Measurement in Information Retrieval Evaluation

William Edward Webber
Department of Computer Science and Software Engineering
The University of Melbourne

Submitted in total fulfilment of the requirements
of the degree of Doctor of Philosophy

Produced on acid-free paper

September 2010

Abstract

Full-text retrieval systems employ heuristics to match documents to user queries. Retrieval correctness cannot, therefore, be formally proven, but must be evaluated through human assessment. To make evaluation automatable and repeatable, assessments of which documents are relevant to which queries are collected in advance, to form a test collection. Collection-based evaluation has been the standard in retrieval experiments for half a century, but only recently have its statistical foundations been considered.

This thesis makes several contributions to the reliable and efficient measurement of the behaviour and effectiveness of information retrieval systems. First, the high variability in query difficulty makes effectiveness scores difficult to interpret, analyze, and compare. We therefore propose the standardization of scores, based on the observed results of a set of reference systems for each query. We demonstrate that standardization controls variability and enhances comparability. Second, while testing evaluation results for statistical significance has been established as standard practice, the importance of ensuring that significance can be reliably achieved for a meaningful improvement (the power of the test) is poorly understood. We introduce the use of statistical power analysis to the field of retrieval evaluation, finding that most test collections cannot reliably detect incremental improvements in performance. We also demonstrate the pitfalls in predicting score standard deviation during design-phase power analysis, and offer some pragmatic methodological suggestions.

Third, in constructing a test collection, it is not feasible to assess every document for relevance to every query. The practice instead is to run a set of systems against the collection, and pool their top results for assessment. Pooling is potentially biased against systems which are neither included in nor similar to the pooled set. We propose a robust, empirical method for estimating the degree of pooling bias, through performing a leave-one-out experiment on fully pooled systems and adjusting unpooled scores accordingly. Fourth, there are many circumstances in which one wishes directly to compare the document rankings produced by different retrieval systems, independent of their effectiveness. These rankings are top-weighted, non-conjoint, and of arbitrary length, and no suitable similarity measures have been described for such rankings. We propose and analyze such a rank similarity measure, called rank-biased overlap, and demonstrate its utility, on real and simulated data.

Finally, we conclude the thesis with an examination of the state and function of retrieval evaluation. A survey of published results shows that there has been no measurable improvement in retrieval effectiveness over the past decade. This lack of progress has been obscured by the general use of uncompetitive baselines in published experiments, producing the appearance of substantial and statistically significant improvements for new systems without actually advancing the state of the art.

Declaration

This is to certify that

(i) this thesis contains only my original work towards the degree of Doctor of Philosophy except where indicated in the Preface;

(ii) due acknowledgement has been made in the text to all other material used; and,

(iii) this thesis is less than 100,000 words in length, exclusive of tables, maps, bibliographies, and appendices.

William Webber
Department of Computer Science and Software Engineering
The University of Melbourne
September 30, 2010

Acknowledgments

I must first thank my supervisors, Alistair Moffat and Justin Zobel, for their advice and support throughout my candidature. Having shepherded me through my Masters, they gamely undertook to see my through my PhD, as well.

The Australian Research Council has supported my work financially, and the Department of Computer Science and Software Engineering have provided me with space and facilities.

During the period of my candidature, Vivian Lin and I have progressed from strangers to partners to betroth'ed to spouses, just a little faster than my research has moved from conception to publication. I am grateful to her, to my step-son, Yesie Lin-Viota, and to my mother, Jeananne Webber, for their encouragement and forebearance.

Preface

Publications arising from this thesis

The work on metric predictivity described in Section 3.2.5 was presented as a poster at the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Webber, Moffat, Zobel, and Sakai, 2008c). This work was done jointly with Tetsuya Sakai, then of Newswatch, Inc., Japan.

The work on score standardization in Chapter 4 was presented initially at the 12th Australasian Document Computing Symposium (Webber, Moffat, and Zobel, 2007b), and in a more extended form at the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Webber, Moffat, and Zobel, 2008b).

The work on the application of statistical power analysis to information retrieval evaluation in Chapter 5 was presented at the 17th ACM International Conference on Information and Knowledge Management (Webber, Moffat, and Zobel, 2008a).

The work on score adjustment for the correction of pooling bias in Chapter 6 was presented at the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Webber and Park, 2009). This work was done jointly with Laurence Park, then of the University of Melbourne.

The work on the rank-biased overlap measure of rank similarity in Chapter 7 was published in ACM Transactions on Information Systems (Webber, Moffat, and Zobel, 2010).

The analysis of the effectiveness of TREC participant systems over time in Section 8.2.1 was presented as a poster at the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Armstrong, Moffat, Webber, and Zobel, 2009c), and the survey of published retrieval scores on TREC collections in Section 8.2.2 was presented at the 18th ACM International Conference on Information and Knowledge Management (Armstrong, Moffat, Webber, and Zobel, 2009a). Both items of work were done jointly with Tim Armstrong, then of the University of Melbourne.

The survey of the state of evaluation practice in keyword retrieval from databases, described in Section 8.3.2, was published as an invited article in the IEEE Data Engineering Bulletin (Webber, 2010).

Simulation- and randomization-based experiments have been re-run for this thesis, and precise numbers may differ slightly from those of the original publications.

Document preparation, tools and data

The thesis was prepared with the LaTeX document formatting language, using the `algorithm`, `algpseudocode`, `amsfonts`, `amsmath`, `amsopn`, `booktabs`, `caption`,

float, graphicx, listings, multirow, natbib, rotating, times, url, and xcolor packages.

Statistical analysis, simulation experiments, and general programming tasks were performed in the R language and environment for statistical computing, using the plotrix, xtable, and doMC packages. The trec_eval program was used to generate similarity metrics it supports; other metrics were implemented in C or C++, compiled using the gcc compiler.

Graphics were prepared using R, the TikZ graphics system for TeX, and Inkscape. Images in Figures 1.2, 1.3, 6.1, and 6.3 were taken from the icon library of GNOME desktop project, and are licensed under the GNU General Public License. The image of the mouse in Figure 3.12 was created by ArtFavor of OpenClipArt, and was downloaded from http://www.openclipart.org/detail/39871. The image of the ruler in Figure 3.12 was created by jetxee of OpenClipArt, and was download from http://www.openclipart.org/detail/39871.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In the early 1950s, technical librarianship faced a crisis. The scientific boom sparked by the Second World War had released a flood of publications, approaching a million new articles each year. Scientists could no longer stay abreast of current research by general reading alone. Papers relevant to a new project, but not previously known to the researcher, had to be retrieved at the project's outset, and the librarian had to facilitate this retrieval. A variety of cataloguing schemes had been suggested as tools for retrieval, but none had been rigorously tested for effectiveness, and all were labour-intensive to implement.

In responding to technical information's rapid growth, librarians and information scientists developed the field of information retrieval. The defining discovery of the field was that complex schemes for organizing and cataloguing information into hierarchical taxonomies did little better than simply indexing the plain words occurring in the text: the crucial part of information retrieval lay in the process of retrieval. The finding that taxonomy was redundant was little short of scandalous—after all, Western information science had since Aristotle been founded on subdividing knowledge by genus and species. But the effect was liberating. Word occurrences are readily indexed by computer, and retrieval technology could be constructed on top of such indexes without having to solve deep problems in human language analysis and semantics. Significantly, the sufficiency of word occurrence indexing was not argued theoretically (which, after centuries of such theoretical dispute, would hardly have had an impact), but demonstrated empirically, through careful evaluation.

In the mid 1990s, users of the newly-emerged web faced a crisis. The number of web sites was growing rapidly, and finding information by following a trail of links from a few popular central sites was no longer an adequate access method. Manually curated directories such as that of Yahoo! (Figure 1.1) were popular, but manual curation was expensive and scaled poorly. Experienced users could not keep up with the growth in the number of sites, even in areas of personal interest to them; and, for novice users, the task of finding useful information on the web was daunting.

Faced with the mushrooming growth of the web in the second half of the 1990s, a new kind of service provider turned to the decades-old technology of information retrieval, producing the web search engine. Web search transformed information retrieval from the rarefied activity of librarians, researchers, journalist fact-checkers, and intelligence analysts, to the daily activity of almost the entire computer-enabled population. In doing so, search providers finally bridged a long-established gap between theory and practice. As early as the 1960s, researchers had developed statistical techniques

Figure 1.1: The Yahoo! home page, `http://www.yahoo.com`, on August 5th, 1997, the month before the domain name `google.com` was registered. Note the mixture of a search interface (at the top) with a manually curated, hierarchical directory of the web (taking up the rest of the page). (Retrieved from the Internet Archive, `http://web.archive.org/web/19970805071802/http://www10.yahoo.com/`.)

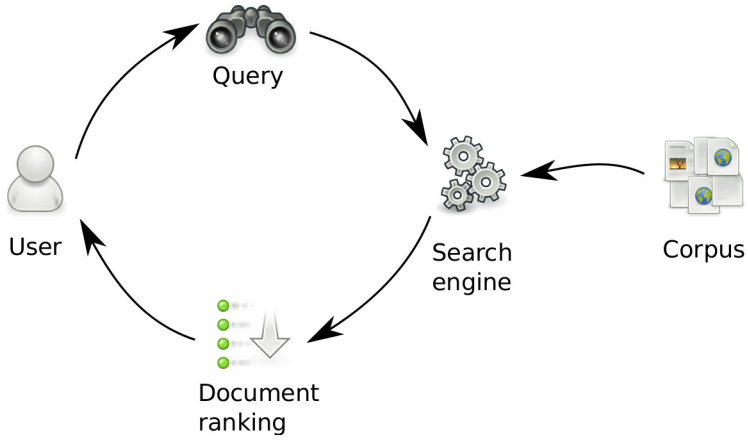


Figure 1.2: Schematic of the information retrieval process. The user poses a query to the search engine. The engine then retrieves documents from the corpus that it estimates to be relevant to the query, ranks them by decreasing probability or degree of relevance, and returns them to the user. The process can be iterated. The effectiveness of the retrieval process can be measured by the utility or satisfaction that the user receives from the results list.

for effectively retrieving and ranking documents against plain keyword queries. The retrieval technology deployed in practice, though, used logical, Boolean query languages that relied upon the patience and expertise of the querier to formulate complex query expressions, precisely specifying their information need. But web users little expertise, and less patience, for constructing complex queries. Search engines therefore turned to simple queries and sophisticated retrieval, finally deploying, on a massive scale, the techniques developed three decades earlier, so creating the modern search engine. To the surprise once more of some search technologists, simple keyword search simply worked. In an increasingly competitive search market, though, how could a provider verify the effectiveness of their search results, and compare their offering with that of their competitors?

Search technology connects simple queries with unannotated documents, relieving both the producer and the consumer of information from the complexity of matching information resources to information needs (Figure 1.2). The result is tools that allow neophyte users to find relevant information, across billions of web documents, in a fraction of a second. But in doing away with complex, formal information representations in favour of rough approximations, statistical information retrieval introduced an important problem. It is not possible to objectively and deterministically state that an information object matches an information request, even in the terms in which the request is formulated. One can say that a document has been manually assigned a certain classification under a hierarchical taxonomy; one can even say that a document contains a Boolean combination of terms; but one cannot conclusively say that an uncategorized document meets a user's information need as expressed by a handful of keywords. The contemporary retrieval system sits at the interface between computational formalism on the one hand, and the ambiguity of human cognition on the other. There is uncertainty in what the retrieval system should do, and therefore in how correct a set of results are.

The ambiguity of the retrieval task makes the question of retrieval effectiveness a crucial and contested one. Methods for evaluating effectiveness are therefore essential, in both research and deployment. Retrieval evaluation relies fundamentally on human assessment of result quality. The noncomputability of effectiveness makes information retrieval a deeply empirical discipline, closer to natural or even social science than to formal computational theory. The complex, interlocked relations that connect imprecise queries, uncurated documents, and inchoate information needs, are not given, but must be hypothesized and tested on observed search behaviour.

The importance of empirical evaluation in information retrieval has been recognized since the field began; the initial work that established the primacy of retrieval over indexing gained much of its impact from the meticulous and painstaking experimental work on which it was based. But the same scale of data that makes retrieval technology necessary, also makes manual assessment costly. While result quality can be measured by directly assessing user satisfaction with, or utility gained from, retrieval results, such direct measurement of the user's satisfaction with the results lists as a whole is neither reusable nor reliably repeatable. Assessing the results of any single system is time-consuming, and there are many competing retrieval algorithms, each tuned by numerous parameters. A parameter change that takes a few minutes to decide upon, and a few seconds to run, could take days to manually assess. Moreover, if each research group produces its own, independent assessments of retrieval quality, then not only is much effort duplicated, but also reproduciblity is impaired, and the potential for bias is introduced. And tuning nowadays is often performed automatically through machine learning; fitting a manual review stage into each learning iteration would be unworkable.

The need for scale and automatability, plus the desire for repeatability and objectivity, has led the information retrieval community to develop hybrid evaluation technologies, part manual, part automated. The most important of the evaluation tools is the test collection: a corpus of documents, with a set of queries (known as *topics*) to run against the corpus, and judgments of which documents are (independently) relevant to each query. These relevance judgments must be manually formed: but once made, the test collection can in principle be reused indefinitely for fully automated evaluation. The result is an automated and re-usable evaluation method, based on a simplified model of retrieval (see Figure 1.3 on page 7).

