Chapter 3
Retrieval Evaluation

3.1 Introduction

Before the final implementation of an information retrieval system, an evaluation of the system is usually carried out. The type of evaluation to be considered depends on the objectives of the retrieval system. Clearly, any software system has to provide the functionality it was conceived for. Thus, the first type of evaluation which should be considered is a functional analysis in which the specified system functionalities are tested one by one. Such an analysis should also include an error analysis phase in which, instead of looking for functionalities, one behaves erratically trying to make the system fail. It is a simple procedure which can be quite useful for catching programming errors. Given that the system has passed the functional analysis phase, one should proceed to evaluate the performance of the system.

The most common measures of system performance are time and space. The shorter the response time, the smaller the space used, the better the system is considered to be. There is an inherent tradeoff between space complexity and time complexity which frequently allows trading one for the other. In Chapter 8 we discuss this issue in detail.

In a system designed for providing data retrieval, the response time and the space required are usually the metrics of most interest and the ones normally adopted for evaluating the system. In this case, we look for the performance of the indexing structures (which are in place to accelerate the search), the interaction with the operating system, the delays in communication channels, and the overheads introduced by the many software layers which are usually present. We refer to such a form of evaluation simply as *performance evaluation.*

In a system designed for providing information retrieval, other metrics, besides time and space, are also of interest. In fact, since the user query request is inherently vague, the retrieved documents are not exact answers and have to be ranked according to their relevance to the query. Such relevance ranking introduces a component which is not present in data retrieval systems and which plays a central role in information retrieval. Thus, information retrieval systems require the evaluation of how precise is the answer set. This type of evaluation is referred to as *retrieval performance evaluation.*

In this chapter, we discuss retrieval performance evaluation for information retrieval systems. Such an evaluation is usually based on a test reference collection and on an evaluation measure. The test reference collection consists of a collection of documents, a set of example information requests, and a set of relevant documents (provided by specialists) for each example information request. Given a retrieval strategy S, the evaluation measure quantifies (for each example information request) the *similarity* between the set of documents retrieved by S and the set of relevant documents provided by the specialists. This provides an estimation of the *goodness* of the retrieval strategy S.

In our discussion, we first cover the two most used retrieval evaluation measures: recall and precision. We also cover alternative evaluation measures such as the E measure, the harmonic mean, satisfaction, frustration, etc. Following that, we cover four test reference collections namely, TIPSTER/TREC, CACM, CISI, and Cystic Fibrosis.

3.2 Retrieval Performance Evaluation

When considering retrieval performance evaluation, we should first consider the retrieval task that is to be evaluated. For instance, the retrieval task could consist simply of a query processed in batch mode (i.e., the user submits a query and receives an answer back) or of a whole interactive session (i.e., the user specifies his information need through a series of interactive steps with the system). Further, the retrieval task could also comprise a combination of these two strategies. Batch and interactive query tasks are quite distinct processes and thus their evaluations are also distinct. In fact, in an interactive session, user effort, characteristics of the interface design, guidance provided by the system, and duration of the session are critical aspects which should be observed and measured. In a batch session, none of these aspects is nearly as important as the quality of the answer set generated.

Besides the nature of the query request, one has also to consider the setting where the evaluation will take place and the type of interface used. Regarding the setting, evaluation of experiments performed in a laboratory might be quite distinct from evaluation of experiments carried out in a real life situation. Regarding the type of interface, while early bibliographic systems (which still dominate the commercial market as discussed in Chapter 14) present the user with interfaces which normally operate in batch mode, newer systems (which are been popularized by the high quality graphic displays available nowadays) usually present the user with complex interfaces which often operate interactively.

Retrieval performance evaluation in the early days of computer-based information retrieval systems focused primarily on laboratory experiments designed for batch interfaces. In the 1990s, a lot more attention has been paid to the evaluation of real life experiments. Despite this tendency, laboratory experimentation is still dominant. Two main reasons are the repeatability and the scalability provided by the closed setting of a laboratory.

In this book, we focus mainly on experiments performed in laboratories. In this chapter in particular we discuss solely the evaluation of systems which operate in batch mode. Evaluation of systems which operate interactively is briefly discussed in Chapter 10.

3.2.1 Recall and Precision

Consider an example information request I (of a test reference collection) and its set R of relevant documents. Let $|R|$ be the number of documents in this set. Assume that a given retrieval strategy (which is being evaluated) processes the information request I and generates a document answer set A. Let $|A|$ be the number of documents in this set. Further, let $|Ra|$ be the number of documents in the intersection of the sets R and A. Figure 3.1 illustrates these sets.

The recall and precision measures are defined as follows.

- **Recall** is the fraction of the relevant documents (the set R) which has been retrieved i.e.,

$$Recall = \frac{|Ra|}{|R|}$$

- **Precision** is the fraction of the retrieved documents (the set A) which is relevant i.e.,

$$Precision = \frac{|Ra|}{|A|}$$

Recall and precision, as defined above, assume that all the documents in the answer set A have been examined (or seen). However, the user is not usually presented with all the documents in the answer set A at once. Instead, the

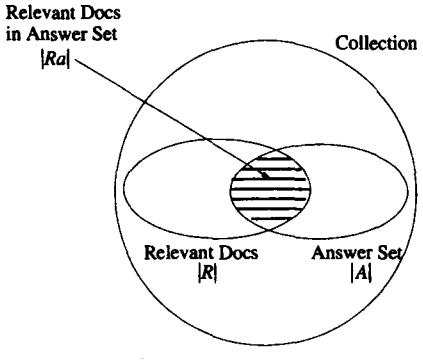

Figure 3.1 Precision and recall for a given example information request.

documents in A are first sorted according to a degree of relevance (i.e., a ranking is generated). The user then examines this ranked list starting from the top document. In this situation, the recall and precision measures vary as the user proceeds with his examination of the answer set A. Thus, proper evaluation requires plotting a precision versus recall curve as follows.

As before, consider a reference collection and its set of example information requests. Let us focus on a given example information request for which a query q is formulated. Assume that a set R_q containing the relevant documents for q has been defined. Without loss of generality, assume further that the set R_q is composed of the following documents

$$R_q = \{d_3, d_5, d_9, d_{25}, d_{39}, d_{44}, d_{56}, d_{71}, d_{89}, d_{123}\} \quad \textbf{(3.1)}$$

Thus, according to a group of specialists, there are ten documents which are relevant to the query q.

