.g., [28]). However, simpler versions assume a total ordering of ranked elements; two elements cannot share the same ranking. Term rankings have *many* terms with identical frequencies, and hence identical rankings. Variants of the Spearman Rank Correlation Coefficient that ignore the effects of tied rankings can give misleading results, as was the case in our initial research on query-based sampling [5].

The Spearman Rank Correlation Coefficient was computed using just the terms in the intersection of V and V'. Use of the intersection is appropriate because the Spearman Rank Correlation Coefficient is used to discover whether the terms in V' are ordered appropriately by the learned frequency information.

Database selection does not require a rank correlation coefficient of 1.0. It is sufficient for the learned resource description to represent the relative importance of index terms in each database to some degree of accuracy. For example, it might be sufficient to know the ranking of a term $\pm 5\%$. Although most database selection algorithms are likely to be insensitive to small ranking errors, it is an open question how much error a given algorithm can tolerate before selection accuracy deteriorates.

4.4 Parameters

Experiments with query-based sampling require making choices about how query terms are selected and how many documents are examined per query.

In our experiments, the first query run on a database was determined by selecting a term randomly from the TREC-123 vocabulary. The initial query could be selected using other criteria, for example selecting a very frequent term, or it could be selected from another resource. Several informal experiments found that the choice of the initial query term had minimal effect on the quality of the resource description learned and the speed of learning, as long as it retrieved at least one document.

Subsequent query terms were chosen by a variety of methods, as described in the following sections. However, in all cases the terms chosen were subject to requirements similar to those placed on index terms in many text retrieval systems: A term selected as a query term could not be a number, and was required to be 3 or more characters long.

We had no hypotheses to guide the decision about how many documents to sample per database query. Instead, a series of experiments was conducted to determine the effect of varying this parameter.

The experiments presented below were ended after examining 500 documents. This stopping criteria was chosen empirically after running several initial experiments, and were biased by our interest in learning resource descriptions from small (ideally, constant) sized samples. Several experiments with each database were continued until several thousand documents were sampled, to ensure that nothing unusual happened.

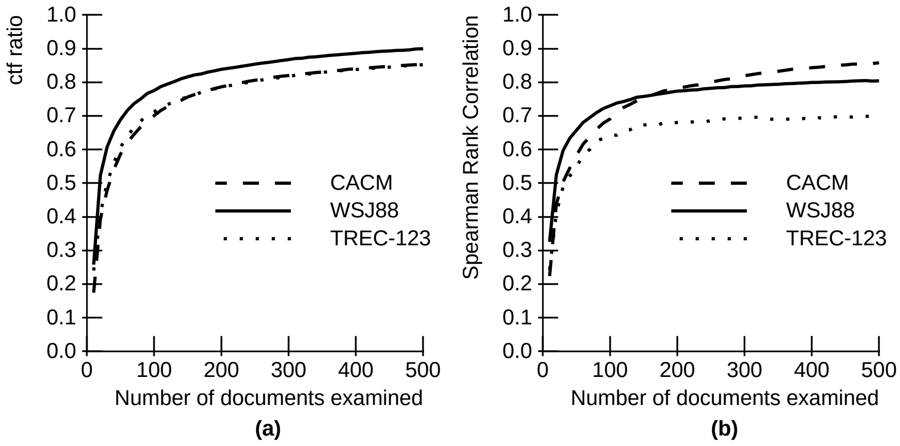

Fig. 1. Measures of how well a learned resource description matches the actual resource description of a full-text database. **(a)** Percentage of database word occurrences covered by terms in the learned resource description. **(b)** Spearman rank correlation coefficient between the term rankings in the learned resource description and the database. (Four documents examined per query. Each point is the average of 10 trials.)

4.5 Results

Four sets of experiments were conducted to study the accuracy of resource descriptions learned under a variety of conditions. The first set of experiments was an initial investigation of query-based sampling with the parameter settings discussed above. We call these the *baseline experiments.* A second set of experiments studied the effect of varying the number of documents examined per query. A third set of experiments studied the effect of varying the way query terms were selected. A fourth set of experiments studied the effect of varying the choice of the collection from which documents were picked. Each set of experiments is discussed separately below.

4.5.1 *Results of Baseline Experiments.* The *baseline experiments* were an initial investigation of query-based sampling. The goal of the baseline experiments was to determine whether query-based sampling produced accurate resource descriptions, and if so, how accuracy varied as a function of the total number of documents examined.

The initial query term was selected randomly from the TREC-123 resource description, as described above. Subsequent query terms were selected randomly from the resource description being learned.

The top four documents retrieved by each query were examined to update the resource description. Duplicate documents, that is, documents that had been retrieved previously by another query, were discarded, hence some queries produced fewer than four documents.

Ten trials were conducted, each starting from a different randomly selected query term, to compensate for the effects of random query term selection. The experimental results reported here are averages of results returned by the ten trials.

Table 3. Effect of varying the number of documents examined per query on how long it takes a sampling method to reach a *ctf* ratio of 80%.

Documents	CACM		WSJ88		TREC-123	
Per	Total		Total		Total	
Query	Docs	Spearman	Docs	Spearman	Docs	Spearman
1	257	0.80	113	0.76	183	0.70
2	242	0.80	116	0.74	200	0.65
4	232	0.80	126	0.75	239	0.68
6	236	0.80	122	0.74	241	0.68
8	236	0.81	111	0.74	244	0.69
10	233	0.81	120	0.74	246	0.66

The variation in the measurements obtained from each trial on a particular database was large ($10-15\%$) at 50 documents, but decreased rapidly. At 150 documents it was $4-5\%$, and at 250 documents it was $2-4\%$. The consistency among the trials suggests that the choice of the initial query term is not particularly important, as long as it returns at least one document. (The effects of different strategies for selecting subsequent query terms are addressed in Section 4.5.3.)

Figure 1a shows that query-based sampling quickly finds the terms that account for 80% of the non-stopword term occurrences in each collection.[2] After about 250 documents, the new vocabulary being discovered consists of terms that are relatively rare in the corpus, which is consistent with Zipf's law [43].

Figure 1b shows the degree of agreement between the term orderings in the learned and actual resource descriptions, as measured by the Spearman Rank Correlation Coefficient. A high degree of correlation between learned and actual orderings is observed for all collections after seeing about 250 documents. The correlation observed for the largest collection (TREC-123) is less than the correlations observed for the smaller collections (CACM and WSJ88). Extending the number of documents sampled beyond 500 does not substantially improve the correlation measure on this large collection.

Results from both metrics support the hypothesis that accurate resource descriptions can be learned by examining only a small fraction of the collection. This result is encouraging, because it suggests that query-based sampling is a viable method of learning accurate resource descriptions.

4.5.2 *Results of Varying Sample Size.* The baseline experiments sampled the four most highly ranked documents retrieved for each query. However, the sampling process could have retrieved more documents, or fewer documents, per query. Doing so could change the number of queries and/or documents required to achieve a given level of accuracy, which in turn could affect the costs of running the algorithm.

A series of experiments was conducted to investigate the effects of varying the number of documents examined per query. Values of 1, 2, 4, 6, 8, and 10 documents per query were tested. As in the prior experiment, ten trials were conducted for each value, each trial starting from a different randomly selected query term,

[2]Recall that stopwords were excluded from the comparison. If stopwords were included in the comparison, the rate of convergence would be considerably faster.