Test collection evaluation has been the bedrock of retrieval research for half a century. Collection-based experimentation has grown even more in importance since the arrival, beginning in the early 1990s, of large scale, collaboratively developed, and readily obtainable test collections. And (to judge from publicly available information) the test collection method is also core to the quality assurance and improvement methods of commercial web search engines.

The practice of retrieval evaluation, though, has run well ahead of the theory. It was only at the end of the 1990s that the reliability, efficiency, and interpretability of evaluation results began to be formally investigated. The delay was in part because it was only after large-scale collaborative experiments had been running for several years that the datasets needed for a critical investigation of evaluation became available. Initial enquiries, while foundational, tended to be either ad-hoc, or else applied statistical methodology developed in other areas to retrieval evaluation without considering the field's distinctive features. These omissions are currently being remedied by the research community.

It is in the context of the effort for greater reliability, accuracy, robustness, and efficiency in collection-based retrieval evaluation that this thesis is presented. Building

on the foundational work in the area, and employing the large evaluation datasets now available, we make major advances in the accuracy and comparability of evaluation scores; in the design of efficient and reliable experiments; in the extensibility of test collections in dynamic evaluation environments; and in the measurement of retrieval similarity without relevance assessment. We also offer these technical contributions with an awareness of the wider context of evaluation, and of the necessity of mixing experimental rigour with research innovation.

Thesis structure

The field of information retrieval has a long history, and evaluation has been central to the field since the beginning; appreciating this history helps us understand the field's current practice and philosophy. An overview of the history of retrieval evaluation is given in Chapter 2. We begin by examining the inspirational early evaluation work done in the library of the Cranfield Aeronautical College, in the United Kingdom, during the late 1950s and early 1960s, which established the test collection method; and the incorporation of this method into automated, computational retrieval by the SMART project at Cornell University, in the United States, beginning in the mid 1960s. These efforts established from the start the field's core evaluation methods. But the full potential for large-scale, comparable retrieval evaluation was not realized until the production of industrial-size test collections by the TREC effort, starting in 1992. TREC provided the first large-scale, standard document corpora, and produced query sets and relevance judgments to go with them. These collections provided the basis for finally demonstrating that the sophisticated statistical retrieval techniques developed over preceding decades worked on large, realistic collections—validating the technology, and spurring the engineering, that in a few years would be deployed in creating the web search engines so ubiquitous and essential today. Moreover, the TREC experiments brought together dozens of competing research teams, each of whom generated retrieval results against the one test collection. Besides demonstrating the contemporary state of the retrieval art, these submissions formed the data sets of runs and scores that have provided the fertile ground for the empirical study of retrieval evaluation. We make rich use of the TREC run data in later chapters.

Chapter 3 lays down the technical foundation of the thesis. The test collection method achieves its automatability and repeatability by working with a simple, abstract model of the retrieval process. The model approximates the human perception of retrieval quality as an assessment of which documents are independently relevant or irrelevant to each query. Once the retrieval of relevant documents has been accepted as the retrieval system's goal, human assessment only needs to be performed once, in advance, and the evaluation cycle can be automated. The ranked list of documents returned by a retrieval system is converted to a vector of relevance assessments, and this vector scored using a retrieval effectiveness metric. Many such metrics have been proposed, as well as competing criteria for choosing between them. The mean score a system achieves across the collection's query set, under whatever metric is selected, is the measure of the system's effectiveness on that collection. But the collection, and particularly its queries, are only representative: what we care about is how predictive the observed score is of the system's effectiveness in general. To make such generalizations in a rigorous way requires the use of statistical methods, particularly tests of statistical significance. Statistical tools also have other applications in the field, such as comparing the system rankings induced by different evaluation methods, and summarizing result data in a human-comprehensible way.

Before test collection evaluation can be deployed, the test collection itself must be constructed. Ideally, every document in the corpus would be assessed for relevance to every query; in reality, exhaustive evaluation of even moderately-sized corpora is impractical. The standard solution is to assess only the pool of documents ranked highly by a representative set of retrieval systems. But such pools are potentially biased against subsequent, unpooled systems, and evidence suggests that this bias, though formerly slight, is becoming more serious as corpus sizes grow. Scoring methods have been suggested for handling the incompleteness of relevance assessments, along with more efficient techniques for locating relevant documents for assessment. Recently, attention has turned to sampling and inferential methods of system evaluation. The (pooled) test collections used in the thesis are those formed by the TREC effort, along with the runsets of systems participating in the official TREC experiments. We conclude Chapter 3, and the background section of the thesis, by describing and analyzing these, our experimental data sets.

The contributions of the thesis begin in Chapter 4, where we propose the method of score standardization to control topic score variability. Topics have enormously variable levels of difficulty, such that effectiveness scores typically differ more between topics than they do between systems. Put another way, the score that a retrieval system achieves on a topic tells us more about the difficulty of the topic than the quality of the system. Individual run scores are therefore largely meaningless in isolation; and, since collection scores are aggregated from run scores, the interpretation of collections scores is also fraught. Moreover, inter-system score deltas vary markedly between topics, meaning that some topics have much greater impact even on comparative scores than others—and frequently these topics are the easier ones, whose higher mean scores allow greater range for difference.

The standard solution to topic variability is to normalize scores by the maximum score achievable on the topic, given the number of (known) relevant documents. In practice, though, the maximum score is a poor indicator of topic difficulty, and normalization gives only a slight decrease in topic variability, as empirical results given in Chapter 4 show. Instead, we propose that topic difficulty and score variance should be empirically measured as the mean and standard deviation of the scores of a set of reference systems. Scores achieved on the topic, both by reference systems and by other systems, are then standardized by the reference factors, leading to identical score means and standard deviations for each topic on the reference set, and greatly decreased variability on other system sets. Standardized scores are immediately interpretable, even in isolation; a standardized score of 0, for instance, means "at the average of the reference set". By using a common reference set across different collections, standardized scores are made comparable between collections (a use to which standardization is put in Chapter 8). These features of standardization are demonstrated on TREC data. We conclude Chapter 4 with an examination of reference set dependency and outlier values, and propose compensatory transformations.

The importance of verifying the generalizability of experimental results through the use of a test of statistical significance is well established in information retrieval. Less understood, though, is the need to predict the reliability of proposed experiments. In particular, the experimental designer, whether using new or existing data, wishes to know how assured they are that, if a meaningful level of improvement is achieved by a new over a baseline system, this improvement will actually result in a statistically significant result—what is known as the statistical power of the test. If an improvement exists, but the experiment is not powerful enough to detect it, then not only is the experiment wasted, but a promising line of research may be neglected. In Chapter 5, we

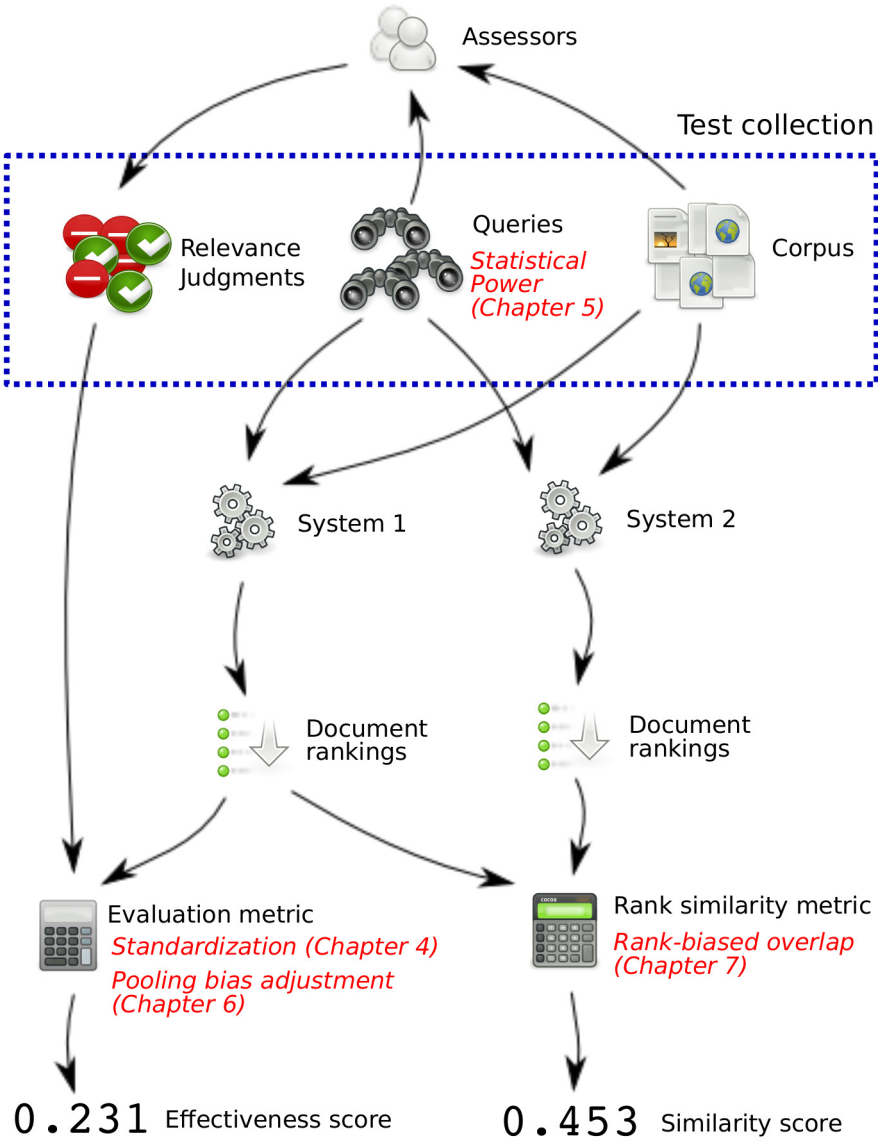


Figure 1.3: Schematic of retrieval evaluation using test collections, with summary of contributions of the thesis.

introduce statistical power analysis to information retrieval. Power analysis provides a tool for assessing the experimental discrimination of existing collections. We turn this tool on the TREC collections and runsets, and conclude that at least 100, and often closer to 200, topics are required to reliably separate a high-performing system from a reasonable baseline. The 50-query topic sets of standard TREC collections are not powerful enough to detect the incremental improvements achievable in a mature technology such as information retrieval.

The experimenter who is unable to use existing collections, due to their lack of power or to other causes of unsuitability, is forced to develop their own collection, at which point design-phase power analysis is essential. Statistical power is a function of several variables: the hypothesized true difference in performance between systems; the strictness of the significance test; the variability of score deltas across topics; and the number of topics used. Typically, the number of topics is the primary variable under the experimenter's control, while the main unknown variable is the standard deviation of inter-system score deltas. We demonstrate in Chapter 5 that delta deviation varies significantly across system pairs. Estimation of standard deviation based on past experience is therefore unreliable, while running a trial experiment sets quite wide bounds. An appealing solution is the incremental one of comparing systems on an increasing number of topics, refining estimates of standard deviation as we go, and continuing until the desired power is achieved. The incremental approach, however, leads to a subtle bias in favour of achieving statistical significance, which we demonstrate and empirically quantify in Chapter 5. For the experiment designer, creating a new topic set for a system comparison, we therefore propose a hybrid methodology. We conclude Chapter 5 by considering the question of whether greater power is achieved, for a given amount of assessor effort, by broad but shallow, or by narrow but deep, evaluation; we find that shallower evaluation, over a larger number of topics, gives considerably greater statistical power.

As discussed in Chapter 3, the exhaustive evaluation of every document for relevance to each topic is infeasible, and a pooling approach is used instead, in which only pooled documents are assessed, and unpooled documents are assumed to be irrelevant. Pooling is potentially biased against unpooled systems, since they may return unpooled but in fact relevant documents. Concern about such bias is growing with the increase in corpus size. In Chapter 6, we propose a robust, empirical solution to pooling bias. The idea is to observe the bias that a system suffers from being excluded from the pool, and adjust the system's score accordingly. The pooling bias against an unpooled system can be estimated by a leave-one-out experiment on the pooled systems, but such an estimate is only accurate to the extent that the unpooled system is similar to the pooled ones. A more reliable method is to pool the otherwise unpooled system, along with the fully pooled systems, on a common subset of queries. Then the unpooled system is held out of the pool for this subset, and the observed bias against it used to derive the adjustment factor. Experiments reported in Chapter 6 demonstrate that this method of score adjustment can reduce pooling bias by 75% with as few as 20 common topics. The proposed method of score adjustment is particularly appealing in a dynamic evaluation environment, such as the lab of a working search engine, where new retrieval methods and new topics are being continually added. New methods will be fully pooled and assessed, alongside existing techniques, against new queries as part of regular evaluation. These new queries can then be used as the common topics to adjust the new method's scores on old queries, without having to revisit the old queries' assessment.