Consider now a new retrieval algorithm which has just been designed. Assume that this algorithm returns, for the query q, a ranking of the documents in the answer set as follows.

Ranking for query q:

1. d_{123} •	6. d_9 •	11. d_{38}
2. d_{84}	7. d_{511}	12. d_{48}
3. d_{56} •	8. d_{129}	13. d_{250}
4. d_6	9. d_{187}	14. d_{113}
5. d_8	10. d_{25} •	15. d_3 •

The documents that are relevant to the query q are marked with a bullet after the document number. If we examine this ranking, starting from the top document, we observe the following points. First, the document d_{123} which is ranked as number 1 is relevant. Further, this document corresponds to 10% of all the relevant documents in the set R_q. Thus, we say that we have a precision of 100% at 10% recall. Second, the document d_{56} which is ranked as number 3 is the next relevant document. At this point, we say that we have a precision of roughly 66% (two documents out of three are relevant) at 20% recall (two of the ten relevant documents have been seen). Third, if we proceed with our examination of the ranking generated we can plot a curve of precision versus recall as illustrated in Figure 3.2. The precision at levels of recall higher than 50% drops to 0 because not all relevant documents have been retrieved. This precision versus recall curve is usually based on 11 (instead of ten) *standard* recall levels which are 0%, 10%, 20%, ..., 100%. For the recall level 0%, the precision is obtained through an interpolation procedure as detailed below.

In the above example, the precision and recall figures are for a single query. Usually, however, retrieval algorithms are evaluated by running them for several distinct queries. In this case, for each query a distinct precision versus recall curve is generated. To evaluate the retrieval performance of an algorithm over

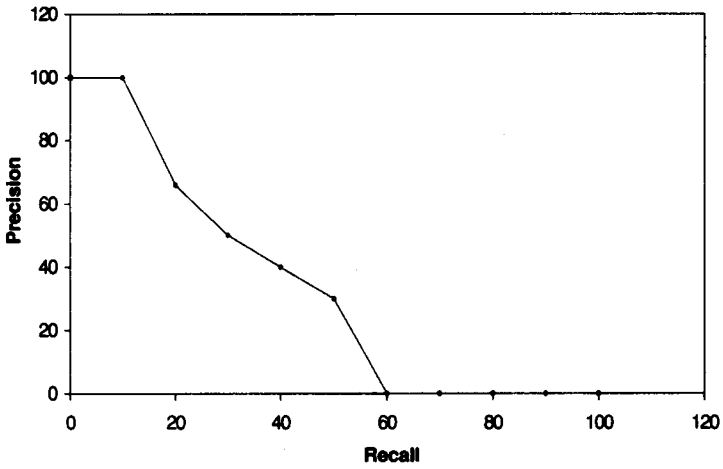

Figure 3.2 Precision at 11 standard recall levels.

all test queries, we average the precision figures at each recall level as follows.

$$\overline{P}(r) = \sum_{i=1}^{N_q} \frac{P_i(r)}{N_q} \tag{3.2}$$

where $\overline{P}(r)$ is the average precision at the recall level r, N_q is the number of queries used, and $P_i(r)$ is the precision at recall level r for the i-th query.

Since the recall levels for each query might be distinct from the 11 standard recall levels, utilization of an interpolation procedure is often necessary. For instance, consider again the set of 15 ranked documents presented above. Assume that the set of relevant documents for the query q has changed and is now given by

$$R_q = \{d_3, d_{56}, d_{129}\} \tag{3.3}$$

In this case, the first relevant document in the ranking for query q is d_{56} which provides a recall level of 33.3% (with precision also equal to 33.3%) because, at this point, one-third of all relevant documents have already been seen. The second relevant document is d_{129} which provides a recall level of 66.6% (with precision equal to 25%). The third relevant document is d_3 which provides a recall level of 100% (with precision equal to 20%). The precision figures at the 11 standard recall levels are interpolated as follows.

Let r_j, $j \in \{0, 1, 2, \ldots, 10\}$, be a reference to the j-th standard recall level (i.e., r_5 is a reference to the recall level 50%). Then,

$$P(r_j) = max_{\,r_j \leq r \leq r_{j+1}} \; P(r) \tag{3.4}$$

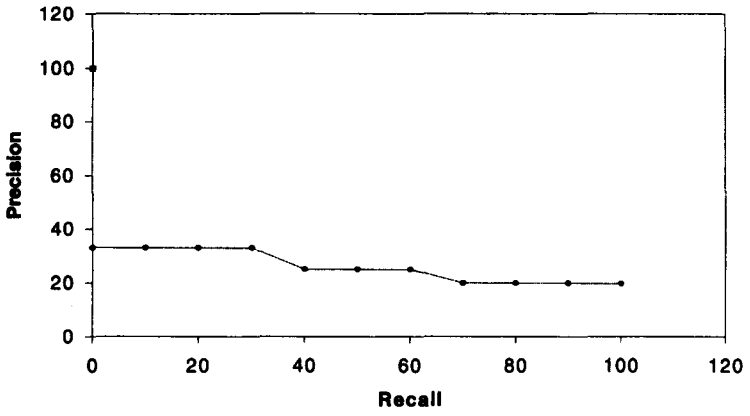

Figure 3.3 Interpolated precision at 11 standard recall levels relative to $R_q = \{d_3, d_{56}, d_{129}\}$.

which states that the interpolated precision at the j-th standard recall level is the maximum known precision at any recall level between the j-th recall level and the $(j+1)$-th recall level.

In our last example, this interpolation rule yields the precision and recall figures illustrated in Figure 3.3. At recall levels 0%, 10%, 20%, and 30%, the interpolated precision is equal to 33.3% (which is the known precision at the recall level 33.3%). At recall levels 40%, 50%, and 60%, the interpolated precision is 25% (which is the precision at the recall level 66.6%). At recall levels 70%, 80%, 90%, and 100%, the interpolated precision is 20% (which is the precision at recall level 100%).