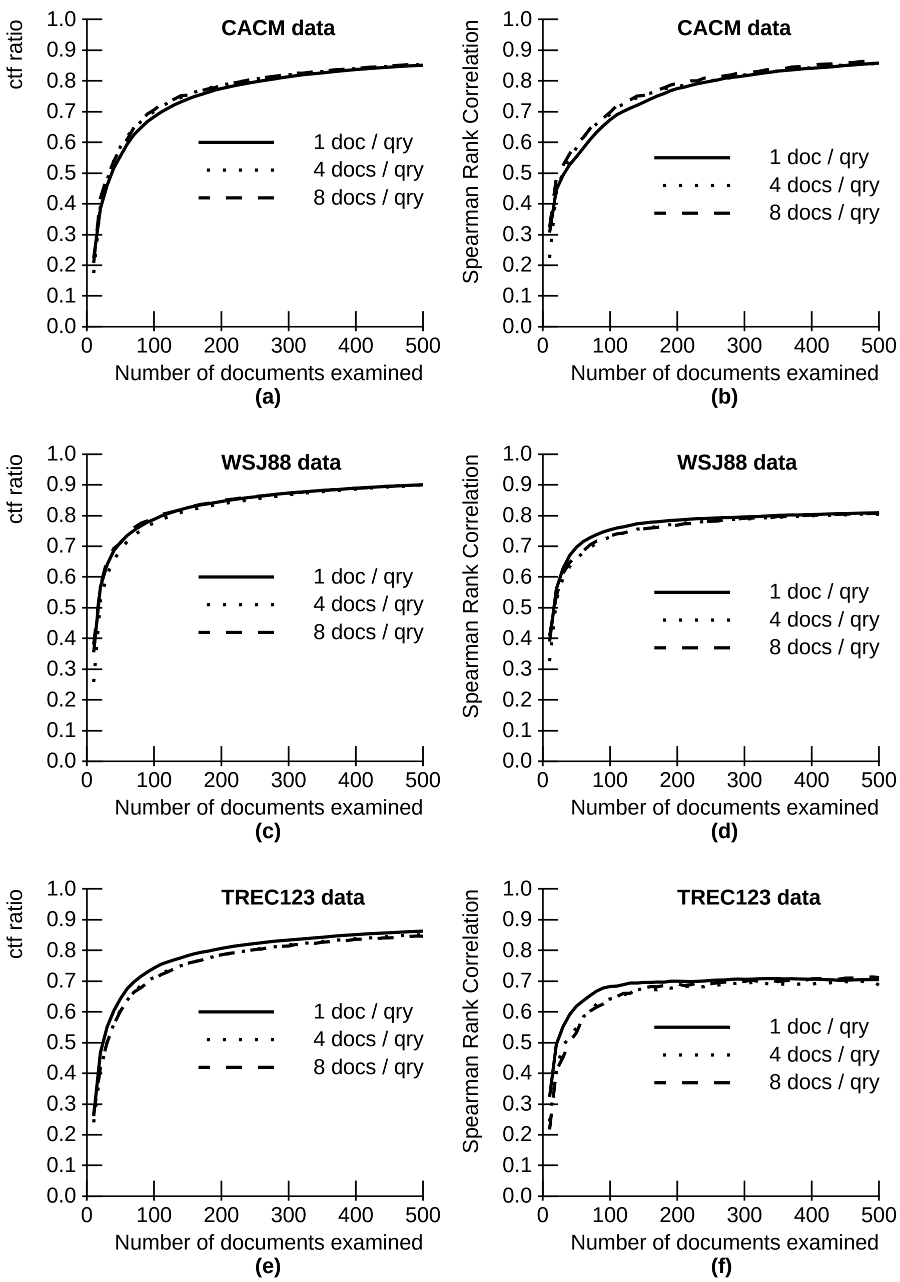

Fig. 2. Measures of how well a learned resource description matches the actual resource description of a full-text database. Each point is the average of 10 trials. **(a), (c), and (e):** Percentage of database word occurrences covered by terms in the learned resource description. **(b), (d), and (f):** Spearman rank correlation coefficient between the term rankings in the learned resource description and the database.

with subsequent query terms chosen randomly from the resource description being learned. Each experimental result reported below is an average of the experimental results from ten trials.

Varying the number of documents per query had little effect on the speed of learning, as measured by the average number of documents required to reach a given level of accuracy. Indeed, the effect was so small that it is difficult to display the results of different values on a single graph. Figure 2 shows results for values of 1, 4, and 8 documents per query on each database. Results for values of 2, 6, and 10 were very similar.

Table 3 provides another perspective on the experimental results. It shows the number of documents required to reach a *ctf* ratio of 80%. Varying the number of documents examined per query from 1 to 10 caused only minor variations in performance for 2 of the 3 databases.

Careful study reveals that examining more documents per query results in slightly faster learning (fewer queries required) on the small, homogeneous CACM database; examining fewer documents per query results in somewhat faster learning on the larger, heterogeneous TREC123 database. However, the effects of varying the number of documents per query are, *on average*, small. The most noticeable effect is that examining fewer documents per query results in a *more consistent* learning speed on *all* databases. There was greater variation among the ten trials when 10 documents were examined per query ($\approx 3 - 5\%$) than when 1 document was examined per query ($\approx 1 - 3\%$).

In this experiment, larger samples worked well with the small homogeneous collection, and smaller samples worked well with the large heterogeneous collection. We do not find this result surprising. Samples are biased by the queries that draw them; the documents within a sample are necessarily similar to some extent. We would expect that many small samples would better approximate a random sample than fewer large samples in collections where there is significant heterogeneity. The results support this intuition.

4.5.3 *Results of Varying Query Selection Strategies.* The baseline experiments select query terms randomly from the resource description being learned. Other selection criteria could be used, or terms could be selected from other sources.

One hypothesis was that it would be best to select terms that appear to occur frequently in the collection, i.e., words that are nearly frequent enough to be stopwords, because they would return the most random sample of documents. We tested this hypothesis by selecting frequent query terms, as measured by document frequency (*df*), collection term frequency (*ctf*), and average term frequency (*avg_tf* = *ctf* / *df*).

One early concern was that learned resource descriptions would be strongly biased by the set of documents that just happened to be examined first, and that this bias would be reinforced by selecting additional query terms from the learned resource description. A solution would be to select terms from a different, more complete resource description. This hypothesis was named the *other resource description*, or *ord* hypothesis, and was compared to the default *learned resource description* or *lrd* approach used in the other experiments. The complete TREC-123 resource description served as the 'other' resource description.