Retrieval effectiveness is not the only basis for comparing the ranked document lists returned by search engines. In many circumstances, what matters is the similarity of

the results produced by different search engines, or by the one search engine at different times, without regard to effectiveness. Even where effectiveness is the ultimate concern, result similarity may be used as a cheaper proxy. An efficiency short-cut in query evaluation cannot be doing much harm to effectiveness if it does not greatly change the document rankings; and only those topics where rankings do change noticeably need to be assessed for effectiveness. To perform comparisons between ranked document lists requires a similarity measure that is suited to the peculiar characteristics of such lists. In Chapter 7, we identify three key characteristics: disjointness; top-weightedness; and arbitrariness of cutoff depth. Rankings with these characteristics we label as indefinite rankings, and a measure of similarity between them which embodies these characteristics is an indefinite rank similarity measure. The literature describes no suitable indefinite rank similarity measures. The standard rank similarity measures do not handle disjointness; and the (small) set of measures that do handle disjointness are either unweighted or non-monotonic in cutoff depth. Indefinite rankings are also found in many situations other than document rankings: indeed, they occur wherever a top-weighted ranking with finite cutoff is induced over an infinite (or very large) domain. Indefinite rank similarity measures therefore have a wide application.

In Chapter 7, we propose the first formally described similarity measure on indefinite rankings, which we call rank-biased overlap (RBO). Instead of correlation, the RBO measure is based upon set overlap, which is a more natural fit for disjoint rankings. The measure therefore handles disjointness readily. It is tunably top-weighted, allowing the experimenter to apply the degree of top-weighting dictated by their experimental context. And, by incorporating a convergent (geometric) series of rank weights, RBO is monotonic in cutoff depth; evaluation to a given depth sets bounds on the score achievable with further evaluation. We further derive a reasonable point estimate within these bounds, one which is consistent in its behaviour with increasing depth of evaluation. We show RBO's utility by using it to compare eleven different search engines, over 113 queries, submitted daily for four months. We also demonstrate RBO's superiority over other disjoint rank similarity measures in comparing experimental runs under efficiency optimizations or parameter tuning, and its tighter correlation with effectiveness scores on simulated data.

The thesis concludes in Chapter 8. We review the detailed, technical path that the thesis has beaten to its destination, then broaden our view for a survey of the state of information retrieval evaluation. A strength of the test collection methodology is its repeatability, allowing different systems to be evaluated under the same experimental conditions. Retrieval effectiveness can therefore be compared between systems and over time. Between-system comparisons are common, both at TREC and in published research, but over-time comparisons are scarce. The seemingly fundamental question of whether retrieval technology is improving over time has rarely been asked; we therefore ask it in Chapter 8. First, we employ score standardization to compare effectiveness in ad-hoc retrieval at TREC, finding no clear evidence of improvement since TREC 3 in 1994. Next, we survey published results on TREC collections over the past ten years. The collections are widely used, and the protocol of testing a new method against a baseline is generally applied, often with statistically significant results. Yet the published results show no upward trend in effectiveness; worse yet, the same, mediocre baseline scores are reported each year, and improved upon by the similar margins. Our finding raises serious questions about the supposed rigour of information retrieval evaluation.

Chapter 8 then turns to examining the current challenges and opportunities facing retrieval evaluation. These challenges include: extending the test collection method-

ology to handle the diversity and ambiguity of web queries, and the scale and dynamism of web corpora; accessing and incorporating user data, such as query logs and click-through records; assessing the potential of crowdsourcing as an evaluation resource; and propagating the achievements of retrieval evaluation into other fields. As impressive as the field's methodological achievements are, though, our finding of much method but little improvement over the past decade reminds us that a strong methodology is not an end in itself. Therefore, we conclude the chapter, and the thesis, with a critical appraisal of the sociology of methodology, and consider to what extent the influence of test collection evaluation is encouraging formulaic research and discouraging innovative ideas.

Chapter 2

Historical background

Information retrieval as a discipline has a fifty-year history, dating back to the beginning of computerization, when the potential for automating the indexing and retrieval of documents was first appreciated. Indeed, the earliest work in the field was done without the use of computers. From its beginning, the discipline has had a strong experimental tradition. Its focus on empirical validation and evaluation is one of the characteristics that distinguish information retrieval from its more theoretically-minded parent, information science. The history of information retrieval evaluation, the subject of this thesis, goes back as far as that of information retrieval itself. The discipline's strong experimental tradition had been one of its strengths. However, an excessive empiricism has arguably narrowed the field's focus. Meanwhile, the development of the web, and the importance of web search engines, pose great challenges to existing experimental methodologies.

In this chapter, we examine the history of information retrieval evaluation over the past half century. We begin in Section 2.1 with the Cranfield tests of the late 1950s and 1960s, which introduced a standard experimental methodology that the field has followed ever since. We also describe the SMART project, which took Cranfield's evaluation methodology and adapted it to computerization in the early 1960s. The SMART project did much to popularize the style of research and publication that now predominates in the area, as well as to develop the area's fundamental technologies of statistical document retrieval. In Section 2.2 we survey developments in the decades that followed the foundational period, notable for their meager contribution to evaluation methodology. We then describe the founding of the TREC project in the early 1990s, which achieved a hundredfold increase in the scale of the collections available to researchers, and did much to inspire a renaissance in experimental investigation of IR techniques—but also, some would argue, drawing attention away from pressing questions about the user search experience. Finally, in Section 2.3, we look at the impact that the rise of the web has had upon the field of information retrieval, and the challenges it poses to retrieval evaluation.

2.1 Foundations of IR evaluation

The field of retrieval evaluation was established by two foundational efforts in experimentation. The first is the Cranfield tests, undertaken in two main stages between 1957 and 1966. The second is the SMART project, starting in the early 1960s and running in

various forms until the end of the century. Although carried out by clerical rather than computerized means, the Cranfield tests established the standard experimental methodology for the field: system evaluation using a fixed test collection, consisting of documents, queries, and assessments of which documents are relevant to which queries. The SMART project adapted this test collection methodology to a computerized environment, and popularized a model of research and publication based on automated experiment that has persisted to today. Both projects were fundamental in establishing the field's experimental methodology, as well as its characteristic empiricism of attitude.

2.1.1 Cranfield

The phrase "information retrieval" was coined in the 1950s (Robertson, 2005), but the origins of the concept go back to the first library catalogues. Early thinking on the subject emerged from librarianship and its theoretical arm, information science. This early thinking was primarily philosophical in nature, concerned with how information should be classified and organized. Different schools held different positions, and debate was carried out between them on philosophical and anecdotal rather than empirical grounds (Robertson, 2008a). However, with the burgeoning volume of publication, and particularly of scientific literature, after the Second World War, practical concerns of how to effectively access this literature became urgent (Cleverdon, 1991; Luhn, 1957). In 1952, it was estimated that three quarters of a million scientific and technical articles were being published annually (Wilson, 1952, page 10), and by 1960, researchers were warning that (Maron and Kuhns, 1960, page 217):

> documentary data are being generated at an alarming rate (the growth rate is exponential—doubling every 12 years for some libraries), and consequently consideration of volume alone make the problem [of retrieval] appear frightening.

The first rigorous experiments in information retrieval were those carried out in the library of the Cranfield Aeronautical College under the direction of the librarian, Cyril Cleverdon. The aim of these tests was to determine the most effective means of indexing and retrieving documents, in particular scientific papers of the sort held in the Cranfield library. The tests were performed in two main stages. The first stage ran from 1957 to 1961, and is commonly called Cranfield 1, while the second, Cranfield 2, ran from 1963 to 1966 (Cleverdon, 1962; Cleverdon et al., 1966; Cleverdon and Keen, 1966; Spärck Jones, 1981b). Cleverdon and his colleagues faced several challenges in carrying out their experiments. Not the least of these was that the experiments were not computerized; they were carried out manually, and quite laboriously, with indexes written out by hand as card catalogues and searches performed by cross-referencing information on these card indexes. As pioneers in the field, the Cranfield project also had to develop an experimental methodology; their multi-volume reports attest to the seriousness with which they took this responsibility. Two of the crucial questions for any such methodology were, first, how to generate the retrieval requests for experimental use, and second, how to determine whether a retrieval request had been successfully answered by the output of a search.

The approach taken to retrieval request formulation and resolution in Cranfield 1 was a straightforward one. From the collection of $18{,}000$ papers on aeronautical science that formed the test corpus, a set of source documents was selected; and for each

source document, several questions were framed by domain experts, to which that document was a satisfactory answer. A total of 1,200 questions were generated in this way. Then, in the experiment, a search on a request was deemed successful if it retrieved the source document (Cleverdon, 1962). Although it bears some resemblance to what is now called known-item search, the source document method of query formulation used at Cranfield 1 was criticized at the time as artificial, and so a new method of request generation and assessment was devised for the second round of experiments (Spärck Jones, 1981b).

The approach taken at Cranfield 2 again started with the selection of a technical paper from the corpus. This time, the author of each such paper was asked to write down the original research question that had inspired the paper; this served as the request. Some 225 requests were generated in this way. However, the originating document was no longer the target of the search, and was removed from the document corpus. Instead, the other papers in the corpus—1,400 of them—were examined to determine whether they were *relevant* to the research question or not. The full set of papers was filtered by Cleverdon's research students. Only those papers that appeared likely to be relevant based on their title, plus some other papers arrived at by bibliographic means, were sent to the author of the research question for relevance assessment, with the remainder summarily classed as irrelevant—a less than exhaustive method of assessment, which attracted criticism later (Salton, 1992). Nevertheless, perfect or not, the result of the assessment was that every paper in the test corpus was marked as relevant or irrelevant to every request. Then the searches were performed. In Cranfield 1, these had been continued until the source document was located. However, at Cranfield 2, a list of papers that matched each request, according to each of the indexing schemes under examination, was returned. The correctness of this result, and hence the effectiveness of the retrieval, was then evaluated using two complementary metrics: *precision*, the proportion of returned documents that were relevant; and *recall*, the proportion of relevant documents that were returned (Cleverdon et al., 1966; Spärck Jones, 1981b).

It was partly for operational reasons that, at Cranfield 2, the set of documents relevant to each request was determined in advance of retrieval: the searchers were not qualified to make relevance assessments, and domain experts were not on hand to assess the documents as they were returned. Automation of assessment was at best a minor consideration, as request processing was still performed manually. However, the model proved to be well suited to computerized retrieval and the automation of its evaluation. With the relevance judgements performed in advance, no further human involvement was required for assessment; and, once the indexing and retrieval process had been computerized, repeated experimental runs could be made and evaluated automatically. The approach pioneered at Cranfield 2 has become the standard model for evaluating information retrieval systems, in part due to its automatability (Voorhees and Harman, 2005b). This standard model can be defined as using a fixed test collection made up of a *document corpus*, a set of queries or *topics*, and assessments of which documents are relevant to which queries, which are known as *qrels*. The retrieval system runs the queries against the document corpus, and for each query it returns a list of matching documents. Each document list is marked up for relevance using the qrels, and system effectiveness is then calculated using relevance-based metrics, such as precision and recall. The system receives a score for each topic, and the per-topic scores are aggregated, typically by taking the arithmetic mean, to provide a system score for the collection as a whole. Because of its origins, the relevance-based test collection model is frequently referred to now as the "Cranfield methodology" or even in recent years as the "Cranfield paradigm" (Voorhees, 2002; Buckley and Voorhees, 2004).

The Cranfield tests did much to inspire information retrieval's empiricism of methodology, and to provide the content of this methodology. The results of the tests, more subtly, also had a strong influence of the field's empiricism of attitude. To appreciate why, we need to understand something of the theoretical context in which the tests were performed. At the time, the dogma of information science was that the effective retrieval of information required sophisticated indexing schemes (Spärck Jones, 1981b). Documents had to be classified into topics, which in turn were arranged in taxonomic hierarchies. Controlled vocabularies were employed to select index terms; or terms were dispensed with altogether, and decimal subject codes used instead. Various schools of thought existed as to which classification schemes and indexing methods should be employed, but the debate between them was carried out more on philosophical than on empirical terms. Indeed, indexing and classification was as much pedagogic and prescriptive as it was functional: it specified how information should be organized in principle, not how it might effectively be accessed in practice (Robertson, 2008a).

By subjecting different indexing philosophies to empirical evaluation, the Cranfield tests ran against the contemporary speculative grain. Nevertheless, the focus of the tests was in accordance with current beliefs about the centrality of indexing languages. Cranfield 1, titled in its report "an investigation into the comparative efficiency of indexing systems", compared four different indexing languages (Cleverdon, 1962); and Cranfield 2, self-described as "tests on index language devices", identified several indexing components, such as taxonomic links and synonyms, which were combined to create 33 different index languages for evaluation (Cleverdon and Keen, 1966; Cleverdon, 1967). Part of Cranfield 2's elaborate indexing process can be observed in Figure 2.1.