The curve of precision versus recall which results from averaging the results for various queries is usually referred to as precision versus recall figures. Such average figures are normally used to compare the retrieval performance of distinct retrieval algorithms. For instance, one could compare the retrieval performance of a newly proposed retrieval algorithm with the retrieval performance of the classic vector space model. Figure 3.4 illustrates average precision versus recall figures for two distinct retrieval algorithms. In this case, one algorithm has higher precision at lower recall levels while the second algorithm is superior at higher recall levels.

One additional approach is to compute average precision at given *document cutoff values*. For instance, we can compute the average precision when 5, 10, 15, 20, 30, 50, or 100 relevant documents have been seen. The procedure is analogous to the computation of average precision at 11 standard recall levels but provides additional information on the retrieval performance of the ranking algorithm.

Average precision versus recall figures are now a standard evaluation strategy for information retrieval systems and are used extensively in the information retrieval literature. They are useful because they allow us to evaluate

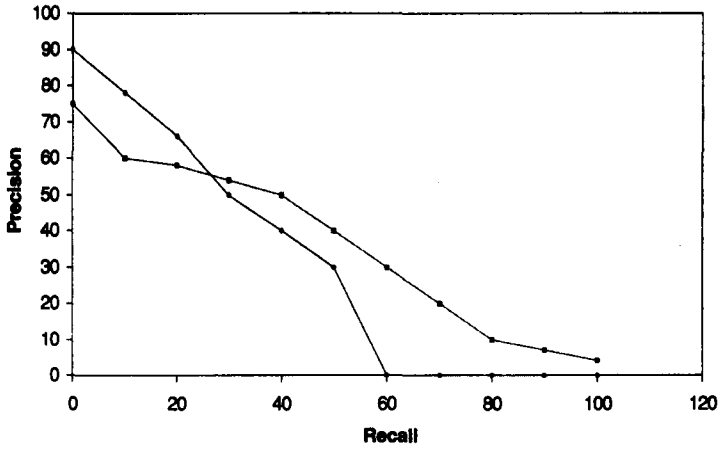

Figure 3.4 Average recall versus precision figures for two distinct retrieval algorithms.

quantitatively both the quality of the overall answer set and the breadth of the retrieval algorithm. Further, they are simple, intuitive, and can be combined in a single curve. However, precision versus recall figures also have their disadvantages and their widespread usage has been criticized in the literature. We return to this point later on. Before that, let us discuss techniques for summarizing precision versus recall figures by a single numerical value.

Single Value Summaries

Average precision versus recall figures are useful for comparing the retrieval performance of distinct retrieval algorithms over a set of example queries. However, there are situations in which we would like to compare the retrieval performance of our retrieval algorithms for the individual queries. The reasons are twofold. First, averaging precision over many queries might disguise important anomalies in the retrieval algorithms under study. Second, when comparing two algorithms, we might be interested in investigating whether one of them outperforms the other for each query in a given set of example queries (notice that this fact can be easily hidden by an average precision computation). In these situations, a single precision value (for each query) can be used. This single value should be interpreted as a summary of the corresponding precision versus recall curve. Usually, this single value summary is taken as the precision at a specified recall level. For instance, we could evaluate the precision when we observe the first relevant document and take this precision as the single value summary. Of course, as seems obvious, this is not a good approach. More interesting strategies can be adopted as we now discuss.

Average Precision at Seen Relevant Documents

The idea here is to generate a single value summary of the ranking by averaging the precision figures obtained after each new relevant document is observed (in the ranking). For instance, consider the example in Figure 3.2. The precision figures after each new relevant document is observed are 1, 0.66, 0.5, 0.4, and 0.3. Thus, the *average precision at seen relevant documents* is given by (1+0.66+0.5+0.4+0.3)/5 or 0.57. This measure favors systems which retrieve relevant documents quickly (i.e., early in the ranking). Of course, an algorithm might present a good average precision at seen relevant documents but have a poor performance in terms of overall recall.

R-Precision

The idea here is to generate a single value summary of the ranking by computing the precision at the R-th position in the ranking, where R is the total number of relevant documents for the current query (i.e., number of documents in the set R_q). For instance, consider the examples in Figures 3.2 and 3.3. The value of R-precision is 0.4 for the first example (because $R = 10$ and there are four relevant documents among the first ten documents in the ranking) and 0.33 for the second example (because $R = 3$ and there is one relevant document among the first three documents in the ranking). The R-precision measure is a useful parameter for observing the behavior of an algorithm for each individual query in an experiment. Additionally, one can also compute an average R-precision figure over all queries. However, using a single number to summarize the full behavior of a retrieval algorithm over several queries might be quite imprecise.

Precision Histograms

The R-precision measures for several queries can be used to compare the retrieval history of two algorithms as follows. Let $RP_A(i)$ and $RP_B(i)$ be the R-precision values of the retrieval algorithms A and B for the i-th query. Define, for instance, the difference

$$RP_{A/B}(i) = RP_A(i) - RP_B(i) \tag{3.5}$$

A value of $RP_{A/B}(i)$ equal to 0 indicates that both algorithms have equivalent performance (in terms of R-precision) for the i-th query. A positive value of $RP_{A/B}(i)$ indicates a better retrieval performance by algorithm A (for the i-th query) while a negative value indicates a better retrieval performance by algorithm B. Figure 3.5 illustrates the $RP_{A/B}(i)$ values (labeled *R-Precision A/B*) for two hypothetical retrieval algorithms over ten example queries. The algorithm A is superior for eight queries while the algorithm B performs better for the two other queries (numbered 4 and 5). This type of bar graph is called a *precision histogram* and allows us to quickly compare the retrieval performance history of two algorithms through visual inspection.

Summary Table Statistics

Single value measures can also be stored in a table to provide a statistical summary regarding the set of all the queries in a retrieval task. For instance, these

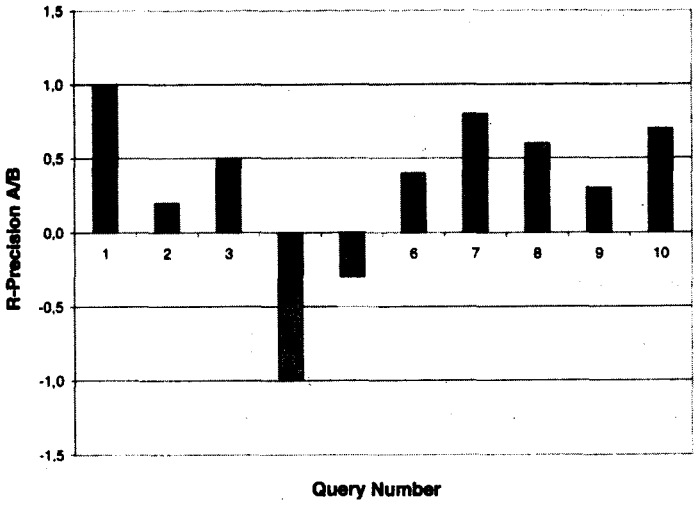

Figure 3.5 A precision histogram for ten hypothetical queries.

summary table statistics could include: the number of queries used in the task, the total number of documents retrieved by all queries, the total number of relevant documents which were effectively retrieved when all queries are considered, the total number of relevant documents which could have been retrieved by all queries, etc.