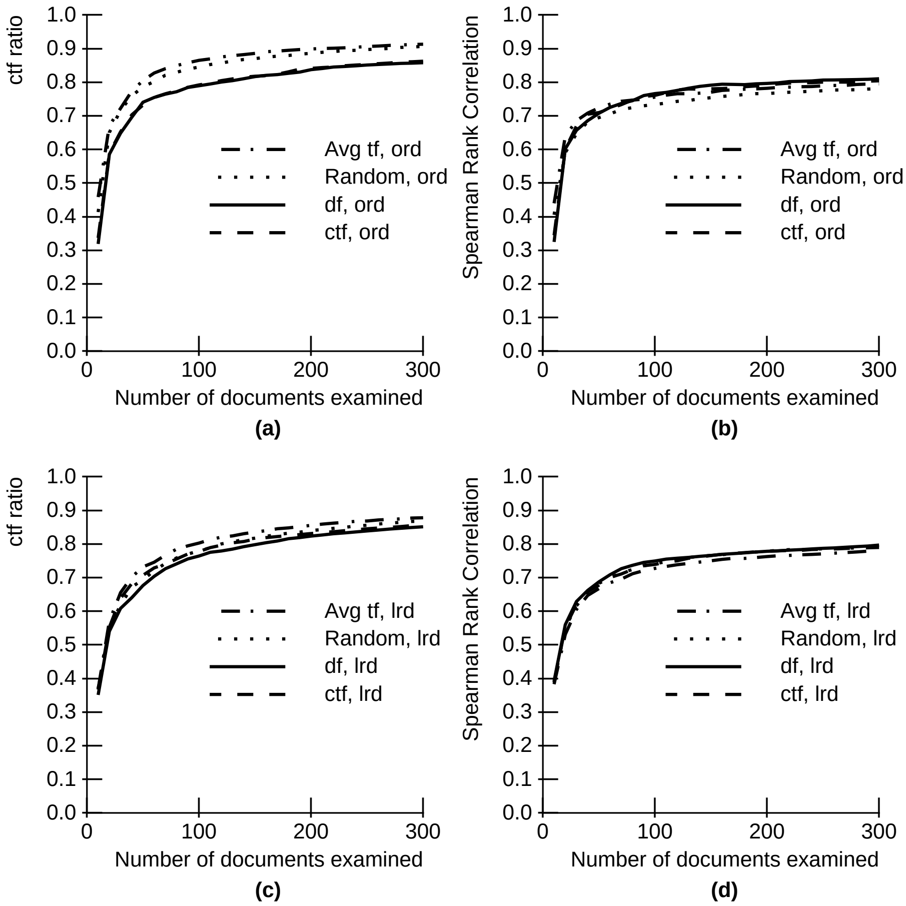

Fig. 3. Measures of how different query selection strategies affect the accuracy of a learned resource description. **(a) and (c):** Percentage of database word occurrences covered by terms in the learned resource description. **(b) and (d):** Spearman rank correlation coefficient between the term rankings in the learned resource description and the database. (1988 Wall Street Journal database. Four documents examined per query. Each point for the random and lrd curves is the average of 10 trials.)

The choice of TREC-123 as the 'other' resource description might be challenged, because WSJ88 is a subset of TREC-123. It is possible that TREC-123 might be a biased, or an unrealistically good, 'other' resource description from which to select terms for sampling WSJ88. We were aware of this possible bias, and were prepared to conduct more thorough experiments if the initial results appeared to confirm the 'other' resource description hypothesis.

A series of experiments was conducted, following the same experimental methodology used in previous experiments, except in how query terms were selected. Query terms were selected either randomly or based on one of the frequency criteria, from either the learned resource description (*lrd*) or the 'other' resource description (*ord*). Four documents were examined per query. Ten trials were conducted for

Table 4. The differences between selecting query terms from an other resource description (ord) or learned resource description (lrd). 'Significant At & Above' is the point on the curves in Figure 3 at which the difference between selecting from ord and lrd resources becomes statistically significant (t test, $p < 0.01$). Values for learned resource descriptions and the random selection method are averages of 10 trials.

		ctf ratio					
Selection Method	Significant At & Above	100 Documents ord	100 Documents lrd	200 Documents ord	200 Documents lrd	300 Documents ord	300 Documents lrd
avg_tf	20 docs	0.8651	0.8026	0.8989	0.8552	0.9130	0.8779
random	20 docs	0.8452	0.7787	0.8859	0.8401	0.9067	0.8678
ctf	190 docs	0.7920	0.7774	0.8412	0.8310	0.8625	0.8558
df	130 docs	0.7895	0.7641	0.8374	0.8234	0.8580	0.8511

each method that selected query terms randomly or from the learned resource description (*lrd*), to compensate for random variation and order effects. Experiments were conducted on all three collections, but results were sufficiently similar that only results for the WSJ88 collection are presented here.

In all of the experiments, selecting terms from the 'other' resource description produced faster learning, as measured by the number of documents required to reach a given level of accuracy (Figure 3). The differences were statistically significant for all four term selection methods (t test, $p < 0.01$). However, the differences were relatively large for the *avg_tf* and random selection methods, and were statistically significant after only 20 documents were observed; the differences were small for the *ctf* and *df* selection methods, and required 130 and 190 documents respectively to achieve statistical significance (Table 4). There might be some value to using an other resource description for *avg_tf* and random term selection methods, but there appears to be little value for the *ctf* and *df* selection methods.

One weakness of selecting query terms from an other resource description is that it can provide terms that do not appear in the target resource ('out of vocabulary' query terms). This characteristic is particularly noticeable with *avg_tf* and random term selection. *Avg_tf* and random selection from an other resource description produced the most accurate results (Table 4), but required many more queries to retrieve a given number of unique documents due to 'out of vocabulary' queries (Table 5). Recall also that the 'other' resource description (TREC-123) was a superset of the target database (WSJ88). The number of failed queries might have been higher if the 'other' resource description had been a less similar database.

The experiments demonstrate that selecting query terms from the learned resource description, as opposed to a more complete 'other' resource description, does *not* produce a strongly skewed sample of documents. Indeed, random and *avg_tf* selection of query terms from the learned resource description provided the best balance of accuracy and efficiency in these experiments. The worst-case behavior, obtained with an other resource description that is a poor match for the target resource, would also favor selecting terms from the learned resource description.

The experiments also demonstrate that selecting query terms randomly from the learned resource description is more effective than selecting them based on high frequency. This result was a surprise, because our hypothesis was that high

Table 5. The number of queries required to retrieve 300 documents using different query selection criteria.

Selection strategy	Random, ord	Random, lrd	avg_tf, ord	avg_tf, lrd	df, ord	df, lrd	ctf, ord	ctf lrd
Number of queries	378	84	6,673	112	78	154	77	154

frequency terms would either occur in many contexts, or would have relatively weak contexts, producing a more random sample. That hypothesis was not supported by the experiments.

4.5.4 *Results of Varying the Databases Sampled.* The results of the experiments described in the preceding sections support the hypothesis that database contents can be determined by query based sampling. However, they do not rule out a competing hypothesis: That a relatively random sample of documents from nearly *any* American English database would produce an equally accurate description of the three test databases. Perhaps these experiments merely reveal properties of American discourse, for example, that certain words are used commonly.

If the competing hypothesis is true, then query-based sampling is not necessary; a partial description from any relatively similar resource would produce similar results at lower computational cost. More importantly, it would cast doubt on whether partial resource descriptions distinguish databases sufficiently to enable accurate database selection. If the partial resource descriptions for most American English databases are very similar, a database selection algorithm would presumably have great difficulty identifying the databases that best match a specific information need.

A series of experiments was conducted to test the hypothesis that relatively random samples of documents from different American English database would produce equally accurate descriptions of the three test databases.

The experimental method consisted of comparing the resource descriptions created by query-based sampling of various databases to the actual, complete resource description for the test databases. For example, resource descriptions created by query-based sampling of CACM, WSJ88, and TREC-123 databases were compared to the actual description for the CACM database (Figures 4a and 4b). The hypothesis would be supported if each of the learned resource descriptions were roughly comparable in how well they matched the actual, complete resource description of a particular database.

Experiments were conducted with the CACM, WSJ88, and TREC-123 databases. Comparisons were performed over 300-500 examined documents. The experimental results are summarized in Figure 4.

The experimental results indicate that a description learned for one resource, particularly a large resource, can contain the vocabulary that occurs frequently in other resources. For example, the resource descriptions learned for the TREC-123 database contained the vocabulary that is frequent, and presumably important, in the WSJ88 and CACM databases (Figures 4a and 4c). The results also suggest that prior knowledge of database characteristics might be required to decide which descriptions to use for each database. The CACM resource description, for example, lacked much of the vocabulary that is important to both the WSJ88 and TREC-123