Just as the empiricism of the Cranfield tests was alien to the dogmatic attitudes of indexing theory, so too the results of the tests were a direct challenge to its preconceptions (Salton, 1992). Cranfield 1 failed to detect significant differences in effectiveness between the different indexing languages, despite their different theoretical foundations (Cleverdon, 1962, Chapter 9). Then Cranfield 2, with its careful delineation of different indexing techniques and devices, found that in fact selecting plain index terms, such as could be found directly in the text, outperformed the concept-based and thesaurus-controlled languages that classification theory held to be essential. This was a surprising and controversial result, as Cleverdon notes in his report (Cleverdon and Keen, 1966, page 252):

> Quite the most astonishing and seemingly inexplicable conclusion that arises from the project is that the single term index languages are superior to any other type. ... This conclusion is so controversial and so unexpected that it is bound to throw considerable doubt on the methods which have been used to obtain these results ... A complete recheck has failed to reveal any discrepancies, and ... there is no other course except to attempt to explain the results which seem to offend against every canon on which we were trained as librarians.

The Cranfield tests are an instance of the triumph of empiricism over theoretical speculation. Long-held, intuitively attractive beliefs about how information should be classified were demonstrated not to be true—or at least, not to be useful. Information retrieval has ever since been a deeply, even stubbornly empirical field. It is famously difficult to publish in the field, no matter how clever or compelling the ideas and theories might seem, without at least the appearance of thorough experimental validation.

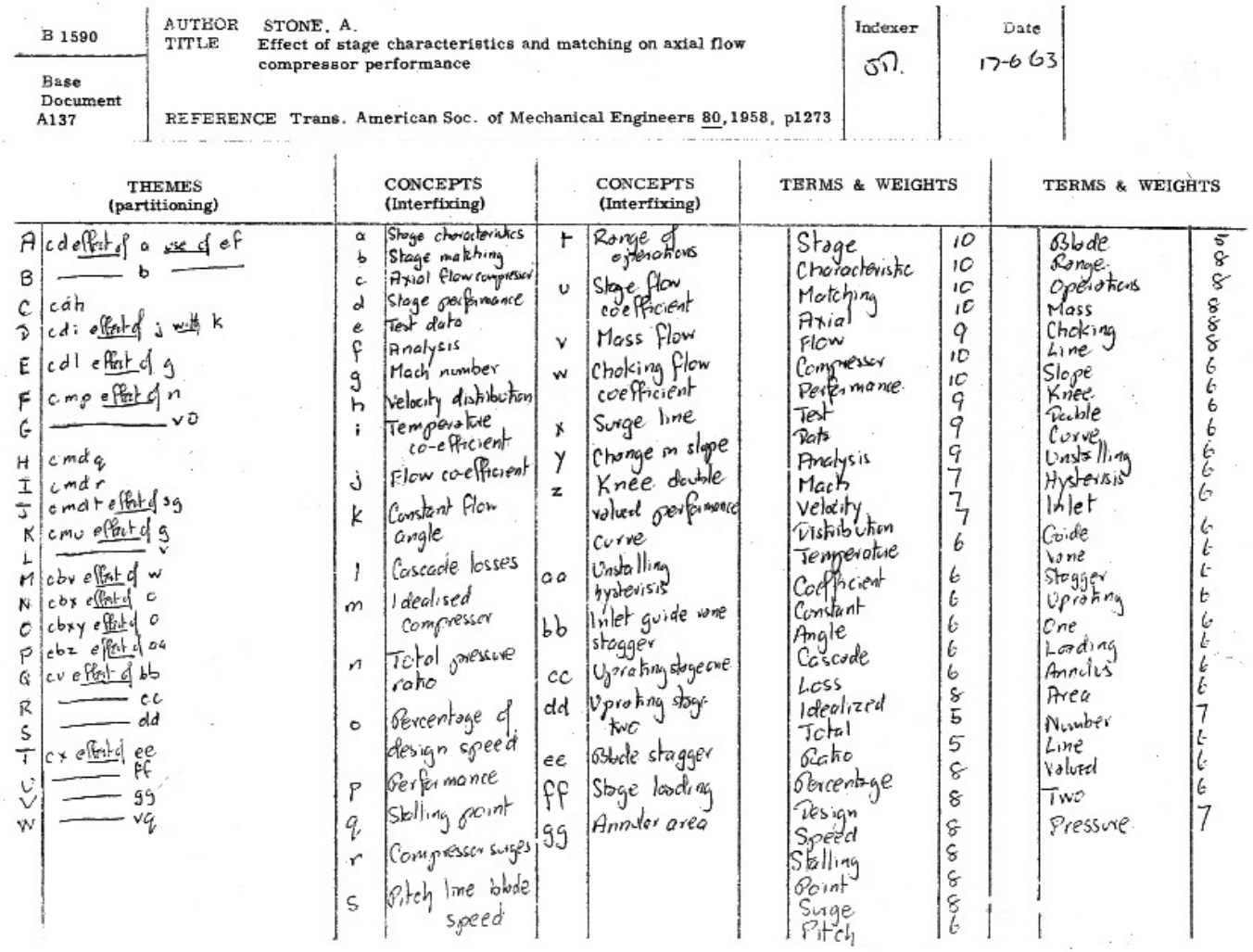


Figure 2.1: Indexing sheet for Document 1590, from the second Cranfield experiment. The indexer first records concepts (columns 2 and 3) found in the document. Concepts are noun phrases, identified by lower case letters (left marginal columns). Then, the indexer joins concepts together to create themes (column 1), which are topics occurring in the document. For instance, the theme **M** can be expanded to "axial flow compressor – stage matching – mass flow – *effect of* choking flow coefficient". Finally, single concept terms, excluding stop words such as "of" but including terms such as "one" and "two", are listed and assigned weights (columns 4 and 5). Such indexing sheets were only the first stage of the indexing process; devices and thesaurus controls were built upon them to generate the 33 different indexing languages. (Reproduced from Cleverdon and Keen (1966, page 51).)

The legacy of Cranfield for information retrieval is not simply the existence of an experimental methodology; it is the requirement that experimental evaluation be performed before work is to be accepted.

2.1.2 SMART

If the Cranfield experiments seem backwards-looking from today's perspective, with their manual execution and focus on indexing languages, the experiments of the SMART project feel surprisingly modern. Initiated by Gerard Salton and his collaborators in the early 1960s, first at Harvard University but soon moving to Cornell, and continuing in one form or another to the late 1990s, SMART was from the beginning an exploration of the possibilities of statistically-based, fully-automatic indexing and retrieval (Salton and Lesk, 1965), as first proposed by Luhn (1957). Evaluation at SMART used the test collection model developed at Cranfield, and as soon as the Cranfield collection became available in machine-readable form it was incorporated into the SMART test suite. SMART added an important innovation both to retrieval method and evaluation methodology: that the answer to a search should be a list of documents ranked in decreasing degree (or probability) of relevance, rather than an unordered set of exact matches. The notion of ranking arises naturally from that of statistically-determined degrees of match between a document and a query (Rocchio, 1966, Chapter 4), and is of course familiar from present-day web search engines.

At the time of SMART, computing resources were expensive. A single run of 225 queries on the $1{,}400$-document Cranfield collection in 1973 took 11 minutes and cost 86 dollars—equivalent to a professor's daily salary (Robertson, 2008a, page 443). Nor were computers particularly convenient to use, as Lesk's laborious 63-page 1966 manual for the SMART system attests (Lesk, 1966). In 1962 Cleverdon cited as one of the advantages of manual retrieval that it was so much faster than automatic methods: "it has been a matter of surprise to find the time delay which many organizations appear willing to tolerate for the doubtful benefit of using some form of mechanical retrieval" (Cleverdon, 1962, page 88). Nevertheless, SMART demonstrated the possibilities of fully automated evaluation: precisely repeatable experiments, formal control over experimental variables, and the automation of the experimental process itself. Salton describes the SMART system's functionality as being "not only for language analysis and retrieval, but also for the evaluation of search effectiveness by processing each search request in several different ways while comparing the results obtained in each case" (Salton, 1966a, page 1). Of course, to realize the potential of a fully-automated evaluation system, one needed an evaluation methodology that could be fully automated—and this was precisely what was provided by the test collection model developed at Cranfield and adopted by SMART.

Not just the methods and goals, but also the publications of the SMART project feel modern compared to those of Cranfield. Cranfield produced large, multi-volume reports; books, essentially. The SMART publications, however, followed the scientific paper model: shorter, self-contained papers, with the common pattern of introduction, literature review, theoretical exposition, and experimental evaluation (Salton, 1966b, 1971). And, of course, the experimental evaluation used was the test collection methodology that SMART developed.

The SMART project also parallelled Cranfield in its confrontation of theoretical dogma with empirical fact. Whereas for Cranfield the dogma was the information science one of classificatory indexing, for SMART the presumptions came mostly from the field of linguistics. As Salton (1981, pages 317–318) later described it:

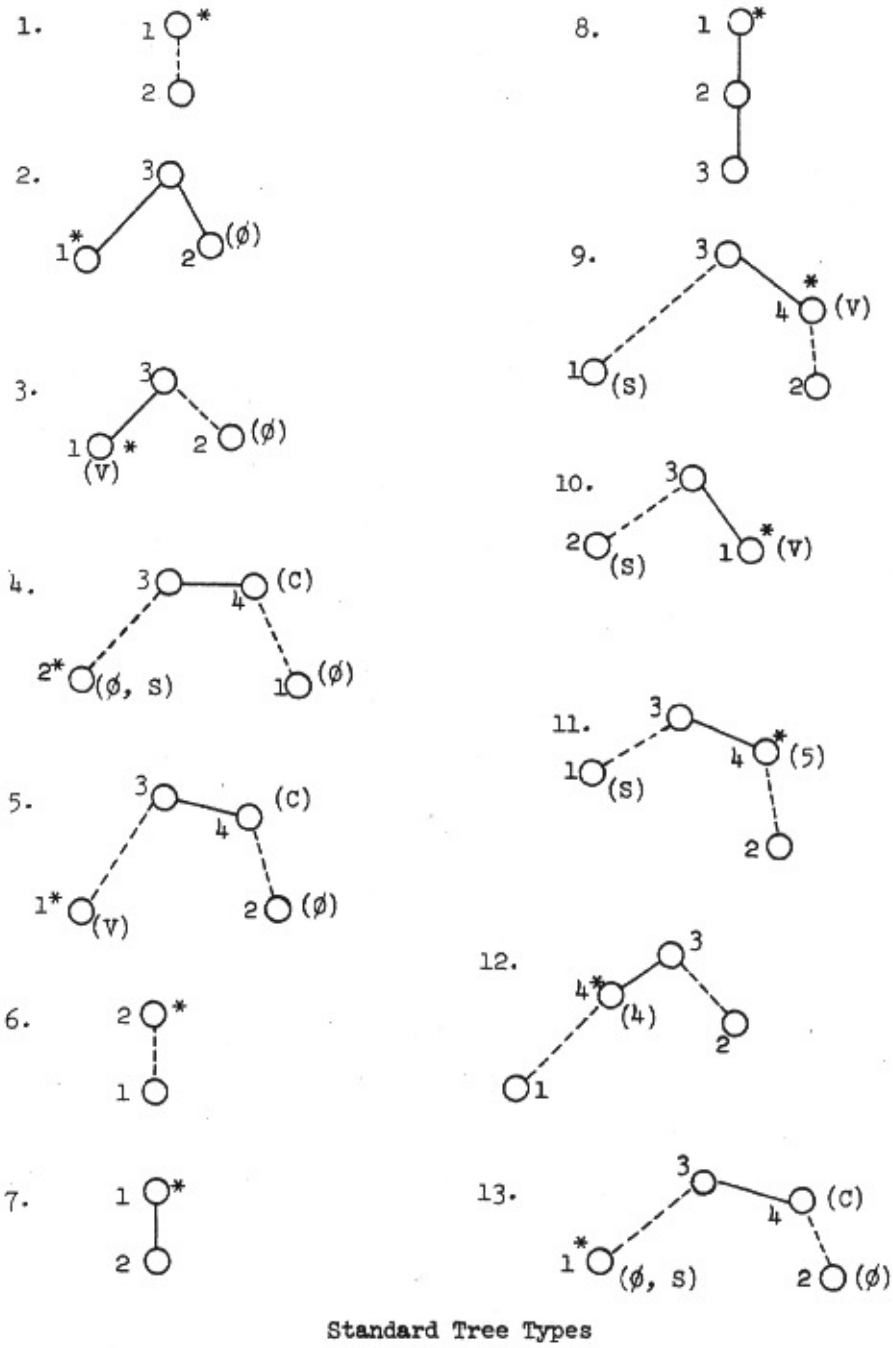


Figure 2.2: Standard syntax parse trees from SMART, 1966. These trees allowed the user to specify simple syntax parsing and matching in indexed documents. The asterisked node is the matched concept. Solid lines indicate direct dependence, dashed indirect. Labels in brackets mark parts of speech; for instance, "(V)" is a verb, "(Ø)" an object, "(5)" an object clause. The user was also able to specify their own syntax parse trees. (Reproduced from Lesk (1966, page 47).)

> Thus linguists led the way by pointing out that a number of linguistic processes were 'essential' for the generation of effective content identifiers characterizing natural language texts. Among the linguistic techniques of interest, the following were considered to be of greatest importance: (a) The use of hierarchical term arrangements ... (b) The use of synonym dictionaries, or thesauri ... (c) The utilization of syntactic analysis ... (d) The use of semantic analysis.