Precision and Recall Appropriateness

Precision and recall have been used extensively to evaluate the retrieval performance of retrieval algorithms. However, a more careful reflection reveals problems with these two measures [451, 664, 754]. First, the proper estimation of maximum recall for a query requires detailed knowledge of all the documents in the collection. With large collections, such knowledge is unavailable which implies that recall cannot be estimated precisely. Second, recall and precision are related measures which capture different aspects of the set of retrieved documents. In many situations, the use of a single measure which combines recall and precision could be more appropriate. Third, recall and precision measure the effectiveness over a set of queries processed in batch mode. However, with modern systems, interactivity (and not batch processing) is the key aspect of the retrieval process. Thus, measures which quantify the *informativeness* of the retrieval process might now be more appropriate. Fourth, recall and precision are easy to define when a linear ordering of the retrieved documents is enforced. For systems which require a weak ordering though, recall and precision might be inadequate.

3.2.2 Alternative Measures

Since recall and precision, despite their popularity, are not always the most appropriate measures for evaluating retrieval performance, alternative measures have been proposed over the years. A brief review of some of them is as follows.

The Harmonic Mean

As discussed above, a single measure which combines recall and precision might be of interest. One such measure is the harmonic mean F of recall and precision [422] which is computed as

$$F(j) = \frac{2}{\frac{1}{r(j)} + \frac{1}{P(j)}} \tag{3.6}$$

where $r(j)$ is the recall for the j-th document in the ranking, $P(j)$ is the precision for the j-th document in the ranking, and $F(j)$ is the harmonic mean of $r(j)$ and $P(j)$ (thus, relative to the j-th document in the ranking). The function F assumes values in the interval $[0, 1]$. It is 0 when no relevant documents have been retrieved and is 1 when all ranked documents are relevant. Further, the harmonic mean F assumes a high value only when both recall and precision are high. Therefore, determination of the maximum value for F can be interpreted as an attempt to find the best possible compromise between recall and precision.

The E Measure

Another measure which combines recall and precision was proposed by van Rijsbergen [785] and is called the E evaluation measure. The idea is to allow the user to specify whether he is more interested in recall or in precision. The E measure is defined as follows.

$$E(j) = 1 - \frac{1 + b^2}{\frac{b^2}{r(j)} + \frac{1}{P(j)}}$$

where $r(j)$ is the recall for the j-th document in the ranking, $P(j)$ is the precision for the j-th document in the ranking, $E(j)$ is the E evaluation measure relative to $r(j)$ and $P(j)$, and b is a user specified parameter which reflects the relative importance of recall and precision. For $b = 1$, the $E(j)$ measure works as the complement of the harmonic mean $F(j)$. Values of b greater than 1 indicate that the user is more interested in precision than in recall while values of b smaller than 1 indicate that the user is more interested in recall than in precision.

User-Oriented Measures

Recall and precision are based on the assumption that the set of relevant documents for a query is the same, independent of the user. However, different users might have a different interpretation of which document is relevant and which one is not. To cope with this problem, *user-oriented* measures have been proposed such as coverage ratio, novelty ratio, relative recall, and recall effort [451].

As before, consider a reference collection, an example information request I, and a retrieval strategy to be evaluated. Let R be the set of relevant documents for I and A be the answer set retrieved. Also, let U be the subset of R which is known to the user. The number of documents in U is $|U|$. The intersection of the sets A and U yields the documents known to the user to be relevant which were retrieved. Let $|Rk|$ be the number of documents in this set. Further, let $|Ru|$ be the number of relevant documents previously unknown to the user which were retrieved. Figure 3.6 illustrates the situation. The *coverage ratio* is defined as the fraction of the documents known (to the user) to be relevant which has actually been retrieved i.e.,

$$coverage = \frac{|Rk|}{|U|}$$

The *novelty ratio* is defined as the fraction of the relevant documents retrieved which was unknown to the user i.e.,

$$novelty = \frac{|Ru|}{|Ru| + |Rk|}$$

A high coverage ratio indicates that the system is finding most of the relevant documents the user expected to see. A high novelty ratio indicates that the system is revealing (to the user) many new relevant documents which were previously unknown.

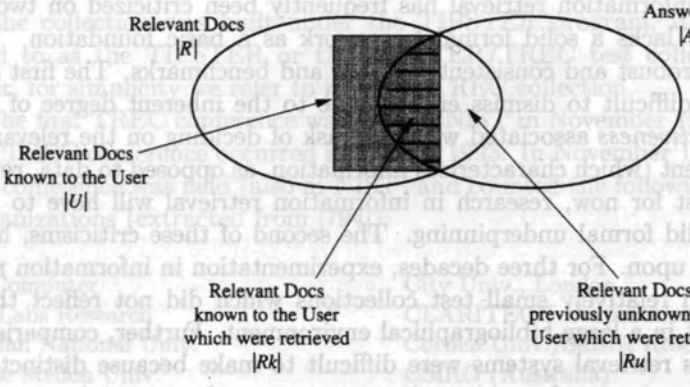

Figure 3.6 Coverage and novelty ratios for a given example information request.

Additionally, two other measures can be defined as follows. The *relative recall* is given by the ratio between the number of relevant documents found (by the system) and the number of relevant documents the user expected to find. In the case when the user finds as many relevant documents as he expected, he stops searching and the relative recall is equal to 1. The *recall effort* is given by the ratio between the number of relevant documents the user expected to find and the number of documents examined in an attempt to find the expected relevant documents.