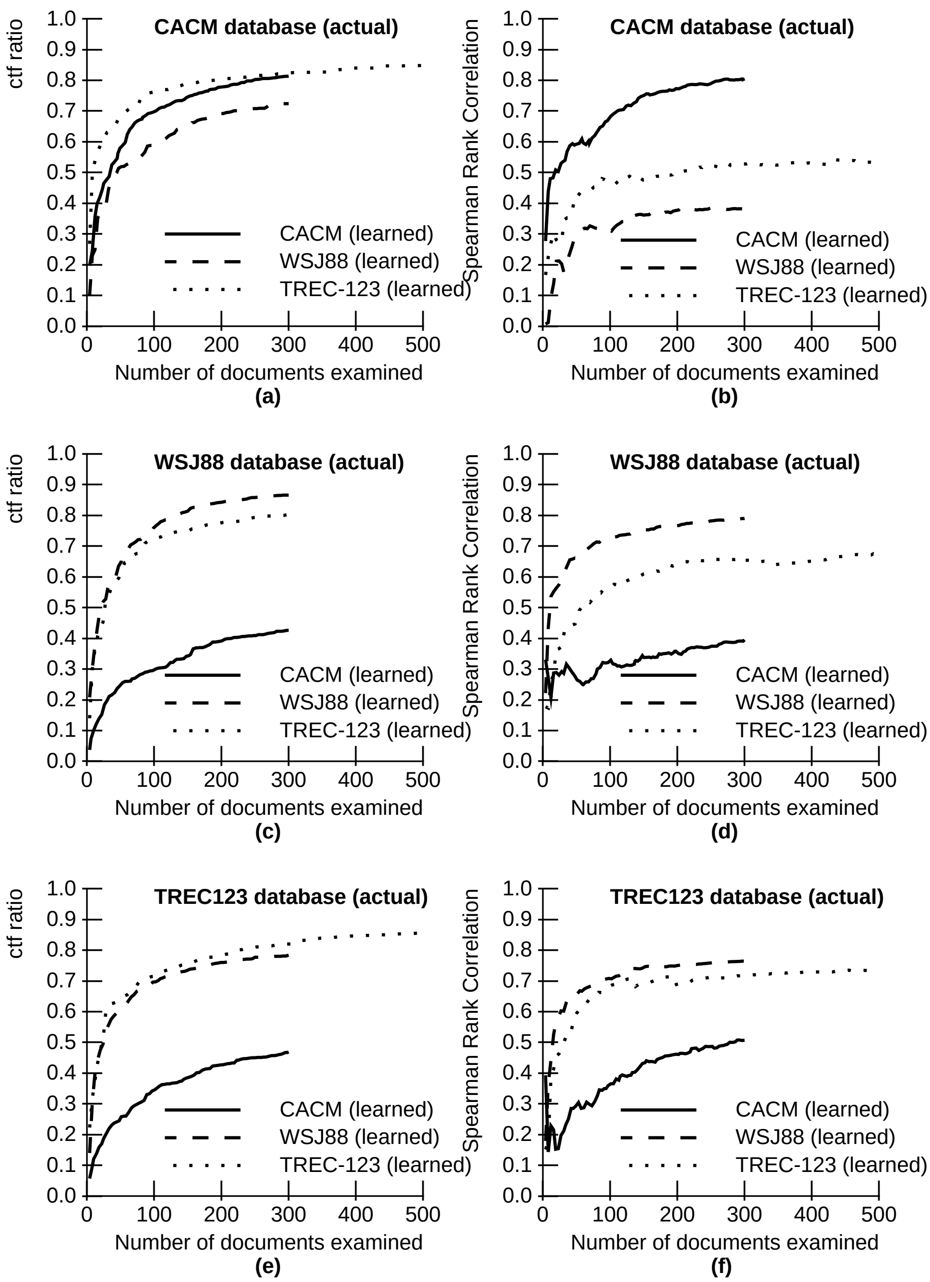

Fig. 4. Measures of how well learned resource descriptions for three different databases match the actual resource description of a given database. **(a), (c) and (e):** Percentage of actual database term occurrences that are covered by terms in different learned resource descriptions. **(b), (d) and (f):** Spearman rank correlation coefficient between the actual term rankings and term rankings in different learned resource description. (Four documents examined per query.)

resources (Figures 4c and 4e).

The problem with using the description learned for one resource to describe another, different resource is more apparent when relative term frequency is considered. Relative term frequency is important because it indicates which terms are common in a database, and most database selection algorithms prefer databases in which query terms are common. In these experiments, the relative frequency of vocabulary items in the three test databases was rarely correlated (Figures 4b, 4d, and 4f). For example, neither the WSJ88 nor the TREC-123 databases gave an accurate indication of relative term frequency in the CACM database (Figure 4b). Likewise, neither the CACM nor the TREC-123 database gave an accurate indication of term frequency for the WSJ88 database (Figure 4d). The one exception to this trend was that the WSJ88 database did appear to give a relatively accurate indication of relative term frequency in the TREC-123 database (Figure 4f).[3]

These experiments refute the hypothesis that the experimental results of the earlier sections are based upon language patterns that are common across different collections of American English text. There may be considerable overlap of vocabulary among the different databases, but there are also considerable differences in the relative frequencies of terms in each database. For example, the term "computer" occurs in all three databases, but its relative frequency is much higher in the CACM database than in the WSJ88 and TREC-123 databases.

Post-experiment analysis indicates that an improved experimental methodology would provide even stronger evidence refuting the alternate hypothesis. The *ctf* ratio does not measure the fact that the description learned for TREC-123 contains many terms not in the CACM database (Figure 4a). Hence, the *ctf* ratio results in Figures 4a, 4c, and 4e can overstate the degree to which the learned vocabulary from one database reflects the actual vocabulary of a different database. A large dictionary of American English would yield a *ctf* ratio close to 1.0 for all three of our databases, but few people would argue that it accurately described any of them.

5. EXPERIMENTAL RESULTS: SELECTION ACCURACY

The experiments described in the previous section investigate how quickly and reliably the learned resource description for a database converges upon the actual resource description. However, we do not know how accurate a resource description needs to be for accurate resource selection. Indeed, we do not even know that description accuracy is correlated with selection accuracy, although we presume that it is.

The second group of experiments investigated the accuracy of resource selection as a function of the number of documents examined. The experimental method was based on comparing the effectiveness of the *database ranking algorithm* when using complete and learned resource descriptions. Databases were ranked with the INQUERY IR system's default database ranking algorithm [7; 4].

The following sections describe the data, the type of resource description used, the metrics, parameter settings, and finally, experimental results.

[3]This exception may be caused by the fact that about 10% of the TREC-123 database consists of Wall Street Journal data.

Table 6. Summary statistics for the 100 databases in the testbed.

Resource Description	Documents Per Database			Bytes Per Database		
	Minimum	Average	Maximum	Minimum	Average	Maximum
Actual	752	10,782	39,723	28,070,646	33,365,514	41,796,822
Learned	300	300	300	229,915	2,701,449	15,917,750

5.1 Data

The TREC-123 database described above (Section 4.1) was divided into 100 smaller databases of roughly equal size (about 33 megabytes each), but varying in the number of documents they contained (Table 6). Each database contained documents from a single source, ordered as they were found on the TREC CDs; hence documents in a database were also usually from similar timeframes. CD 1 contributed 37 databases, CD 2 contributed 27 databases, and CD 3 contributed 36 databases.

Queries were based on TREC topics 51-150 [17]. We used query sets INQ001 and INQ026, both created by the UMass CIIR as part of its participation in TREC-2 and Tipster 24 month evaluations [6]. Queries in these query sets are long, complex, and have undergone automatic query expansion.

The relevance assessments were the standard TREC relevance assessments supplied by the U.S. National Institute for Standards and Technology [17].

5.2 Resource Descriptions

Each experiment used 100 resource descriptions (one per database). Each resource description consisted of a list of terms and their document frequencies (*df*), as in previous experiments. Terms on a stopword list of 418 common or closed-class words were discarded. The remaining terms were stemmed with KStem [21].

5.3 Metrics

Several methods have been proposed for evaluating resource selection algorithms [16; 14; 7; 23; 11]. The most appropriate for our needs is a recall-oriented metric called $\hat{R}$ [11; 10] that measures the percentage of relevant documents contained in the n top-ranked databases.[4] $\hat{R}$ is defined as:

$$\hat{R}(n) \; = \; \frac{\Sigma_{i=1}^{n} R_i}{\Sigma_{i=1}^{N} R_i} \tag{3}$$

where n is the number of databases searched, N is the total number of databases, and R_i is the number of relevant documents contained by the i'th database.

$\hat{R}$ is a cumulative metric; $\hat{R}(2) \leq \hat{R}(3)$, because searching the top 3 databases always returns at least as many relevant documents as searching just the top 2 databases.