The SMART system was developed to incorporate these advanced linguistic features—some of them can be seen in Figure 2.2—and the early SMART experiments set out to test how useful these features were. Contrary to expectations, they found that such linguistic devices did not raise performance over that achieved with simple natural language terms (Salton, 1981). As with Cranfield, this was at first taken to indicate a flaw with the system or experimental design, so strong was the prior presumption that linguistic devices had to improve retrieval effectiveness. But the results of Cranfield and of SMART, arrived at by such different mechanisms, helped confirm each other, and were further confirmed by later experiments. The SMART experiments are another example of empiricism trumping theory.

The SMART project continued for three and a half decades, under the stewardship of Gerald Salton, Chris Buckley, and collaborators. Alumni of the project went on to play important roles in the TREC effort, in academic research, and in the development of commercial web search engines. The legacy of Cyril Cleverdon also led to a strong tradition of IR research in the United Kingdom, with foundational work being done in document similarity measurement, in retrieval theory, and in the design of test collections, by figures such as Karen Spärck-Jones, Steve Robertson, and Keith van Rijsbergen. These and other early efforts led to the development of the discipline of information retrieval, and the establishment internationally of researchers and research centres too numerous to name here.

2.1.3 Experimentalism and empiricism in IR

Cranfield, SMART, and other early projects laid the foundations that have determined the structure of the field of information retrieval. They did this in two ways. First, they established an influential experimental methodology, of a fixed test collection containing static relevance judgements, a set-up ideally suited to automated retrieval experiments. Second, they were exemplars of empirical caution against theoretical presumption, through their important, negative results. Cranfield and SMART proved that a series of ideas the theoreticians asserted must work in fact didn't, and that the simple methods the theoreticians assumed were insufficient actually sufficed.

As it happens, information retrieval technology has not changed greatly in its approach since these early days, at least not on traditional text collections; simple term indexing as validated by Cranfield and statistical models as developed by SMART are still the mainstay of automatic retrieval technology, and successful elaborations of them have been few in number and only marginal in effect (Spärck Jones, 2000). There has also been, to the frustration of some, a failure to develop information retrieval as a predictive and explanatory science, rather than one that is merely descriptive (Belkin, 1981; Robertson, 2009). This narrowness in the field's scientific vision can in part be traced back to the results of Cranfield and SMART: having instrumented a series of theoretically-proposed techniques in a way that they could be manipulated as experimental variables, the studies found that in fact the techniques did not work as theory

said they should. Since then, many experimenters have tended to view the retrieval system as a black box: input goes in, results come out, the performance is measured, but, often, no serious attempt is made to predict the system's behaviour beforehand or explain it after (Robertson and Hancock-Beaulieu, 1992). The black box model itself is an example of trenchant empiricism.

2.2 The origins and influence of TREC

Having described the beginnings of IR evaluation in the Cranfield tests and the early years of the SMART project, it would be all too easy for a historical narrative to jump forward a quarter century to the foundation of the TREC effort in 1992. The annual TREC experiments and the large-scale test collections they produce have come to dominate the IR evaluation landscape, leaving the preceding decades in an obscure shade. The TREC organizers themselves explicitly link their project back to Cranfield, introducing the phrase "the Cranfield paradigm" into widespread usage to describe the methodology used at TREC (Voorhees, 2002), and noting the evaluation efforts of the intervening years mostly for what, methodologically, they failed to achieve (Harman, 1992b). In terms of the goals of TREC, there had indeed been little progress made over the previous two decades. But at the same time, to understand both the achievements and the limitations of TREC and the view of IR evaluation that it embodies, we need to understand what happened in these evaluative middle ages.

2.2.1 Evaluation's middle years

The first Cranfield experiments inspired a range of other investigations into index languages and devices during the 1960s, of variable quality and tending to come either to similar conclusions as Cranfield or to no reliable conclusions at all. Due largely to the example of Cranfield's meticulous attention to detail, as well as to the subsequent availability of the Cranfield collection, the methodological quality of experiments improved during the 1970s (Spärck Jones, 1981a). The SMART project and the increasing accessibility of computing resources also inspired work on statistical models and automated retrieval, including ongoing work at SMART itself (Salton, 1981).

Despite these developments, in 1975 Spärck Jones and van Rijsbergen characterized the work to date as being only "pilot studies" (Spärck Jones and van Rijsbergen, 1975), and in 1981 Spärck Jones remarked that "the most striking feature of the test history of the past two decades is its lack of consolidation" (Spärck Jones, 1981a, page 245). This failure to consolidate research findings was ascribed to inconsistency in method, the absence of a common experimental framework, and the expense of conducting large experiments. There was a strong inclination to re-use collections that others had formed, with the Cranfield collection being particularly popular (Spärck Jones and van Rijsbergen, 1976); but these collections had been created for particular purposes, were small, and were of variable quality.

As a solution to methodological failures of early experiments and the resultant lack of standardization, in 1975 Sparck Jones and van Risjbergen proposed the formation of an "ideal" test collection (or, to be precise, collections) (Spärck Jones and van Rijsbergen, 1975). There were several criteria that they laid out for such a collection. One of these was scale: it needed 10,000 to 30,000 documents, and it needed a few hundred queries. At the time, many experiments were being carried out with as little as a few hundred documents and a few dozen queries (Spärck Jones and van Rijsbergen, 1975, Ap-

pendix B). A careful mixture of homogeneity and heterogeneity was stressed, to allow for the control and manipulation of experimental variables such as document content and authorship. Costing of the project was incomplete and varied depending upon the assumptions made (whether documents could be purchased in electronic form from an operational system, for instance, or whether they would have to be typed in), but total costs for collection construction on pessimal assumptions (ignoring maintenance) were estimated to be as high as $US 200,000; close perhaps to $US 800,000 today. Perhaps because of the price tag, the "ideal" test collection was not formed—or at least, some would say, not until TREC (Harman, 2005a).

While the 1970s saw some improvement in methodology, despite the failure of the "ideal" collection proposal, it did not see any great growth in the number of experiments performed. Spärck Jones commented in 1981 that there were as many retrieval tests performed in the decade before 1970 as there were in the decade after (Spärck Jones, 1981a, page 245), and that in the latter 1970s "there has been a noticeable decline in the number of laboratory experiments" (Spärck Jones, 1981a, page 230). There were an increasing number of operational, proprietary retrieval services, and the success of these services gave the impression that the information retrieval problem had been solved (Spärck Jones, 1981a, page 230). Ironically, at the same time, these services were not adopting the retrieval techniques being developed by the research community, such as statistical matching, ranked output, and relevance feedback, but were sticking with older technologies such as Boolean retrieval carried out by trained searchers (Salton, 1981). The failure of industry in the 1970s and 1980s to adopt ideas developed in research led Salton to characterize these as his "lean decades" (Saracevic, 1995). There was also a belief among some researchers that automatic retrieval technology had reached a plateau, and that the batch mode of system evaluation using pre-built test collections was losing its relevance with the development of interactive, online systems (Oddy, 1981).

In a reaction against traditional batch-mode evaluation, the 1980s saw increasing interest in alternative evaluation modes, particularly those focused on user behaviour and interactive systems (Saracevic et al., 1988). In Jean Tague's 1981 essay, "The pragmatics of information retrieval experimentation", she makes frequent reference to the example and methods of Cranfield (Tague, 1981); but when she came to revisit the topic in 1992, she observed that a "paradigmatic shift has occurred in the research front, to user-centred from system-centred models" (Tague-Sutcliffe, 1992, page 467). The Okapi project was carrying out user experiments on an operational, interactive retrieval system installed in a working library, complete with keystroke logging and user interviews (Robertson and Thompson, 1990). These experiments inspired Robertson and Hancock-Beaulieu to call for a move from experiments based on test collections to experiments centred on evaluation facilities (Robertson and Hancock-Beaulieu, 1992). Of course, work was still being performed using what we might now term the "old" test collection model, not least as part of the ongoing SMART project. But even here, evidence of the atrophy of the model can be seen in the age of the test collections employed. Take, for instance, the six test collections used in Salton and Buckley (1990), (Table 2.1). Of these, the most recent was formed in 1979, more than a decade earlier; most predate 1975; and the earliest is the original Cranfield collection from 25 years before. Researchers working on batch-mode automatic retrieval in 1990 were substantially still using the same test collections whose inadequacy Sparck Jones and van Rijsbergen had bemoaned fifteen years earlier in 1975.

The age of the collections being employed in 1990 was not the most glaring problem with them, though. Rather, it was their size. As can be seen from Table 2.1,

Collection	Documents	Queries	Year
Cranfield	1398	225	1966
INSPEC	12684	84	1970
NPL	11429	100	1970
Medlars	1033	30	1973
CISI	1460	112	1977
CACM	3204	64	1979

Table 2.1: Size and year of formation (or first description) of collections being used by the SMART project in 1990 (Salton and Buckley, 1990). Collection dates are derived from Fox (1983) and Spärck Jones and van Rijsbergen (1975).

the SMART project was working with collections of between a thousand and twelve thousand documents. Even in 1981, when operational systems were indexing millions of records (Salton, 1981), this scale of collection had been regarded as inadequate (Oddy, 1981).[1] The small scale of the test collections being used experimentally was one of the major reasons operational systems gave for viewing laboratory research results with scepticism, and for not adopting them in practice (Salton, 1981; Ledwith, 1992). As well as being small, the existing collections were limited in the breadth of their content. The vast majority of them—including all of those listed in Table 2.1—consisted of scientific papers (Robertson, 1981). Moreover, these collections predominantly (though not entirely) contained not the full paper text, but titles and abstracts alone (Spärck Jones and van Rijsbergen, 1976).

2.2.2 The Text REtrieval Conference

The TREC effort, founded in 1992 and continuing to the present, took as its prime motivation the need to create realistically-sized collections, and to consolidate and extend research in retrieval technology through the use of shared, high-quality experimental data and standard evaluation techniques (Prange, 1996; Harman, 1992b). The effort is hosted by the National Institute of Standards and Technology (NIST), a US government agency. The first TREC test collection was taken from the slightly earlier TIPSTER effort, launched in 1989 with DARPA funding to advance the state of the art in document detection and information retrieval (Harman, 1992a). The TIPSTER corpus contained around $750{,}000$ documents, an almost hundredfold increase over existing corpora. Additionally, whereas previous collections normally held only paper abstracts, the TIPSTER corpus was mostly full text, making up 2 GB of text in total. The documents were drawn from several different sources; the majority were newswire items, but there were also magazine articles, parliamentary records, and scientific abstracts (Harman, 1992b).

The TREC effort represented an innovation not just in collection size, but also in experimental method. It was a large-scale, collaborative, and comparative experimental exercise, open to anyone wishing to take part, with dozens of international research teams participating (Voorhees and Harman, 2005a). Participants were provided with

[1] Salton's estimate is surprisingly high; but since MEDLARS was already indexing over 150,000 citations a year in 1965, and since MEDLINE, brought online in 1971, inherited the MEDLARS database from 1966 onwards, Salton's figure seems reasonable, at least in the case of the MEDLINE database. See `http://www.nlm.nih.gov/databases/databases_oldmedline.html` (last downloaded 21st September, 2009).

Figure 2.3: TREC assessors at work for TREC 7 in 1998. Standing is Ellen Voorhees, the chief TREC co-ordinator. Document relevance assessment for the main TREC collections was carried out at NIST by retired intelligence assessors. A pool of candidate documents is formed from the submitted runs of the participating systems. Each of these documents has to be assessed for relevance to the query it was returned for. As many as 2,000 documents had to be assessed for relevance for each query in the collection. (Image taken from `http://www.itl.nist.gov/iad/photos/assessors.html`; reproduced with permission.)

the document corpus and a set of topics to run against it; the runs for each topic were then submitted to TREC for assessment. Evaluation was based on the assessed relevance of documents to queries, but here TREC made two crucial contributions. First, the resources available to TREC allowed for large scale relevance assessment to be performed (see Figure 2.3); the expense of such assessment had done much to constrain the size of earlier collections (Salton, 1981). And second, the top-ranked documents returned by participating systems were pooled to create the set of documents for relevance assessment, as suggested in the 1975 proposal for an "ideal" test collection (Spärck Jones and van Rijsbergen, 1975). It was argued that by merging such a diverse range of inputs, the set of documents most likely to be relevant to each query would be identified. Thus, exhaustive assessment of the full collection, beyond the resources even of the TREC effort, could be avoided, while (it was hoped) still achieving a tolerably complete coverage of the relevance set, allowing the collection to be re-used by researchers in subsequent experiments (Zobel, 1998).

The scale of its collections and the breadth of its participation made TREC the test collection model writ large. Not surprisingly, information retrieval experiment was dominated for more than a decade by the TREC effort, the experimental data it provided, and the experimental methods it pursued. The standardization of methodology allowed for discoveries in retrieval technology to be consolidated, and the large scale and high quality of the collections rendered the demonstration of those discoveries

credible. TREC provided compelling confirmation of earlier findings in information retrieval: that simple statistical methods performed at least as well as complex linguistic ones, for instance, and that query expansion and statistical phrase detection offered real but only slight improvements over plain weighted term matching (Spärck Jones, 2000). And, finally, TREC provided experimental data—lots of it. Hundreds of papers have used TREC test collections (Armstrong, Moffat, Webber, and Zobel, 2009a); a search on Google Scholar for the phrase "TREC data" produces over $1{,}400$ references at the time of writing. In addition to the test collections themselves, TREC has made large amounts of a novel kind of data available: namely, the document rankings or *runsets* submitted by participating groups. These runsets have provided the material for research on evaluation itself, and have inspired a large volume of such work over the past decade, of which this thesis is part.