Other Measures

Other measures which might be of interest include the *expected search length*, which is good for dealing with sets of documents weakly ordered, the *satisfaction*, which takes into account only the relevant documents, and the *frustration*, which takes into account only the non-relevant documents [451].

3.3 Reference Collections

In this section we discuss various reference collections which have been used throughout the years for the evaluation of information retrieval systems. We first discuss the TIPSTER/TREC collection which, due to its large size and thorough experimentation, is usually considered to be the *reference* test collection in information retrieval nowadays. Following that, we cover the CACM and ISI collections due to their historical importance in the area of information retrieval. We conclude this section with a brief discussion of the Cystic Fibrosis collection. It is a small collection whose example information requests were extensively studied by four groups of specialists before generation of the relevant document sets.

3.3.1 The TREC Collection

Research in information retrieval has frequently been criticized on two fronts. First, that it lacks a solid formal framework as a basic foundation. Second, that it lacks robust and consistent testbeds and benchmarks. The first of these criticisms is difficult to dismiss entirely due to the inherent degree of psychological *subjectiveness* associated with the task of deciding on the relevance of a given document (which characterizes information, as opposed to data, retrieval). Thus, at least for now, research in information retrieval will have to proceed without a solid formal underpinning. The second of these criticisms, however, can be acted upon. For three decades, experimentation in information retrieval was based on relatively small test collections which did not reflect the main issues present in a large bibliographical environment. Further, comparisons between various retrieval systems were difficult to make because distinct groups conducted experiments focused on distinct aspects of retrieval (even when the same test collection was used) and there were no widely accepted benchmarks.

In the early 1990s, a reaction to this state of disarray was initiated under the leadership of Donna Harman at the National Institute of Standards and Technology (NIST), in Maryland. Such an effort consisted of promoting a yearly conference, named TREC for Text REtrieval Conference, dedicated to experimentation with a large test collection comprising over a million documents. For each TREC conference, a set of reference experiments is designed. The research groups which participate in the conference use these reference experiments for comparing their retrieval systems.

A clear statement of the purpose of the TREC conferences can be found in the NIST TREC Web site [768] and reads as follows.

> The TREC conference series is co-sponsored by the National Institute of Standards and Technology (NIST) and the Information Technology Office of the Defense Advanced Research Projects Agency (DARPA) as part of the TIPSTER Text Program. The goal of the conference series is to encourage research in information retrieval from large text applications by providing a large test collection, uniform scoring procedures, and a forum for organizations interested in comparing their results. Attendance at TREC conferences is restricted to those researchers and developers who have performed the TREC retrieval tasks and to selected government personnel from sponsoring agencies.
>
> Participants in a TREC conference employ a wide variety of retrieval techniques, including methods using automatic thesauri, sophisticated term weighting, natural language techniques, relevance feedback, and advanced pattern matching. Each system works with the same test collection that consists of about 2 gigabytes of text (over 1 million documents) and a given set of information needs called 'topics.' Results are run through a common evaluation package so that groups can compare the effectiveness of different techniques and can determine how differences between systems affect performance.

Since the collection was built under the TIPSTER program, it is frequently referred to as the TIPSTER or the TIPSTER/TREC test collection. Here, however, for simplicity we refer to it as the TREC collection.

The first TREC conference was held at NIST in November 1992, while the second TREC conference occurred in August 1993. In November 1997, the sixth TREC conference was held (also at NIST) and counted the following participating organizations (extracted from [794]):

Apple Computer
AT&T Labs Research
Australian National Univ.
Carnegie Mellon Univ.
CEA (France)
Center for Inf. Res., Russia
City Univ., London
CLARITECH Corporation
Cornell Univ./SaBIR Research, Inc.
CSIRO (Australia)
Daimler Benz Res. Center, Ulm
Dublin Univ. Center

Duke Univ./Univ. of Colorado/Bellcore
ETH (Switzerland)
FS Consulting, Inc.
GE Corp./Rutgers Univ.
George Mason Univ./NCR Corp.
Harris Corp.
IBM T.J. Watson Res. (2 groups)
ISS (Singapore)
ITI (Singapore)
APL, Johns Hopkins Univ.
LEXIS-NEXIS
MDS at RMIT, Australia
MIT/IBM Almaden Res. Center
MSI/IRIT/Univ. Toulouse
NEC Corporation
New Mexico State Univ. (2 groups)
NSA (Speech Research Group)
Open Text Corporation
Oregon Health Sciences Univ.
Queens College, CUNY
Rutgers Univ. (2 groups)
Siemens AG
SRI International
TwentyOne
Univ. California, Berkeley
Univ. California, San Diego
Univ. Glasgow
Univ. Maryland, College Park
Univ. Massachusetts, Amherst
Univ. Montreal
Univ. North Carolina (2 groups)
Univ. Sheffield/Univ. Cambridge
Univ. Waterloo
Verity, Inc.
Xerox Res. Centre Europe

The seventh TREC conference was held again at NIST in November of 1998.

In the following, we briefly discuss the TREC document collection and the (benchmark) tasks at the TREC conferences. As with most test collections, the TREC collection is composed of three parts: the documents, the example information requests (called *topics* in the TREC nomenclature), and a set of relevant documents for each example information request. Further, the TREC conferences also include a set of tasks to be used as a benchmark.

The Document Collection

The TREC collection has been growing steadily over the years. At TREC-3, the collection size was roughly 2 gigabytes while at TREC-6 it had gone up to roughly 5.8 gigabytes. In the beginning, copyright restrictions prevented free distribution of the collection and, as a result, the distribution CD-ROM disks had to be bought. In 1998, however, an arrangement was made which allows free access to the documents used in the most recent TREC conferences. As a result, TREC disk 4 and TREC disk 5 are now available from NIST at a small fee (US$200 in 1998) to cover distribution costs. Information on how to obtain the collection (which comes with the disks) and the topics with their relevant document sets (which have to be retrieved through the network) can be obtained directly from the NIST TREC Web site [768].