$\hat{R}$ is a desirable metric when the accuracy of the database ranking algorithm is to be measured independently of other system components, and when the goal is to rank databases containing many relevant documents ahead of databases containing few relevant documents.

[4]The metric called $\hat{R}$ was called R in [23]. We use the more recent and more widely known name, $\hat{R}$, in this paper.

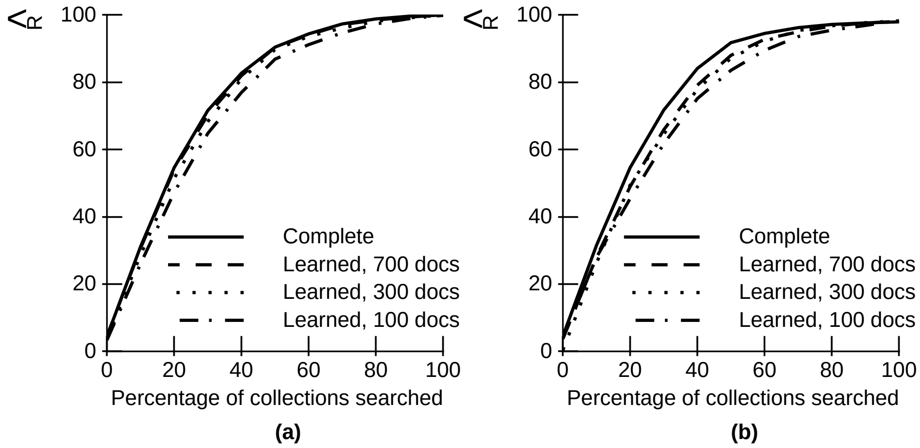

Fig. 5. Measures of collection ranking accuracy using resource descriptions of varying accuracy. **(a)** Topics 51-100 (TREC query set INQ026). **(b)** Topics 101-150 (TREC query set INQ001). (Four documents examined per query. TREC volumes 1, 2, and 3.)

5.4 Parameter Settings

The experiments in Section 4 suggested that any relatively small sample size is effective, and that different choices produce only small variations in results. We chose a sample size of four (4 documents per query), to be consistent with the baseline results in previous experiments. Query terms were chosen randomly from the learned resource description, as in the baseline experiments.

It was unclear from the experiments in Section 4 when enough samples had been taken. We chose to build resource descriptions from samples of 100 documents (about 25 queries), 300 documents (about 75 queries), and 700 documents (about 175 queries) from each database, in order to cover the space of "reasonable" numbers of samples. If results varied dramatically, we were prepared to conduct additional experiments.

The collection ranking algorithm itself forces us to set one additional parameter. The collection ranking algorithm normalizes term frequency statistics ($df_{i,j}$) using the length, in words, of the collection (cw_j) [7]. However, we do not know how to estimate collection size with query-based sampling. In our experiments, term frequency information (df) was normalized using the length, in words, of the set of sampled documents used to construct the resource description.

5.5 Experimental Results

The experimental results are summarized in the two graphs in Figure 5 (one per query set). The baseline in each graph is the curve showing results with the actual resource description ("complete resource descriptions"). This is the best result that the collection ranking algorithm can produce when given a complete description for each collection.

Our interest is in the difference between what is achieved with complete information and what is achieved with incomplete information. Both graphs show only a small loss of effectiveness when resource descriptions are based on 700 documents.

Losses grow as less information is used, but the loss is small compared to the information reduction. Accuracy at "low recall", i.e., when only 10-20% of the databases are searched, is quite good, even when resource descriptions are based on only 100 documents.

These results are consistent with the results presented in Section 4. The earlier experiments showed that term rankings in the learned and actual resource descriptions were highly correlated after examining 100-300 documents.

These experimental results also demonstrate that it is possible to rank databases without knowing their sizes. The size of the pool of documents sampled from a database was an effective surrogate for actual database size in these tests. Our testing did not reveal whether this result is general, a characteristic of the CORI database selection algorithm, or a quirk due to the 100 database testbed. The distribution of database sizes in the testbed ranged from 752 documents to 39,723 documents, and from 28 megabytes to 42 megabytes (Table 6). A more thorough study of this characteristic would require testbeds with a wider variety of size distributions.

6. EXPERIMENTAL RESULTS: RETRIEVAL ACCURACY

The experiments described in the previous section demonstrate that resource descriptions learned with query-based sampling enable accurate resource ranking. Accurate resource ranking is generally viewed as a prerequisite to accurate document retrieval, but it is not a guarantee. The final document ranking depends upon how results from different databases are merged, which can be influenced by the quality of the resource descriptions for each database.

A third group of experiments investigated the accuracy of document retrieval in the presence of learned resource descriptions. The experimental method was based on comparing the accuracy of the final *document* rankings produced by a distributed IR system when it uses complete and learned resource descriptions to make decisions about where to search. Databases were ranked, selected, and searched, and results were merged into a final document ranking by the INQUERY IR system's default database ranking and result merging algorithms [4].

6.1 Data

The data consisted of the same 100 databases that were used to test database selection accuracy. Section 5.1 provides details.

6.2 Resource Descriptions

Each database was described by a learned resource description created from a sample of 300 documents, as done in other experiments (4 documents per query, query terms chosen randomly from the learned resource description). A sample size of 300 documents was chosen because in previous experiments it provided reasonably accurate resource descriptions at a relatively low cost (about 75 queries per database).

Each of the 100 resource descriptions (one per database) consisted of a list of terms and their document frequencies (*df*), as in previous experiments. Terms on a stopword list of 418 common or closed-class words were discarded. The remaining terms were stemmed with KStem [21].

6.3 Metrics

The effectiveness of archival search systems is often measured either by Precision at specified document ranks, or by Precision at specified Recall points. Precision at specified Recall points (e.g., "11-point Recall") was the standard for many years, because it normalizes results based on the number of relevant documents; results for "easy" queries (many relevant documents) and "hard" queries (few relevant documents) are more comparable. However, when there are many relevant documents, as can be the case with large databases, Precision at specified Recall points focuses attention on results that are irrelevant to many search patrons (e.g., at rank 50 and 100).

Precision at specified document ranks is often used when the emphasis is on the results a person would see in the first few screens of an interactive system. Precision at rank n is defined as:

$$P(n) \;=\; \frac{R_r}{n} \qquad (4)$$

where R_r is the number of retrieved relevant documents in ranks 1 through n.

Precision in our experiments was measured at ranks 5, 10, 15, 20, and 30 documents, as is common in experiments with TREC data [17; 18]. These values indicate the accuracy that would be observed at various points on the first two or three screens of an interactive system.

6.4 Parameter Settings

All INQUERY system parameters were set to their default values for this experiment. The only choices made for these experiments were decisions about how many databases to search, and how many documents to return from each database.

INQUERY searched the 10 databases ranked most highly for the query by its database selection algorithm. The number 10 was chosen because it has been used in other recent research on distributed search with the INQUERY system [39; 40]. The database selection algorithm ranked databases using either the learned resource descriptions or the complete resource descriptions, as determined by the experimenter.

Each searched database returned its most highly ranked 30 documents. The number 30 was chosen because Precision was measured up to, but not beyond, rank 30.

The returned documents (10 × 30) were merged, using INQUERY's default algorithm for merging "multi-database" search results. The algorithm for merging results from multiple searches is based on estimating an *idf*−normalized score D' for a document with a score of D in a collection with a score of C as:

$$D_s \;=\; (D - D_{min})/(D_{max} - D_{min}) \qquad (5)$$

$$C_s \;=\; (C - C_{min})/(C_{max} - C_{min}) \qquad (6)$$

$$D' \;=\; (D_s + 0.4 \times D_s \times C_s)/1.4 \qquad (7)$$

where D_{max} and D_{min} are the maximum and minimum possible scores any document in that database could obtain for the particular query, and C_{max} and C_{min} are

Table 7. Precision of a search system using complete and learned resource descriptions for database selection and result merging. TREC volumes 1, 2, and 3, divided into 100 databases. 10 databases were searched for each query.