The TREC effort is now in its 19th year. It began with a single collection of predominantly newswire data, and two retrieval tasks. The first of these was that of ad hoc retrieval for once-off queries; the second was routing or classification of documents based upon an initial set marked as relevant. The inaugural experiment saw the participation of twenty-five different research groups, already a large number. As the value of a collaborative experimental environment proved itself, new experimental tracks and tasks were added, and the number of participating groups grew. In 2005, some 117 different research groups participated in TREC, across seven different tracks, including genomics, question answering, and spam detection (Voorhees, 2005a). Participant numbers have declined slightly in subsequent years, but this is at least in part due to the emergence of regional and specialist collaborative experiments. Inspired by TREC, these collaborations include NTCIR, founded in 1999, which focuses on Asian language and cross-language retrieval (Kando, 1999); the Cross-Language Evaluation Forum (CLEF), started in 2000, which focuses on European language and cross-language retrieval (Peters, 2000); and the Initiative for the Evaluation of XML Retrieval (INEX), begun in 2002 (Gövert and Kazai, 2002).[2]

2.2.3 TREC and ad hoc retrieval

The primary initial task in TREC was *ad hoc retrieval*: retrieving documents, based on textual evidence alone, for once-off user queries, without other contextual information. Ad-hoc retrieval has long been the core task in information retrieval, and evaluation methods directed towards it are the main focus of this thesis. One of the main initial results of TREC was to confirm earlier findings in ad hoc retrieval, most particularly the viability of statistical retrieval techniques on large collections. It is less clear, though, what further improvements to ad hoc technology the TREC effort enabled. Organizers (Voorhees and Harman, 2000) and participants (Buckley et al., 1996; Robertson, 2008a) assert that the TREC regime led to significant improvements in retrieval effectiveness, at least in the early years of the effort; but how long this improvement continue for is uncertain. Aside from the fine-tuning of similarity metrics and advances in engineering, the main source of improvement in early TREC results seems to have been the introduction of a document length normalization component to the long-standing *tf-idf* (term frequency, inverse document frequency) model. The need for length normalization had not been apparent with earlier test collections, simply because they consisted

[2]NTCIR originally stood for "NACSIS Test Collection for Information Retrieval", then "NII Test Collection for Information Retrieval", and most recently "NII/NICT Testbeds and Community for Information access Research", after the successive sponsoring organizations, all of them Japanese government research institutions.

of abstracts of similar lengths (Spärck Jones, 2000, page 62). Document length normalization was key to the successful Pivoted Cosine and BM25 retrieval formulae. Once this adjustment had been made, it is open to question whether the TREC effort enabled further substantive improvements in ad hoc retrieval technology, at least as measured by the TREC collections themselves. We examine this issue in Chapter 8.

It is possible to see TREC as finally fulfilling the 1975 blueprint of an "ideal" collection. Certainly, the TREC collections more than met Sparck Jones and van Rijsbergen's criterion for size of document corpus. There is, though, at least one significant aspect in which TREC fell short of the "ideal" requirements, and that is in the number of queries. The 1975 report stated (in its highly abbreviated language) that "re requests: < 75 are of no real value; 250 are minimally acceptable; $> 1{,}000$ are needed for some purposes" (Spärck Jones and van Rijsbergen, 1975, page 64). However, the number of queries produced for each year's test collection at TREC was only 50, and while queries accumulated from year to year, the document corpus frequently changed, too. So while the TREC document corpora represented a hundred-fold increase on the earlier collections reported in Table 2.1, the topic set sizes were a retrogression; even the Cranfield collection contained 225 topics. Indeed, it was not until 2004 that any TREC test collection contained the 250 queries deemed in 1975 to be "minimally acceptable", this threshold finally being achieved by the TREC 13 Robust test collection, which incorporated topics from TRECs 6, 7, 8, and 12. In a review of the first half dozen iterations of TREC, Sparck Jones remarks that the query set size of 50 "is not as large as is really desirable" (Spärck Jones, 2000, page 54), and the question of whether the TREC query sets are adequate in size has been a recurrent one (Zobel, 1998; Voorhees and Buckley, 2002; Sanderson and Zobel, 2005). Most analysis of the question has been post-hoc, looking at existing test collections and the runs made by participating systems against them. In Chapter 5, we examine the issues involved in trying to determine in advance how many queries are necessary to reliably demonstrate a meaningful difference in effectiveness between two retrieval systems.

For all its achievements in standardizing methodology and increasing the scale of experimental data, the influence of TREC on the practice of retrieval evaluation has not been unequivocally positive. As we have seen, at the time of TREC's founding, the batch-mode test collection model of evaluation that it adopted was in decline, in favour of more user-centric and interactive approaches. The greater interest in interactive modes of evaluation was driven in part by the increased availability of online, interactive retrieval systems; users were no longer forced to use batch retrieval services, so why should researchers continue batch-mode evaluation? And the trend towards interactivity was about to accelerate with the arrival of the web. In some ways, then, while TREC represented a leap forward in volume of data, it represented a step backwards in breadth of experimental vision (Blair, 2002). The Okapi project team, for instance, notes in their report for TREC 1 that the lack of interactivity and highly complex topic specifications "does not at all represent the kind of retrieval activities for which Okapi was designed" (Robertson et al., 1992). Nor should this remark be interpreted as sour grapes: Okapi was one of the most successful participants in TREC. An attempt was made to run an interactive track at TREC, but after numerous problems it was eventually abandoned. Proposals were put forward to increase the contextual realism of the TREC experiments—for instance, by attempting to explicitly capture environmental variables and user background—but they were not adopted (Spärck Jones, 2000). The TREC effort proved not to be the appropriate forum for interactive and user-focused studies (Robertson et al., 2000).

The TREC approach also prolonged the black-box model of evaluation, of purely

descriptive experiments that reported what score a method achieved, rather than explanatory or predictive glass-box experiments that looked inside the system to provide a convincing theoretical model of information retrieval (Spärck Jones, 2000; Robertson, 2008a). And these failures of TREC in interactivity, user-focused study, and explanatory science were all the more significant because of TREC's tremendous success in providing tools and methods for batch-mode evaluation. TREC's system-centric success, and the dependable model of investigation and publication that it provided, arguably has drawn research focus away from other, user-centric imperatives. As Robertson (2008a) observes:

> It is very much easier for (say) a PhD student in the field to work on mathematical models and ranking algorithms, using the TREC material in the usual way and never questioning the validity of relevance judgements, than to venture into the jungle of real users with real anomalous states of knowledge.

The irony of quoting this passage in a thesis concerned with evaluation based on static relevance judgments is not missed by the author. We return to the issue of TREC's restrictive influence in Chapter 8, where we find compelling evidence that the TREC model and data has encouraged a publication culture that approves work that follows the standard methodology, even if that work contributes little technological progress.

2.3 Information retrieval as a universal service

The TREC conferences began in 1992; their immediate predecessor, the TIPSTER project, was instituted in 1990 (Harman, 1992a). Another event occurred between those two years that was to change the direction of information retrieval as a whole even more profoundly than TREC changed the direction of information retrieval evaluation. In August of 1991, the first web site was put online at CERN. Over the next decade, the web rapidly grew from a publishing tool for researchers to become the information nexus of the digital age. The research community met the arrival of the web unusually well resourced in the scale of experimental data available to it, thanks to TREC. This was timely indeed: the final verification and consolidation of long-standing techniques in the area came just in time for deployment in operational search engines, and technology and engineering experience developed in dealing with the large-scale TREC collections prepared the way for handling the large and rapidly growing data volume of the web. But research datasets were soon dwarfed in scale by the growth of the web; and evaluation methods, too, came to be stretched by the particular demands of web search. We now examine the impact that the emergence of the web has had on information retrieval, and the challenges the web poses to retrieval evaluation.

2.3.1 The web and TREC

The nature of the web makes running information retrieval systems over it challenging, but also vital. Traditional information repositories are generally curated, mostly homogeneous, either static or else accumulating new material without modifying or deleting the old, and consistent in their level of reliability and trustworthiness. On the web, pretty much anyone can publish pretty much anything. There is no consistent quality control. Every style, format, and content imaginable is represented, as are all major human languages and many formal ones besides. New content is added at a great rate,

and old content is continually modified or deleted. And, as the commercial and social importance of the web has grown, so too has the prevalence of attempts to consciously distort information and mislead users and search services. Running an information retrieval service in this environment is enormously challenging. However, without search engines, the web is for users unmanageable and largely inaccessible. Search engines have become a universal service, as important in most people's daily life as a traditional retrieval service was for dedicated information workers of the past.

The web has propelled the development of new retrieval techniques. Link analysis and the use of anchor text are the most obvious, but spam detection, page quality assessment, URL analysis, and a multitude of other features have become important. The web also, at last, drove the widespread operational adoption of some long-standing research technologies. Free text queries, statistical similarity calculations, and result ranking had been used in research systems since the 1960s, but at the end of the 1980s operational systems were still restricted to Boolean retrieval using complex queries composed by trained searchers (Salton, 1992), as indeed are some specialist retrieval services today (such as medical or legal literature retrieval systems). Complex Boolean retrieval was no longer workable on the web: search users lacked the training or patience even for the simplest Boolean operators, and answer sets were too large for humans to process without the assistance of relevance ranking. Search engines were finally forced to adopt technology demonstrated by researchers decades earlier.

The TREC effort adapted to the new demands of evaluating web search, but with only partial success. The advent of the web found TREC heading in the wrong direction. Topics in TREC 1 and TREC 2 were long and sophisticated, and included high-level concept, factor and definition fields, as illustrated in Figure 2.4 (Spärck Jones, 2000). We have seen that the Okapi project found themselves having to retrogress their interactive, short-query system, developed in an operational setting, to TREC's complex query, batch-mode environment (Robertson et al., 1992). By TREC 4 in 1995, the year that web search engines first appeared, TREC topics had been pared back to simple, one-line queries, as can been see in Figure 2.5. The shortening was performed to make the topics more similar to "what users normally submit to operational retrieval systems" (Harman, 1995, page 5). Spärck Jones (2000, page 53) states that the operational systems particularly in mind were web search engines, though this is not stated explicitly in that year's overview paper. In any case, the document collection continued to be newswire data. It was not until the Very Large Collection (VLC) track of TREC 7 in 1998 that a fully web-derived corpus was introduced (Hawking et al., 1998).

The VLC2 collection introduced at TREC 7 was ambitious in scale. The corpus was a crawl of a substantial portion of the web, made by the Internet Archive in 1997. The crawl contained 100GB of data and almost 18 million web pages, making it comparable in size to the indexed corpora of contemporary web search systems (Hawking and Craswell, 2005, page 205), It was even possible to compare the retrieval effectiveness of the submitted TREC systems on the static collection against that of commercial search engines on the live web, using the collection's qrels, with the research systems coming out ahead, albeit on queries atypical of web search (Hawking et al., 1998). The rapid growth in the web, however, soon left the corpus behind. By TREC 9 in 2000, the second year of the TREC Web track proper, the VLC2 collection was only a thirtieth of the size of document sets indexed by commercial search engines (Hawking, 2000). Though only three years old, it was already quite out of date as a snapshot of the web. In 2003, its final year as the largest TREC collection, it was one two-hundredth of web scale (Hawking and Craswell, 2005, page 206).

The challenge of performing public evaluation of web retrieval (as opposed to

```
<top>
<head> Tipster Topic Description

<num> Number:  053

<dom> Domain:  International Economics

<title> Topic:  Leveraged Buyouts

<desc> Description:
Document mentions a leveraged buyout valued at or above
200 million dollars.

<smry> Summary:
Document mentions a leveraged buyout valued at or above
200 million dollars.

<narr> Narrative:
A relevant document will cite a leveraged buyout (LBO)
valued at or above 200 million dollars. The LBO may be at
any stage, e.g., considered, proposed, pending, a fact.
The company (being) taken private must be identified.
The offer may be expressed in dollars a share.

<con> Concept(s):
1. leveraged buyout, LBO
2. take private, go private
3. management-led leveraged buyout

<fac> Factor(s):
<price> Price:  >= 200 million dollars
</fac>

<def> Definition(s):
Leveraged Buyout (LBO) - Takeover of a company using
borrowed funds, with the target company's assets serving
as security for the loans taken out by the acquiring
firm, which repays the loans out of the cash flow of the
acquired company or from the sale of the assets of the
acquired firm.
</top>
```

Figure 2.4: Sample topic from TREC 1. The topics have two labelling and eight descriptive fields. The descriptive fields include: concepts, which provide suggested search phrases; factors, which are logical statements that matching documents must satisfy; and definitions of specialist terminology.

```
<top>
<num> Number:  203

<desc> Description:
What is the economic impact of recycling tires?
</top>
```

Figure 2.5: Sample topic from TREC 4. By TREC 4, under the influence of operational retrieval practice, topics had been pared back to a single descriptive field, which held the query itself. Subsequent TRECs restored the "Narrative" field, introduced the "Title" field, and allowed the "Description" field to expand again. However, the classification fields of TREC 1 and 2 were not re-introduced, and participants increasingly derived their queries solely from the short, keyword-rich "Title" fields.

evaluation with the resources, and within the confines, of a commercial search engine lab) is not primarily that of collecting a corpus of web size. The effort involved in crawling web-scale corpora is still within the capacity of the research community, as is attested by the 426GB, 25 million document TREC Terabyte corpus released in 2004 (Clarke et al., 2004), or the billion-page ClueWeb collection, introduced to the TREC Web Track in 2009 (Clarke et al., 2009). The problem, rather, is with the dynamism of the web, its rate of growth and change, and the increasing richness and complexity of the information sources available in it. Static research collections quickly become dated and unrepresentative. More significantly, there are important sources of information on user search behaviour, such as query logs and click-through data, which are not freely available to public research. Given these challenges, and given the competition from well-resourced commercial search engine labs, it is an open question whether public information retrieval research, and collaborative efforts like TREC, can remain relevant to the whole retrieval task (at least as it is performed on the web), or whether they will be relegated to work on specific sub-problems.