The TREC collection is distributed in six CD-ROM disks of roughly 1 gigabyte of compressed text each. The documents come from the following sources:

WSJ	→ *Wall Street Journal*
AP	→ Associated Press (news wire)
ZIFF	→ Computer Selects (articles), Ziff-Davis
FR	→ Federal Register

DOE → US DOE Publications (abstracts)
SJMN → *San Jose Mercury News*
PAT → US Patents
FT → *Financial Times*
CR → Congressional Record
FBIS → Foreign Broadcast Information Service
LAT → *LA Times*

Table 3.1 illustrates the contents of each disk and some simple statistics regarding the collection (extracted from [794]). Documents from all subcollections are

Disk	*Contents*	*Size Mb*	*Number Docs*	*Words/Doc. (median)*	*Words/Doc. (mean)*
1	WSJ, 1987-1989	267	98,732	245	434.0
	AP, 1989	254	84,678	446	473.9
	ZIFF	242	75,180	200	473.0
	FR, 1989	260	25,960	391	1315.9
	DOE	184	226,087	111	120.4
2	WSJ, 1990-1992	242	74,520	301	508.4
	AP, 1988	237	79,919	438	468.7
	ZIFF	175	56,920	182	451.9
	FR, 1988	209	19,860	396	1378.1
3	SJMN, 1991	287	90,257	379	453.0
	AP, 1990	237	78,321	451	478.4
	ZIFF	345	161,021	122	295.4
	PAT, 1993	243	6,711	4,445	5391.0
4	FT, 1991-1994	564	210,158	316	412.7
	FR, 1994	395	55,630	588	644.7
	CR, 1993	235	27,922	288	1373.5
5	FBIS	470	130,471	322	543.6
	LAT	475	131,896	351	526.5
6	FBIS	490	120,653	348	581.3

Table 3.1 Document collection used at TREC-6. Stopwords are not removed and no stemming is performed (see Chapter 7 for details on stemming).

tagged with SGML (see Chapter 6) to allow easy parsing (which implies simple coding for the groups participating at TREC conferences). Major structures such as a field for the document number (identified by <DOCNO>) and a field for the document text (identified by <TEXT>) are common to all documents. Minor structures might be different across subcollections to preserve parts of the structure in the original document. This has been the philosophy for formatting decisions at NIST: preserve as much of the original structure as possible while providing a common framework which allows simple decoding of the data.

An example of a TREC document is the document numbered 880406-0090

```
<doc>

<docno> WSJ880406-0090 </docno>
<hl> AT&T Unveils Services to Upgrade Phone Networks Under
Global Plan </hl>
<author> Janet Guyon (WSJ Staff) </author>
<dateline> New York </dateline>

<text>
American Telephone & Telegraph Co. introduced the first of a new
generation of phone services with broad ...
</text>

</doc>
```

Figure 3.7 TREC document numbered WSJ880406-0090.

in the *Wall Street Journal* subcollection which is shown in Figure 3.7 (extracted from [342]). Further details on the TREC document collection can be obtained from [794, 768].

The Example Information Requests (Topics)

The TREC collection includes a set of example *information requests* which can be used for testing a new ranking algorithm. Each request is a description of an information need in natural language. In the TREC nomenclature, each test information request is referred to as a *topic*. An example of an information request in TREC is the topic numbered 168 (prepared for the TREC-3 conference) which is illustrated in Figure 3.8 (extracted from [342]).

The task of converting an information request (topic) into a system query (i.e., a set of index terms, a Boolean expression, a fuzzy expression, etc.) must be done by the system itself and is considered to be an integral part of the evaluation procedure.

The number of topics prepared for the first six TREC conferences goes up to 350. The topics numbered 1 to 150 were prepared for use with the TREC-1 and TREC-2 conferences. They were written by people who were experienced users of real systems and represented long-standing information needs. The topics numbered 151 to 200 were prepared for use with the TREC-3 conference, are shorter, and have a simpler structure which includes only three subfields (named Title, Description, and Narrative as illustrated in the topic 168 above). The topics numbered 201 to 250 were prepared for use with the TREC-4 conference and are even shorter. At the TREC-5 (which included topics 251-300) and TREC-6 (which included topics 301-350) conferences, the topics were prepared with a composition similar to the topics in TREC-3 (i.e., they were expanded with respect to the topics in TREC-4 which were considered to be too short).

<top>

<num> Number: 168
<title> Topic: Financing AMTRAK

<desc> Description:
A document will address the role of the Federal Government in financing the operation of the National Railroad Transportation Corporation (AMTRAK).

<narr> Narrative: A relevant document must provide information on the government's responsibility to make AMTRAK an economically viable entity. It could also discuss the privatization of AMTRAK as an alternative to continuing government subsidies. Documents comparing government subsidies given to air and bus transportation with those provided to AMTRAK would also be relevant.

</top>

Figure 3.8 Topic numbered 168 in the TREC collection.

The Relevant Documents for Each Example Information Request

At the TREC conferences, the set of relevant documents for each example information request (topic) is obtained from a pool of possible relevant documents. This pool is created by taking the top K documents (usually, $K = 100$) in the rankings generated by the various participating retrieval systems. The documents in the pool are then shown to human assessors who ultimately decide on the relevance of each document.

This technique for assessing relevance is called the *pooling method* [794] and is based on two assumptions. First, that the vast majority of the relevant documents is collected in the assembled pool. Second, that the documents which are not in the pool can be considered to be not relevant. Both assumptions have been verified to be accurate in tests done at the TREC conferences. A detailed description of these relevance assessments can be found in [342, 794].

The (Benchmark) Tasks at the TREC Conferences

The TREC conferences include two main information retrieval tasks [342]. In the first, called *ad hoc* task, a set of new (conventional) requests are run against a fixed document database. This is the situation which normally occurs in a library where a user is asking new queries against a set of static documents. In the second, called *routing* task, a set of fixed requests are run against a database whose documents are continually changing. This is like a filtering task in which the same questions are always being asked against a set of dynamic documents (for instance, news clipping services). Unlike a pure filtering task, however, the retrieved documents must be ranked.

For the ad hoc task, the participant systems receive the test information requests and execute them on a pre-specified document collection. For the routing task, the participant systems receive the test information requests and two distinct document collections. The first collection is used for training and allows the tuning of the retrieval algorithm. The second collection is used for testing the tuned retrieval algorithm.

Starting at the TREC-4 conference, new secondary tasks, besides the ad hoc and routing tasks, were introduced with the purpose of allowing more specific comparisons among the various systems. At TREC-6, eight (specific) secondary tasks were added in as follows.