	Topics 51-100 (query set INQ026)		Topics 101-150 (query set INQ001)	
Document Rank	Complete Resource Descriptions	Learned Resource Descriptions	Complete Resource Descriptions	Learned Resource Descriptions
5	0.5960	0.6080 (+2.0%)	0.5920	0.5560 (−6.1%)
10	0.5720	0.5960 (+4.1%)	0.5640	0.5580 (−1.1%)
15	0.5613	0.5893 (+5.0%)	0.5547	0.5360 (−3.3%)
20	0.5480	0.5880 (+7.2%)	0.5450	0.5230 (−4.0%)
30	0.5240	0.5533 (+5.6%)	0.5107	0.5040 (−1.3%)

the maximum and minimum scores any collection could obtain for the particular query. This scaling compensates for the fact that while a system like INQUERY can in theory produce document scores in the range $[0, 1]$, in practice the *tf.idf* algorithm makes it mathematically impossible for a document to have a score outside a relatively narrow range. D_{min} and C_{min} are usually 0.4, and D_{max} and C_{max} are usually about 0.6. Their exact values are query-dependent, and are calculated by setting the tf component of the $tf.idf$ formula to 0.0 and 1.0 for every query term [4].

Although the theoretical justification for this heuristic normalization is weak, it has been effective in practice [1; 2; 4; 22] and has been used in INQUERY since 1995.

6.5 Experimental Results

Databases were ranked with either an index of complete resource descriptions (baseline condition) or an index of learned resource descriptions (test condition). The top 10 databases were searched; each returned 30 documents. The result lists returned by each database were merged to produce a final result list of 30 documents. (The scores used to rank the databases determined the value of C in Equation 6.) Precision was measured at ranks 5, 10, 15, 20, and 30 documents. The experimental results are summarized in Table 7.

The experimental results indicate that distributed, or "multi-database", retrieval is as effective with learned resource descriptions as it is with complete resource descriptions. Precision with one query set (INQ026, topics 51-100) averaged 4.8% higher using learned descriptions, with a range of 2.0 to 7.2%. Precision with the other query set (INQ001, topics 101-150) averaged 3.2% lower using learned descriptions, with a range of −1.1% to −6.1%. Both the improvement and the loss were too small for a person to notice.

These experimental results extend the results of Section 5, which indicated that using learned resource descriptions to rank collections introduced only a small amount of error into the ranking process. One might argue that the amount of error was too small to cause a noticeable change in search results, but there was no evidence to support that argument. These results demonstrate that the small errors introduced by learned resource descriptions do not noticeably reduce the accuracy of the final search results.

The accuracy of the document ranking depends also on merging results from

Table 8. A comparison of the 50 most frequent terms, as measured by document frequency, in a text database and in a learned resource description constructed for that database. 1988 Wall Street Journal database. 300 documents examined (4 documents per query).

Rank	Actual	Learned	Rank	Actual	Learned
1	million	company	26	group	york
2	new	million	27	concern	operate
3	company	new	28	exchange	stock
4	make	make	29	high	hold
5	corp	corp	30	sale	executive
6	base	base	31	operate	close
7	business	business	32	price	group
8	two	market	33	unit	international
9	trade	co	34	increase	increase
10	co	report	35	hold	general
11	market	president	36	billion	time
12	close	two	37	end	exchange
13	president	billion	38	yesterday	sale
14	stock	say	39	product	change
15	early	concern	40	interest	result
16	wsj	early	41	offer	service
17	month	share	42	recent	manage
18	u.s.	unit	43	america	made
19	staff	plan	44	manage	work
20	report	expect	45	current	america
21	plan	three	46	part	buy
22	say	trade	47	three	national
23	time	interest	48	bank	official
24	expect	product	49	executive	end
25	york	month	50	call	director

different collections accurately. The experimental results indicate that learned resource descriptions support this activity as well. This result is important because INQUERY's result merging algorithm estimates a normalized document score as a function of the collection's score and the document's score with respect to its collection. The results indicate that not only are collections *ranked* appropriately using learned descriptions, but that the *scores* used to rank them are highly correlated with the scores produced with complete resource descriptions. This is further evidence that query-based sampling produces very accurate resource descriptions.

7. A PEEK INSIDE: SUMMARIZING DATABASE CONTENTS

Our interest is primarily in an automatic method of learning resource descriptions that are sufficiently accurate and detailed for use by automatic database selection algorithms. However, a resource description can also be used to indicate to a person the general nature of a given text database.

The simplest method is to display the terms that occur frequently and are not stopwords. This method can be effective just because the database is, in some

Table 9. The 18 topics covered by the Combined Health Information database.

AIDS education	Disease Prevention/Health Promotion
Alzheimer's Disease	Epilepsy Education and Prevention
Arthritis; Musculoskeletal and Skin Diseases	Health Promotion and Education
Cancer Patient Education	Kidney and Urologic Diseases
Cancer Prevention and Control	Maternal and Child Health
Complementary and Alternative Medicine	Medical Genetics and Rare Disorders
Deafness and Communication Disorders	Oral Health
Diabetes	Prenatal Smoking Cessation
Digestive Diseases	Weight Control

sense, guaranteed to be about the words that occur most often. For example, the list of the top 50 words found by sampling the 1988 Wall Street Journal (Table 8) contains words such as "market", "interest", "trade", "million", "stock", and "exchange", which are indeed suggestive of the overall subject of the database.

Table 8 also compares the top 50 words in the learned resource description with the top 50 words in the database. It demonstrates that after 300 documents the learned resource description is reasonably representative of the vocabulary in the target text database *and* it is representative of the relative importance (ranks) of the terms; in this example, there is 76% agreement on the top 50 terms after seeing just 300 documents.

Controlled experiments are essential to understanding the characteristics of a new technique, but less controlled, 'real world' experiments can also be revealing. A simple database sampling system was built to test the algorithm on databases found on the Web. The program was tested initially on the Microsoft Customer Support Database at a time when we understood less about the most effective parameter settings. Accurate resource descriptions were learned, but at the cost of examining many documents [5].

We chose for this paper to reproduce the earlier experiment on a more easily accessible Web database, using sampling parameters that were consistent with parameter settings described elsewhere in this paper. The Combined Health Information Database [29], which is published by several health-related agencies of the U.S. government (National Institutes of Health, Centers for Disease Control and Prevention, and Health Resources and Services Administration) was selected. The database contains health-related information on 18 topics, which are summarized in Table 9.

The initial query term was chosen randomly from the TREC-123 database. Subsequent query terms were chosen randomly from the resource description that was being learned. Four documents were examined per query. The experiment was ended after 300 documents were examined. Terms in the resource description were sorted by collection term frequency (*ctf*), and the top 100 terms were displayed. The results are shown in Table 10.

One can see easily that the database contains documents about health-related topics. Terms such as "hiv", "aids", "health", "prevention", "risk", "cdc", "transmission", "medical", "disease", "virus", "drug" and "immunodeficiency" show up high in the list.

Table 10. The top 100 words found by sampling the U.S. National Institutes of Health (NIH) Combined Health Information database. Terms are ranked by collection term frequency (*ctf*) in the sampled documents. 300 documents were examined (4 documents per query).