2.3.2 Evaluating web-scale search

Although not all search is web search, the web is the most prominent contemporary search domain. Covering this domain poses many challenges to TREC-style evaluation. A basic one is to the completeness of relevance assessments. Test collections are designed to be reusable, and this re-usability depends on the set of relevance judgements being tolerably comprehensive and unbiased (Zobel, 1998). Since exhaustive relevance assessment is impractical, document selection is made by pooling system runs (Harman, 2005a). As corpora grow in size, however, the coverage of the assessment set becomes more questionable—particularly for the often under-specific short queries typical of web search. Pools become filled with easily matched documents, such as those rich in query keywords, and relevance sets created from such pools are biased against innovative methods that go beyond keyword matching (Buckley et al., 2007). As the expense of providing reusable assessments for each query is increasing, so too is the number of queries needed to cover the diversity of web search. The standard 50 queries of a TREC collection is arguably insufficient even for the homogeneous query and data sets of the Ad-Hoc newswire collections (Spärck Jones, 2000). It

is certainly inadequate for the heterogeneous query and documents types of the web.

While the web challenges the test collection model for TREC participants, it breaks it completely for operational systems. Evaluation is crucial to commercial web search engines, and they devote considerable resources to it (Hawking and Craswell, 2005; Huffman and Hochster, 2007). Before the web, operational information retrieval systems typically had a monopoly over the information source they were serving. Web search engines, however, all work over the same data sources, making competition intense, and quality of results crucial. Static, TREC-style collections are of limited value in this environment. The operational corpus is rapidly growing and changing, not just in content but in style. Keeping the testing corpus static creates a highly undesirable bias in favour of old documents; on the other hand, allowing the corpus to grow soon renders static relevance sets incomplete. The query stream is also continually changing, and the test environment must reflect it, or else leave retrieval algorithms stale. And in a setting of endlessly repeated system evaluation and tuning, the set of test queries must be large and always changing, to avoid over-fitting.

Attempting, TREC-style, to create comprehensive relevance sets through deep assessment is prohibitively expensive for query sets of operational scale, and in any case is ultimately pointless if query and document sets are undergoing continual change. And if evaluation reflects the tendency of web search users to only look a short way down result lists, deep relevance assessment becomes redundant. Instead, testers are necessarily driven to shallow evaluation over large query sets. Anecdotal evidence indicates that commercial web search systems indeed perform system evaluation over sets not of dozens of queries as in TREC, but of tens of thousands of queries; however, for each query, they have not thousands of relevance assessments as in TREC, but a dozen or fewer (Najork and Craswell, 2008). Such relevance sets are not naively reusable in the original TREC manner, but nor is it economically feasible to throw them away after each experiment. Instead, methods need to be found that allow for the reuse of shallow relevance assessment in a highly dynamic environment of changing documents, queries, and systems. We propose one such method in Chapter 6.

The growth in data scale, and improvements in technology, have provoked interest in applying machine learning and data mining technology to the web, interest which has extended to system evaluation and enhancement as well. For instance, one of the most popular areas of contemporary research interest is "learning to rank" (Trotman, 2005). Here, rather than manually developing and tuning retrieval algorithms, search systems use automated methods to select the mix of features and feature weightings used to predict document relevance and thus estimate the optimal document ranking. The nature of the web is, again, a major driver behind this interest: it presents a far greater range of possible features than traditional collections, and its dynamic nature requires the constant readjustment and redevelopment of retrieval methods. While learning to rank still requires relevance assessments to train from, one can also anticipate the growth of a range of unsupervised methods, in which retrieval effectiveness is not directly measured, but rather the degree of similarity between different rankings is of interest. For instance, an operational search provider might be interested in tracking the speed and nature of a rival's changes to their ranking algorithm. Such ranking comparisons have particular features (top-weightedness, disjointness) that make traditional rank similarity metrics unsuitable. And these features can be found not just in the document rankings produced by search engines, but several other fields besides. We propose a metric suitable for calculating similarity between rankings of this sort in Chapter 7.

As serious as the issues of scale in retrieval evaluation, are those of the adequacy of the test collection model itself. The standard retrieval evaluation methodology is

becoming increasingly unsatisfactory as a model of information access on the web. Users search interactively, refining and adapting queries in response to intermediate results and their developing understanding of their information need; but the standard evaluation method only models batch retrieval. Users come to search engines with a great variety of background states of knowledge, some of which could be captured by search engines; but the standard model reduces the user to a query and a set of independent document relevance assessments. There are many choices about how to present search results, and indeed there are possible methods of information provision quite different from the query–results model; but the standard test collection evaluation model reduces each returned document to its relevance value, not even accounting for basic presentation features such as document summaries. Many of the problems with retrieval evaluation methods were apparent before both the web and TREC; the failure of automated batch evaluation to capture the interactive user experience has been widely appreciated since the late 1980s at least. However, the added possibilities of web retrieval bring the inadequacies of system-centric, batch-mode evaluation into even sharper focus. Meanwhile, the TREC effort has served to reinforce the rigidity of the traditional evaluation model, by providing large-scale, high-quality experimental datasets that are for the most part only usable within the traditional model—and at the same time setting the bar higher for those wishing to undertake research outside this model and unable to access voluminous experimental data. Some of the effects of this influence are examined in Chapter 8.

These are significant challenges for the future direction of evaluation in information retrieval. Nevertheless, the resilience of the system-centric evaluation model should not be ignored. Its essential elements appear to have been adopted by operational systems for their internal testing: creation or sampling of a set of test queries, assessment of document–query relevance (or at least utility), marking up of document rankings for relevance to test queries, and scoring of the runs accordingly. The reasons for retaining this evaluation model, for all its artificiality, are clear: it is robust, repeatable, readily automated, and at least partially reusable. Other, more user-centric, and probably more expensive evaluation methods are required in addition to the system-centric model, but they are unlikely to replace it, at least for the time being. As such, it is important that the system-centric model continues to develop its reliability, sensitivity, efficiency, and, within the limitations of its basic model, its flexibility. It is to the furthering of these qualities that the main contributions of this thesis are aimed.

2.3.3 Extending retrieval evaluation's legacy

Information retrieval has a fifty-year history as an independent discipline. From its beginning in the Cranfield experiments and through its elaboration by the long-running SMART project, the discipline has been marked by a strong empirical tradition. The experimental methodology developed at Cranfield has proven resilient and enduring: system-centric evaluation using a test collection of documents, queries, and relevance assessments. It was with the TREC effort, begun in 1992, that this methodology finally came into its own; at last, test collections of sufficient scale were formed to consolidate the discoveries of the previous decades and provide a firm basis for future work. Not the least of the outcomes of TREC has been a keen attention to assessing and improving the standard experimental methodology; and it is in this stream of work that the current thesis can be located.

Meanwhile, the emergence of the web has taken information retrieval from being the modest domain of research librarians and proprietary databases, to a universal ser-

vice of everyday use; and, of course, a hugely lucrative business besides. The prominence of web search has made effective evaluation all the more important, at the same time that the peculiarities of web search have made it all the more difficult. Part of the solution to this challenge lies in developing new evaluation methods, which is a task we must leave to others; part, in extending existing methods to meet the new demands, which is the task we undertake here. This chapter has reviewed the historical background of our task; the next lays out its theoretical foundations.

Chapter 3

Technical background

The previous chapter described the historical development of retrieval evaluation by test collection, from the Cranfield experiments to the TREC effort. We now examine the assumptions, applications, and challenges of this mode of evaluation. Test collection evaluation is system-centric, replacing the user with an abstract model of the retrieval process. This model, and its embodiment in the test collection method, is presented in Section 3.1. Retrieval effectiveness is quantified using an evaluation metric; common metrics and metric components are described in Section 3.2. Generalizing a system's effectiveness beyond a particular collection requires the use of statistical tools, examined in Section 3.3. The practical challenges involved in test collection formation and the interpretation of results are discussed in Section 3.4. Finally, the TREC test collections and runsets, which form our main data set, are introduced in Section 3.5.

3.1 Mode, model, method

Test collections enable a system-centric mode of evaluation, based upon a simple model of information retrieval. Under this model, a retrieval system is effective if it returns documents that are relevant to the user's information need. The test collection supports the automated evaluation of relevance-based effectiveness, through its three components: documents; statements of information needs called topics; and judgments as to which documents are relevant to which topics. The mode, model, and method of test collection evaluation is discussed below.

3.1.1 User and system studies

Evaluation experiments can be performed in any of several modes. For instance, experiments can be *observational*, watching and interpreting search behaviour in person or through query logs; or they can be *operational*, modifying a working system and observing how usage changes. But it is two other experimental modes, and the distinction between them, that are most important to understanding the test collection method: the *user-* and *system-centric* modes of experiment.

The success of a retrieval system is defined by the user's satisfaction or utility. Therefore, it seems most natural that retrieval experiments directly involve users, both in searching the system, and in assessing its results. Bringing users into a laboratory setting allows a range of manipulations and observations to be made, from role-playing

retrieval scenarios to monitoring user attention with eyetracking devices. Evaluation can by questionnaire, or by setting the user a task and seeing how effectively they perform it. Such studies offer a rich range of experimental opportunities. The problem is the cost: user studies are expensive and time consuming. Also, the investment in one study cannot be reused in subsequent experiments; each time a change in the retrieval method is made, even if it is just the tuning of a parameter, the user study must be repeated. And the experimental results cannot be precisely replicated: redoing the same study under the same conditions but with different users, or even the same users at a different time, can potentially lead to different results (Voorhees, 2008).

Re-usability of assessment effort, and replicability of results, are the prime motivations for the *system-centric* mode of evaluation, embodied in the test collection. System-centric experiments require no direct user involvement, allowing them to be automated and precisely reproduced. Creating the test collection is no less expensive than performing an equivalent user experiment; but, once created, the collection can be reused unchanged, making further experiments much less costly. The benefits of automation, replicability, and reuse, have made system-centric evaluation the predominant experimental mode—at the expense, some would say, of the realism and richness of user-centric evaluation (Robertson, 2008a). The challenge in implementing the system-centric mode is that the assessment of retrieval effectiveness still relies on human judgment, so some way must be found to capture this judgment in a reusable way. The first step is to develop a model of the retrieval process, from which the role of the user can be abstracted.

3.1.2 Modelling information retrieval

The automated evaluation of a user activity requires a model of user behaviour and perception, one that defines what effective system performance is. In the test collection model, retrieval begins with a user's *information need.* The need is expressed to the system as a *query*, to which the system responds with a ranked list of documents or *ranking* (refer back to Figure 1.2). The user examines some prefix of the ranking, looking for documents that are *relevant* to their need. This model leads to the following proposition, a simplified version of the *probability ranking principle* (Maron and Kuhns, 1960; Robertson, 1977):

Proposition 3.1 *The function of an information retrieval system is to take a user's query, and return a list of documents that are relevant to that query, ranked by order of probable relevance.*

Proposition 3.1 leads to the following corollary:

Corollary 3.2 *A retrieval system's effectiveness should be evaluated based on the number, proportion, and ranking of relevant documents it returns.*

Proposition 3.1 and Corollary 3.2 define the *topical relevance* interpretation of information retrieval. (Corollary 3.2 could be extended to consider each document's *degree* of relevance; in this thesis, though, as in much IR evaluation, documents will be assumed either wholly relevant or wholly irrelevant.)

Several simplifications are made in the model described above. It assumes that *relevance* (Maron and Kuhns, 1960; Mizzaro, 1997) can be measured as a topical match between information need and document. The model is one of *batch retrieval*: there is a single query, with a single result, and no possibility of query refinement. It is also a

model solely of *document retrieval*: issues of information extraction and presentation are not considered; nor is interface design or speed of response incorporated. Finally, the model assumes *independence of document utility*: a document's usefulness to the user is independent of which other documents appear in the ranking. Simplistic though they are, these assumptions are made because they greatly ease the evaluation of effectiveness. Once a document has been assessed for relevance to a query, that assessment can be reused, without having to consider interactions with other documents or previous searches. The model does not capture the full complexity of the retrieval process; but it can be embodied in an efficient, repeatable experimental methodology.