- **Chinese** Ad hoc task in which both the documents and the topics are in Chinese.
- **Filtering** Routing task in which the retrieval algorithm has only to decide whether a new incoming document is relevant (in which case it is taken) or not (in which case it is discarded). No ranking of the documents taken needs to be provided. The test data (incoming documents) is processed in time-stamp order.
- **Interactive** Task in which a human searcher interacts with the retrieval system to determine the relevant documents. Documents are ruled relevant or not relevant (i.e., no ranking is provided).
- **NLP** Task aimed at verifying whether retrieval algorithms based on natural language processing offer advantages when compared to the more traditional retrieval algorithms based on index terms.
- **Cross languages** Ad hoc task in which the documents are in one language but the topics are in a different language.
- **High precision** Task in which the user of a retrieval system is asked to retrieve ten documents that answer a given (and previously unknown) information request within five minutes (wall clock time).
- **Spoken document retrieval** Task in which the documents are written transcripts of radio broadcast news shows. Intended to stimulate research on retrieval techniques for spoken documents.
- **Very large corpus** Ad hoc task in which the retrieval systems have to deal with collections of size 20 gigabytes (7.5 million documents).

For TREC-7, the NLP and the Chinese secondary tasks were discontinued. Additionally, the routing task was retired as a main task because there is a consensus that the filtering task is a more realistic type of routing task. TREC-7 also included a new task called *Query Task* in which several distinct query versions were created for each example information request [794]. The main goal of this task is to allow investigation of query-dependent retrieval strategies, a well known problem with the TREC collection due to the sparsity of the given information requests (which present very little overlap) used in past TREC conferences.

Besides providing detailed descriptions of the tasks to be executed, the TREC conferences also make a clear distinction between two basic techniques for transforming the information requests (which are in natural language) into query statements (which might be in vector form, in Boolean form, etc.). In the TREC-6 conference, the allowable query construction methods were divided into *automatic* methods, in which the queries were derived completely automatically from the test information requests, and *manual* methods, in which the queries were derived using any means other than the fully automatic method [794].

Evaluation Measures at the TREC Conferences

At the TREC conferences, four basic types of evaluation measures are used: summary table statistics, recall-precision averages, document level averages, and average precision histograms. Briefly, these measures can be described as follows (see further details on these measures in Section 3.2).

- **Summary table statistics** Consists of a table which summarizes statistics relative to a given task. The statistics included are: the number of topics (information requests) used in the task, the number of documents retrieved over all topics, the number of relevant documents which were effectively retrieved for all topics, and the number of relevant documents which could have been retrieved for all topics.
- **Recall-precision averages** Consists of a table or graph with average precision (over all topics) at 11 standard recall levels. Since the recall levels of the individual queries are seldom equal to the standard recall levels, interpolation is used to define the precision at the standard recall levels. Further, a non-interpolated average precision over seen relevant documents (and over all topics) might be included.
- **Document level averages** In this case, average precision (over all topics) is computed at specified document cutoff values (instead of standard recall levels). For instance, the average precision might be computed when 5, 10, 20, 100 relevant documents have been seen. Further, the average R-precision value (over all queries) might also be provided.
- **Average precision histogram** Consists of a graph which includes a single measure for each separate topic. This measure (for a topic t_i) is given, for instance, by the difference between the R-precision (for topic t_i) for a target retrieval algorithm and the average R-precision (for topic t_i) computed from the results of all participating retrieval systems.

3.3.2 The CACM and ISI Collections

The TREC collection is a large collection which requires time consuming preparation before experiments can be carried out effectively at a local site. Further,

the testing itself is also time consuming and requires much more effort than that required to execute the testing in a small collection. For groups who are not interested in making this investment, an alternative approach is to use a smaller test collection which can be installed and experimented with in a much shorter time. Further, a small collection might include features which are not present in the larger TREC collection. For instance, it is well known that the example information requests at TREC present very little overlap among themselves and thus are not very useful for investigating the impact of techniques which take advantage of information derived from dependencies between the current and past user queries (an issue which received attention at the TREC-7 conference). Further, the TREC collection does not provide good support for experimenting with algorithms which combine distinct evidential sources (such as co-citations, bibliographic coupling, etc.) to generate a ranking. In these situations, alternative (and smaller) test collections might be more appropriate.

For the experimental studies in [271], five different (small) test collections were developed: ADI (documents on information science), CACM, INSPEC (abstracts on electronics, computer, and physics), ISI, and Medlars (medical articles). In this section we cover two of them in detail: the CACM and the ISI test collections. Our discussion is based on the work by Fox [272].

The CACM Collection

The documents in the CACM test collection consist of all the 3204 articles published in the *Communications of the ACM* from the first issue in 1958 to the last number of 1979. Those documents cover a considerable range of computer science literature due to the fact that the CACM served for many years as the premier periodical in the field.

Besides the text of the documents, the collection also includes information on structured *subfields* (called *concepts* by Fox) as follows:

- author names
- date information
- word stems from the title and abstract sections
- categories derived from a hierarchical classification scheme
- direct references between articles
- bibliographic coupling connections
- number of co-citations for each pair of articles.

The subfields 'author names' and 'date information' provide information on authors and date of publication. The subfield 'word stems' provides, for each document, a list of indexing terms (from the title and abstract sections) which have been stemmed (i.e., reduced to their grammatical roots as explained in Chapter 7). The subfield 'categories' assigns a list of classification categories (from the Computing Reviews category scheme) to each document. Since the

categories are fairly broad, the number of categories for any given document is usually smaller than five. The subfield 'direct references' provides a list of pairs of documents $[d_a,d_b]$ in which each pair identifies a document d_a which includes a direct reference to a document d_b. The subfield 'bibliographic coupling' provides a list of triples $[d_1,d_2,n_{cited}]$ in which the documents d_1 and d_2 both include a direct reference to a same third document d_j and the factor n_{cited} counts the number of documents d_j cited by both d_1 and d_2. The subfield 'co-citations' provides a list of triples $[d_1,d_2,n_{citing}]$ in which the documents d_1 and d_2 are both cited by a same third document d_j and the factor n_{citing} counts the number of documents d_j citing both d_1 and d_2. Thus, the CACM collection provides a unique environment for testing retrieval algorithms which are based on information derived from cross-citing patterns — a topic which has attracted much attention in the past.