Term	*ctf*	*df*	Term	*ctf*	*df*	Term	*ctf*	*df*
hiv	1931	254	lg	296	296	control	168	86
aids	1561	291	mj	296	296	department	166	90
health	1161	237	ve	296	296	notes	163	163
prevention	666	195	verification	296	296	nt	163	163
education	534	293	yr	296	296	state	160	64
information	439	184	code	295	292	program	158	80
persons	393	174	english	294	280	video	148	32
number	384	296	ac	292	292	acquired	144	140
author	370	294	physical	282	267	deficiency	139	137
material	361	293	print	281	257	research	138	74
document	356	296	treatment	280	127	syndrome	138	138
human	355	212	cn	279	279	factors	137	95
source	346	296	corporate	279	279	drugs	132	68
report	328	89	description	278	266	united	132	80
accession	323	296	pd	266	266	centers	131	67
public	323	156	programs	264	112	world	131	55
update	317	296	organizations	261	126	box	130	121
community	313	107	positive	254	150	cdc	128	75
language	310	296	care	248	83	children	122	45
services	310	129	virus	246	192	patient	119	42
descriptors	308	296	disease	241	120	center	118	67
format	308	296	service	241	133	people	117	68
major	305	296	discusses	226	152	agencies	112	65
national	304	132	provides	226	154	government	112	63
transmission	304	114	professionals	217	167	nations	112	41
published	303	296	medical	212	117	describes	110	87
audience	302	293	immunodeficiency	193	180	organization	109	51
availability	302	293	drug	190	74	sex	108	60
abstract	299	296	risk	185	99	std	107	50
date	299	296	issues	182	96	counseling	106	50
chid	297	296	brochure	180	54	refs	103	103
subfile	297	296	immune	179	144	surveillance	103	35
ab	296	296	examines	173	132			
fm	296	296	women	171	61			

Several of the most frequent words appear to indicate little about the database contents, such as "update", "published", "format", and "abstract". These terms could have been removed by using a larger stopword list. However, in general it is unclear which words in a multi-database environment should be considered stopwords, since words that are unimportant in one database may be content words for others.

This particular resource description was based on a very simple approach to

Table 11. The top 50 words found by sampling TREC-123 Terms are ranked by document frequency (*df*) in the sampled documents. 500 documents were examined (4 documents per query).

Term	*ctf*	*df*	Term	*ctf*	*df*	Term	*ctf*	*df*
two	460	159	say	228	94	plan	163	79
new	553	158	made	246	94	million	199	79
time	437	135	result	249	93	end	556	78
three	269	128	information	706	93	allow	190	78
system	1609	122	develop	525	91	month	222	78
base	421	115	accord	322	91	set	278	77
high	585	115	service	468	90	manage	302	77
make	254	115	general	479	87	national	209	77
state	446	114	call	432	86	change	311	76
report	336	104	number	292	86	long	153	76
product	549	103	company	304	85	problem	170	75
part	371	101	show	223	83	line	271	75
group	513	101	president	339	82	close	207	75
work	256	98	require	432	80	increase	173	75
relate	269	96	people	181	79	second	882	75
operate	396	95	support	283	79	order	236	74
follow	262	94	data	608	79			

tokenizing, case conversion, and stopword removal. For example, all terms were converted to lower case, hence it does not distinguish among terms that differ only in case, such as "aids" and "AIDS". This distinction is important in this particular database, and illustrates some of the issues that a 'real world' system must address. Appropriate lexical processing is not necessarily a major barrier, but accuracy in 'real world' settings probably requires that it be addressed.

The Wall Street Journal and Combined Health Information databases are homogeneous to varying degrees, which may make it easier to summarize their contents with brief lists of frequent terms. This summarization technique may be less effective with larger, heterogeneous databases such as TREC-123. The top 50 words in the TREC-123 database (Table 11) provide some evidence that the database contains documents about U.S. national and business news, but it would be difficult to draw firm conclusions about the database contents from this list of words alone.

Although simple word lists are effective for summarizing database contents in some situations, they are not necessarily the most effective techniques. Frequent phrases and common relationships can be better.

Indeed, one consequence of the sampling approach to creating learned resource descriptions is that it makes more powerful summarizations possible. The sampling process is not restricted just to word lists and frequency tables, nor is it restricted to just the information the database chooses to provide. Instead, it has a set of several hundred documents from which to mine frequent phrases, names, dates, relationships, and other interesting information. This information is likely to enable construction of more powerful and more informative summaries than is possible with the simple resource descriptions used by cooperative methods.

8. OTHER USES

The set of documents sampled from a single database reflects the contents of that database. One use of these documents is to build a resource description for a single database, as described above. However, other uses are possible.

One potential use is in a query expansion database. Recent research showed that query expansion significantly improves the accuracy of database selection [39]. The state-of-the-art in query expansion is based upon analyzing the searched corpus for co-occurrence patterns, but what database(s) should be used when the task is database selection? This question has been unanswered.

If the documents sampled from each database were combined into a query expansion corpus, the result would be a set of documents that reflects the contents and word co-occurrence patterns across *all* of the available databases. It would require little additional effort for a database selection service to create a query expansion database in this manner.

Co-occurrence-based query expansion can be viewed as a form of data mining. Other forms of data mining could also be applied to the set of documents sampled from all databases. For example, frequent concepts, names, or relationships might be extracted and used in a visualization interface.

The ability to construct a single database that acts as a surrogate for a set of databases is significant, because it could be a way of rapidly porting many familiar Information Retrieval tools to environments containing many databases. Although there are many unanswered questions, this appears to be a promising direction for future research.

9. CONCLUSIONS

Our hypothesis was that an accurate description of a text database can be constructed from documents obtained by running queries on the database. Preliminary experiments [5] supported the hypothesis, but were not conclusive. The experiments presented in this paper test the hypothesis extensively, from multiple perspectives, and confirm the hypothesis. The resource descriptions created by *query-based sampling* are sufficiently similar to resource descriptions created from complete information that it makes little difference which is used for database selection.

Query-based sampling avoids many of the limitations of cooperative protocols such as STARTS. Query-based sampling can be applied to older ('legacy') databases and to databases that have no incentive to cooperate. It is not as easily defeated by intentional misrepresentation. It also avoids the problem of needing to reconcile the differing tokenizing, stopword lists, word stemming, case conversion, name recognition, and other representational choices made in each database. These representation problem are perhaps the most serious weakness of cooperative protocols, because they exist even when all parties *intend* to cooperate.

The experimental results also demonstrate that the cost of query-based sampling, as measured by the number of queries and documents required, is reasonably low, and that query-based sampling is robust with respect to variations in parameter settings.

Finally, and perhaps most importantly, the experiments described in this paper demonstrate that a fairly small partial description of a resource can be as effective for distributed search as a complete description of that resource. This result suggests that much of the information exchanged by cooperative protocols is unnecessary, and that communications costs could be reduced significantly without affecting results.

The demonstrated effectiveness of partial resource descriptions also raises questions about which terms are necessary for describing text collections. Query-based sampling identifies terms across a wide frequency range, but it necessarily favors the frequent, non-stopword terms in a database. Luhn suggested that terms in the middle of the frequency range would be best for describing documents [24]. It is an open question whether terms in the middle of the frequency range would be best for describing collections, too.

Several other open questions remain, among them whether the number of documents in a database can be estimated with query-based sampling. We have shown that this information may not be required for database selection, but it is nonetheless desirable information. It is also an open question how many documents must be sampled from a resource to obtain a description of a desired accuracy, although 300-500 documents appears to be very effective across a range of database sizes.

The work reported here can be extended in several directions, to provide a more complete environment for searching and browsing among many databases. For example, the documents obtained by query-based sampling could be used to provide query expansion for database selection, or to drive a summarization or visualization interface showing the range of information available in a multi-database environment. More generally, the ability to construct a single database that acts as a surrogate for a large set of databases offers many possibilities for interesting research.

ACKNOWLEDGMENTS

We thank Aiqun Du for her work in the early stages of the research reported here. We also thank the reviewers for their many helpful suggestions, and a reviewer for the SIGIR conference for suggesting the experiments in Section 4.5.4.

This material is based on work supported in part by the Library of Congress and Department of Commerce under cooperative agreement number EEC-9209623, and in part by NSF grants IIS-9873009, EIA-9983253, and EIA-9983215. Any opinions, findings, conclusions or recommendations expressed in this material are the authors', and do not necessarily reflect those of the sponsors.

REFERENCES

[1] J. Allan, L. Ballesteros, J. P. Callan, W. B. Croft, and Z. Lu. Recent experiments with INQUERY. In D. Harman, editor, *Proceedings of the Fourth Text REtrieval Conference (TREC-4)*. National Institute of Standards and Technology Special Publication, 1996.

[2] J. Allan, J. Callan, M. Sanderson, J. Xu, and S. Wegman. INQUERY and TREC-7. In D. K. Harman and E. M. Voorhees, editors, *The Seventh Text REtrieval Conference (TREC-7)*, pages 201–216. National Institute of Standards and Technology, Special Publication 500-242, 1999.

[3] C. Baumgarten. A probabilistic model for distributed information retrieval. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 258–266. ACM Press, 1997.

[4] J. Callan. Distributed information retrieval. In W.B. Croft, editor, *Advances in information retrieval*, chapter 5, pages 127–150. Kluwer Academic Publishers, 2000.

[5] J. Callan, M. Connell, and A. Du. Automatic discovery of language models for text databases. In *Proceedings of the ACM-SIGMOD International Conference on Management of Data*, pages 479–490. ACM, 1999.

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

[7] J. P. Callan, Z. Lu, and W. B. Croft. Searching distributed collections with inference networks. In *Proceedings of the Eighteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–28, Seattle, 1995. ACM.

[8] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A distributed anonymous information storage and retrieval system. In *ICSI Workshop on Design Issues in Anonymity and Unobservability*, Berkeley, California, July 25-26 2000.

[9] N. Craswell, P. Bailey, and D. Hawking. Server selection on the World Wide Web. In *Proceedings of the Fifth ACM Conference on Digital Libraries*, pages 37–46. ACM, 2000.

[10] J. French, A. Powell, J. Callan, C. Viles, T. Emmitt, K. Prey, and Y. Mou. Comparing the performance of database selection algorithms. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 238–245. ACM, 1999.

[11] J.C. French, A.L. Powell, C.L. Viles, T. Emmitt, and K.J. Prey. Evaluating database selection techniques: A testbed and experiment. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1998.

[12] N. Fuhr. A decision-theoretic approach to database selection in networked IR. *ACM Transactions on Information Systems*, 17(3):229–249, 1999.

[13] L. Gravano, K. Chang, H. García-Molina, and A. Paepcke. STARTS Stanford proposal for Internet meta-searching. In *Proceedings of the ACM-SIGMOD International Conference on Management of Data*, 1997.

[14] L. Gravano and H. García-Molina. Generalizing GLOSS to vector-space databases and broker hierarchies. In *Proceedings of the 21st International Conference on Very Large Databases (VLDB)*, pages 78–89, 1995.

[15] L. Gravano, H. García-Molina, and A. Tomasic. The effectiveness of GLOSS for the text database discovery problem. In *Proceedings of the ACM-SIGMOD International Conference on Management of Data*, pages 126–137. ACM, 1994. SIGMOD Record 23(2).

[16] L. Gravano, H. García-Molina, and A. Tomasic. Precision and recall of GLOSS estimators for database discovery. Technical Report STAN-CS-TN-94-10, Computer Science Department, Stanford University, 1994.

[17] D. Harman, editor. *The Second Text REtrieval Conference (TREC2)*. National Institute of Standards and Technology Special Publication 500-215, Gaithersburg, MD, 1994.

[18] D. Harman, editor. *Proceedings of the Third Text REtrieval Conference (TREC-3)*. National Institute of Standards and Technology Special Publication 500-225, Gaithersburg, MD, 1995.

[19] D. Hawking and P. Thistlewaite. Methods for information server selection. *ACM Transactions on Information Systems*, 17(1):40–76, 1999.

[20] H. S. Heaps. *Information Retrieval: Computational and Theoretical Aspects*. Academic Press, New York, 1978.

[21] R. Krovetz. *Word Sense Disambiguation for Large Text Databases*. PhD thesis, University of Massachusetts at Amherst, 1995.

[22] L. Larkey, M. Connell, and J. Callan. Collection selection and results merging with topically organized U.S. patents and TREC data. In *Proceedings of the 9th International Conference on Information and Knowledge Management (CIKM)*, pages 282–289. ACM, 2000.

[23] Z. Lu, J.P. Callan, and W.B. Croft. Measures in collection ranking evaluation. Technical Report 96-39, Department of Computer Science, University of Massachusetts, 1996.

[24] H.P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research*, 2:159–165, 1958.

[25] R. S. Marcus. An experimental comparison of the effectiveness of computers and humans as search intermediaries. *Journal of the American Society for Information Science*, 34:381–404, 1983.

[26] W. Meng, K. L. Liu, C. T. Yu, X. Wang, Y. Chang, and N. Rishe. Determining text databases to search in the Internet. In A. Gupta, O. Shmueli, and J. Widom, editors, *Proceedings of 24th International Conference on Very Large Data Bases*, pages 14–25, New York, 1998. Morgan Kaufmann.

[27] W. Meng, K. L. Liu, C. T. Yu, W. Wu, and N. Rishe. Estimating the usefulness of search engines. In *Proceedings of the 15th International Conference on Data Engineering*, pages 146–153, Sydney, 1999. IEEE Computer Society Press.

[28] M.J. Moroney. *Facts from figures*. Penguin, Baltimore, 1951.

[29] National Institutes of Health, editor. *Combined Health Information Database*. http://chid.nih.gov/. National Institutes of Health, Washington, D.C., 1999.

[30] National Information Standards Organization. *Information Retrieval (Z39.50): Application Services Definition and Protocol Specification (ANSI/NISO Z39.50-1995)*. NISO Press, Bethesda, MD, 1995.

[31] A. Powell, J. French, J. Callan, M. Connell, and C. Viles. The impact of database selection on distributed searching. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 232–239. ACM, 2000.

[32] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical recipies in C: The art of scientific computing*. Cambridge University Press, 1992.

[33] H. R. Turtle. *Inference Networks for Document Retrieval*. PhD thesis, University of Massachusetts at Amherst, 1990.

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

[35] C. L. Viles and J. C. French. Dissemination of collection wide information in a distributed Information Retrieval system. In *Proceedings of the Eighteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 12–20, Seattle, 1995. ACM.

[36] E. M. Voorhees, N. K. Gupta, and B. Johnson-Laird. Learning collection fusion strategies. In *Proceedings of the Eighteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 172–179, Seattle, 1995. ACM.

[37] E.M. Voorhees and R.M. Tong. Multiple search engines in database merging. In *Proceedings of the 2nd ACM International Conference on Digital Libraries*, Philadelphia, 1997. ACM.

[38] R. Weiss, B. Velez, M.A. Sheldon, C. Nemprempre, P. Szilagyi, A. Duda, and D.K. Gifford. HyPursuit: A hierarchical network search engine that exploits content-link hypertext clustering. In *Proceedings of the Seventh ACM Conference on Hypertext*, pages 180–193, Washington, D.C., 1996. ACM.

[39] J. Xu and J. Callan. Effective retrieval of distributed collections. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 112–120, Melbourne, 1998. ACM.

[40] J. Xu and W.B. Croft. Cluster-based language models for distributed retrieval. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 254–261, Berkeley, 1999. ACM.

[41] B. Yuwono and D. L. Lee. Search and ranking algorithms for locating resources on the World Wide Web. In *Proceedings of the 12th International Conference on Data Engineering*, pages 164–171, New Orleans, 1996.

[42] B. Yuwono and D. L. Lee. Server ranking for distributed text retrieval systems on the Internet. In *Proceedings of the Fifth International Conference on Database Systems for Advanced Applications (DASFAA)*, pages 41–49, Melbourne, 1997.

[43] G. K. Zipf. *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley, Reading, MA, 1949.