The CACM collection also includes a set of 52 test information requests. For instance, the information request numbered 1 reads as follows.

> What articles exist which deal with TSS (Time Sharing System), an operating system for IBM computers?

For each information request, the collection also includes two Boolean query formulations and a set of relevant documents. Since the information requests are fairly specific, the average number of relevant documents for each information request is small and around 15. As a result, precision and recall figures tend to be low.

The ISI Collection

The 1460 documents in the ISI (often referred to as CISI) test collection were selected from a previous collection assembled by Small [731] at the Institute of Scientific Information (ISI). The documents selected (which are about information sciences) were those most cited in a cross-citation study done by Small. The main purpose of the ISI collection is to support investigation of similarities based on terms and on cross-citation patterns.

The documents in the ISI collection include three types of subfields as follows.

- author names
- word stems from the title and abstract sections
- number of co-citations for each pair of articles.

The meaning of each of these subfields is as in the CACM collection.

The ISI collection includes a total of 35 test information requests (in natural language) for which there are Boolean query formulations. It also includes 41 additional test information requests for which there is no Boolean query formulation (only the version in natural language). The information requests are

fairly general which resulted in a larger number of relevant documents to each request (around 50). However, many of these relevant documents have no terms in common with the information requests which implies that precision and recall figures tend to be low.

Statistics for the CACM and ISI Collections

Tables 3.2 and 3.3 provide comparative summary statistics for the CACM and the ISI test collections.

Collection	*Num. Docs*	*Num. Terms*	*Terms/Docs.*
CACM	3204	10,446	40.1
ISI	1460	7392	104.9

Table 3.2 Document statistics for the CACM and ISI collections.

Collection	*Number Queries*	*Terms per Query*	*Relevants per Query*	*Relevants in Top 10*
CACM	52	11.4	15.3	1.9
ISI	35 & 76	8.1	49.8	1.7

Table 3.3 Query statistics for the CACM and ISI collections.

We notice that, compared to the size of the collection, the ISI collection has a much higher percentage of relevant documents per query (3.4%) than the CACM collection (0.5%). However, as already discussed, many of the relevant documents in the ISI collection have no terms in common with the respective information requests which usually yields low precision.

Related Test Collections

At the Virginia Polytechnic Institute and State University, Fox has assembled together nine small test collections in a CD-ROM. These test collections have sizes comparable to those of the CACM and ISI collections, but include their own particularities. Since they have been used throughout the years for evaluation of information retrieval systems, they provide a good setting for the preliminary testing of information retrieval algorithms. A list of these nine test collections is provided in Table 3.4.

3.3.3 The Cystic Fibrosis Collection

The cystic fibrosis (CF) collection [721] is composed of 1239 documents indexed with the term 'cystic fibrosis' in the National Library of Medicine's MEDLINE database. Each document contains the following fields:

Collection	*Subject*	*Num. Docs*	*Num. Queries*
ADI	Information Science	82	35
CACM	Computer Science	3200	64
ISI	Library Science	1460	76
CRAN	Aeronautics	1400	225
LISA	Library Science	6004	35
MED	Medicine	1033	30
NLM	Medicine	3078	155
NPL	Elec. Engineering	11,429	100
TIME	General Articles	423	83

Table 3.4 Test collections related to the CACM and ISI collections.

- MEDLINE accession number
- author
- title
- source
- major subjects
- minor subjects
- abstract (or extract)
- references
- citations.

The collection also includes 100 information requests (generated by an expert with two decades of clinical and research experience with cystic fibrosis) and the documents relevant to each query. Further, 4 separate relevance scores are provided for each relevant document. These relevance scores can be 0 (which indicates non-relevance), 1 (which indicates marginal relevance), and 2 (which indicates high relevance). Thus, the overall relevance score for a document (relative to a given query) varies from 0 to 8. Three of the relevance scores were provided by subject experts while the fourth relevance score was provided by a medical bibliographer.

Table 3.5 provides some statistics regarding the information requests in the CF collection. We notice that the number of queries with at least one relevant document is close to the total number of queries in the collection. Further, for various relevance thresholds (the minimum value of relevance score used to characterize relevance), the average number of relevant documents per query is between 10 and 30.

The CF collection, despite its small size, has two important characteristics. First, its set of relevance scores was generated directly by human experts through a careful evaluation strategy. Second, it includes a good number of information requests (relative to the collection size) and, as a result, the respective query vectors present overlap among themselves. This allows experimentation

Relevance Threshold	*Queries At Least 1 Rel. Doc*	*Min. Num. Rel. Docs*	*Max. Num. Rel. Docs*	*Avg. Num. Rel. Docs*
1	100	2	189	31.9
2	100	1	130	18.1
3	99	1	119	14.9
4	99	1	114	14.1
5	99	1	93	10.7
6	94	1	53	6.4

Table 3.5 Summary statistics for the information requests in the CF collection.

with retrieval strategies which take advantage of past query sessions to improve retrieval performance.

3.4 Trends and Research Issues

A major trend today is research in interactive user interfaces. The motivation is a general belief that effective retrieval is highly dependent on obtaining proper feedback from the user. Thus, evaluation studies of interactive interfaces will tend to become more common in the near future. The main issues revolve around deciding which evaluation measures are most appropriate in this scenario. A typical example is the informativeness measure [754] introduced in 1992.

Furthermore, the proposal, the study, and the characterization of alternative measures to recall and precision, such as the harmonic mean and the E measures, continue to be of interest.

3.5 Bibliographic Discussion

A nice chapter on retrieval performance evaluation appeared in the book by Salton and McGill [698]. Even if outdated, it is still interesting reading. The book by Khorfage [451] also includes a full chapter on retrieval evaluation. A recent paper by Mizzaro [569] presents a very complete survey of relevance studies throughout the years. About 160 papers are discussed in this paper.

Two recent papers by Shaw, Burgin, and Howel [422, 423] discuss standards and evaluations in test collections for cluster-based and vector-based retrieval models. These papers also discuss the advantages of the harmonic mean (of recall and precision) as a single alternative measure for recall and precision. Problems with recall and precision related to systems which require a weak document ordering are discussed by Raghavan, Bollmann, and Jung [664, 663]. Tague-Sutcliffe proposes a measure of informativeness for evaluating interactive user sessions [754].