3.1.3 The test collection methodology

The topical relevance model identifies three elements of the retrieval process: the user's information need; the set of documents from which the system attempts to satisfy the need; and the assessed relevance of each document in the collection to that need. These three elements are represented by the three component of the test collection: a set of *topics*, each describing a different information need; a *corpus* of documents; and assessments of which documents are relevant to which topics, for which the term *qrels* has been coined. Relevance assessments may be graded, allowing a document to be, say, partially or highly relevant to a topic, but (as mentioned above) binary relevance will be assumed in this thesis.

A retrieval system is evaluated against a collection as follows. Each topic is formulated as a query, generally by extracting topic fields, and the query is submitted to the system. The system matches the query against the documents in the collection, typically with the aid of an index it has built, using whatever retrieval algorithm it implements, and returns a list of documents, ranked by decreasing estimated likelihood (or degree) of relevance to the query. The qrels are consulted to determine which documents in the ranking are relevant to the topic, and the ranking is converted into an ordered list of relevance values or a *relevance vector*. The process in illustrated in Figure 3.1. An evaluation metric is then used to score the relevance vector, as described in Section 3.2, and the scores that the system achieves for each topic are aggregated, normally as the arithmetic mean, to derive the system's *effectiveness score* for the collection. The combination of test collection with evaluation metric will be referred to here as a *test environment*.

The methodology described above reduces the user to a query and a set of relevance assessments, and the search result to a vector of binary values, to be further summarized into a single score. The benefit of this simplification is that queries and assessments can be captured in advance, and used without change or variability in repeated experiments on different systems, with different retrieval parameters, at different times, and by different groups. The test collection therefore provides an automated toolset replicating the essentially manual process of information search and result assessment.

3.2 Evaluation metrics

The raw output of a test collection evaluation is an ordered binary vector:

$$\overrightarrow{\mathcal{R}} = \langle r_1, r_2, \ldots \rangle, r_i \in \{0, 1\}$$

for each topic in the collection, representing the relevances of the document ranking returned by the system for that topic. In order to quantify the system's performance,

Topic	(Unused)	Document id	Rank	Similarity score	System id
405	Q0	FT943-10128	1	121.13205	ric8dnx
405	Q0	LA052890-0021	2	119.91743	ric8dnx
405	Q0	LA092489-0134	3	117.35849	ric8dnx
405	Q0	FT942-5468	4	110.26174	ric8dnx
405	Q0	FT944-864	5	106.15862	ric8dnx
405	Q0	FT922-11472	6	103.69264	ric8dnx
405	Q0	LA010889-0109	7	103.28536	ric8dnx
405	Q0	LA022689-0112	8	99.37935	ric8dnx
405	Q0	LA090889-0077	9	96.91350	ric8dnx
405	Q0	FT924-286	10	93.05222	ric8dnx

(a) Ranking

Topic	(Unused)	Document id	Relevance
405	0	FT922-11472	1
405	0	FT924-286	0
405	0	FT942-5468	0
405	0	FT943-10128	1
405	0	FT944-864	0
405	0	LA010889-0109	0
405	0	LA022689-0112	1
405	0	LA052890-0021	1
405	0	LA090889-0077	0
405	0	LA092489-0134	0

(b) Qrels

$\langle 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0 \rangle$

(c) Relevance vector

Figure 3.1: TREC input file formats, and intermediate evaluation output: (a) document ranking; (b) relevance judgments or qrels; and (c) resultant relevance vector. The first ten documents returned by the system `ric8dnx` for Topic 405 in the TREC 8 AdHoc collection are shown, along with the relevance assessments for the same ten documents.

the relevance vector is converted to a numeric score, using an evaluation metric, which can be defined as follows:

Definition 3.3 *An evaluation metric is a function that takes an ordered vector of relevance values, and returns a single numeric score, summarizing those values.*

This is a minimal definition. Many metrics also take as input the full set of relevant documents, or at least the size of this set; a few metrics return a range or set of values; and some metrics support graded relevance assessments. We now proceed to examine how different metrics implement Definition 3.3.

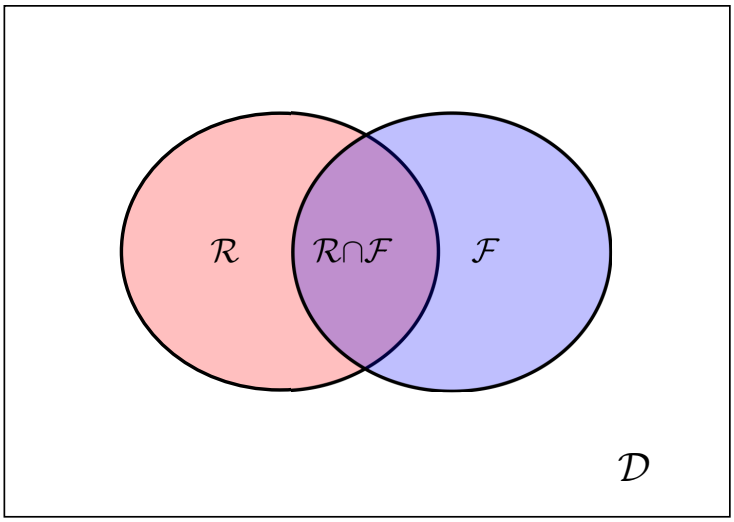


Figure 3.2: Overlapping division of document space $\mathcal{D}$ into documents that are retrieved by the system for an information need ($\mathcal{F}$), and documents that are relevant to that information need ($\mathcal{R}$).

3.2.1 Precision and recall

Two fundamental concepts in retrieval evaluation are *precision* and *recall*. They are defined as follows:

Definition 3.4 *Precision is the proportion of returned documents that are relevant.*

Definition 3.5 *Recall is the proportion of relevant documents that are returned.*

More formally, let $\mathcal{R}$ be the set of relevant documents, and $\mathcal{F}$ be the set of returned (fetched) documents. Then

$$\text{Precision} = \frac{|\mathcal{R} \cap \mathcal{F}|}{|\mathcal{F}|} \tag{3.1}$$

$$\text{Recall} = \frac{|\mathcal{R} \cap \mathcal{F}|}{|\mathcal{R}|} \tag{3.2}$$

The relationship is illustrated in Figure 3.2. Stated informally, precision measures the accuracy of a result, recall its completeness. The two are complementary, and there is a natural tension between them. Full recall is trivially achieved by returning every document in the collection, but then precision will be low; conversely, returning only the documents most likely to be relevant boosts precision, but harms recall. In general, one can trade off precision for recall and vice versa, but can only increase both by improving the effectiveness of the retrieval system. The inverse relationship between precision and recall was one of the important findings of the Cranfield experiments, and, though initially questioned, is now generally accepted (Spärck Jones, 1981b).

Where the search result is an unordered set of documents, as in Boolean retrieval, the interpretation of precision and recall is straightforward. The meaning of these measures is unclear, however, with ranked retrieval, since there is no single set of retrieved documents, but rather a ranking over the document set. The simplest solution is to

treat all documents above some cutoff depth k as an unordered result set, and calculate *precision at cutoff* k (P@k):

$$\text{P@}k(r) = \frac{1}{k}\sum_{i=1}^{k} r_i \tag{3.3}$$

Recall at cutoff k can similarly be defined, provided the number of relevant documents $R = |\mathcal{R}|$ is known:

$$\text{R@}k(r) = \frac{1}{R}\sum_{i=1}^{k} r_i \ . \tag{3.4}$$

Comparing Equations 3.3 and 3.4 shows that R@$k(r)$ = P@$k(r) \times k/R$; and, while P@k is widely used (for instance as P@10), R@k is rarely employed.

Precision at cutoff k is simple to calculate, and the meaning of a given P@k score is easy to interpret. The metric, though, is open to several objections, each of which suggests an alternative form of metric. The first objection has historically been that P@k does not measure recall. A ranking that returns four documents in the top ten receives a precision at ten score of 0.4, regardless of the number of relevant documents it failed to return. A second objection to P@k is that scores are not adjusted for topic difficulty. A P@k score of 1, for instance, cannot be achieved when $R < k$; conversely, if $R \gg k$, P@k scores close to 1 may become overly common. One response to this problem is score normalization, described below. And a final objection to P@k is that it is not sensitive to the rank at which relevant documents are returned, apart from the cutoff k itself, whereas more emphasis should be placed on higher rankings, suggesting that metrics should be rank-weighted.

There is a natural relationship between metrics that incorporate recall and those that adjust for topic difficulty: the larger the number of relevant documents R, the easier in general (though not always) that topic is. It will also be seen that the incorporation of recall often results in metric top-weightedness as well. Nevertheless, for recall-based metrics, the incorporation of recall is the central motivation in itself, while top-weightedness and an adjustment for topic difficulty are side benefits.

3.2.2 Recall-based metrics

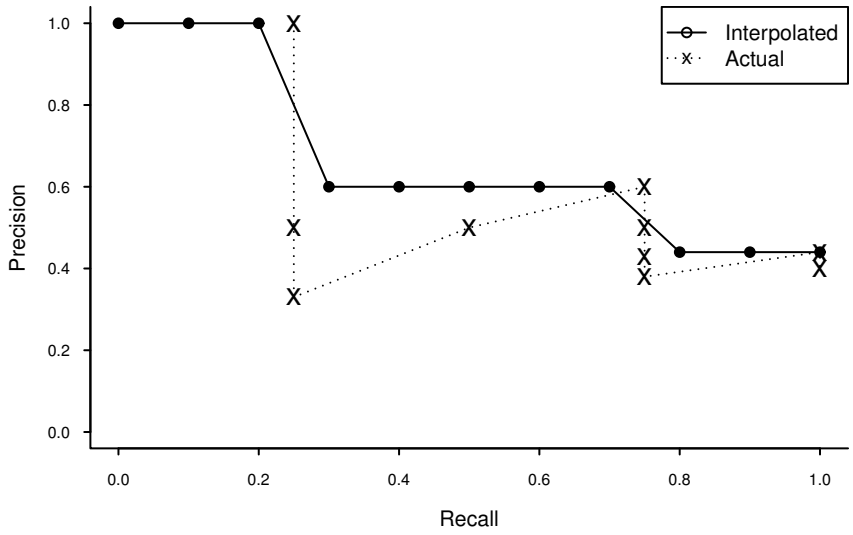
The traditional way to combine recall and precision as measures of ranked retrieval is through a recall–precision graph, which measures precision at different levels of recall (Tague, 1981). Some form of interpolation is generally applied; for instance, setting precision at a given proportional recall level to the highest subsequent precision (Buckley and Voorhees, 2005). Interpolation enforces what is otherwise the strong tendency for precision to stay the same or fall as recall increases. Figure 3.3 gives a worked example of an interpolated recall–precision curve, and illustrates why interpolation is necessary to remove the erratic and jagged nature of the actual recall and precision values at each rank. Such interpolated recall–precision curves, while rich in information on a single topic and system, are unwieldy to compare or even to cite; early papers describing third-party recall–precision curves often descend to verbal descriptions of a graph the author can see but the reader cannot (see, for instance, Spärck Jones (1981a, pages 234–235)). In addition, recall–precision curves assume that the ranking includes all, or at least the great majority of, relevant documents; with large contemporary collections, this assumption is no longer realistic (Buckley and Voorhees, 2005).

	Rank									
	1	2	3	4	5	6	7	8	9	10
r_i	1	0	0	1	1	0	0	0	1	0
R@i	0.25	0.25	0.25	0.5	0.75	0.75	0.75	0.75	1	1
P@i	**1**	0.5	0.33	0.5	**0.6**	0.5	0.43	0.38	**0.44**	0.4

(a) Actual recall-precision

Recall	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Precision	1	1	1	0.6	0.6	0.6	0.6	0.6	0.44	0.44	0.44

(b) Eleven-point interpolated recall-precision



(c) Recall–precision curve

Figure 3.3: Recall–precision calculations: (a) actual, and (b) eleven-point interpolated, recall-precision for a sample ranking, and (c) the corresponding actual and interpolated recall–precision curves. The bolded precision values in (a) are those that are interpolated as the precision values in (b). The total number of relevant documents R is assumed to be 4.

Various ways of combining recall and precision into a single score have been proposed. The most widely adopted of these is *average precision* or AP, calculated by averaging the precision for every position in the ranking at which a relevant document is returned; relevant documents not returned in the ranking by the cutoff depth are assigned a precision of 0.[1] The metric gives an approximation of the area under the

[1]The metric was initially called *mean average precision* or MAP, to distinguish it from the various ways of "averaging precision" presented in the earlier literature. Average precision (AP) is, however, a more logical name, and will be used throughout this thesis; if referring to the mean over a set of topics, we may speak of "mean AP" (as of "mean P@10"), but we will not speak of "MAP".

	Rank										Sum
	1	2	3	4	5	6	7	8	9	10	
r_i	1	0	0	1	1	0	0	0	1	0	
P@i	1		+	0.5	0.6		+		0.44	=	2.54
										$/R$ =	6
										AP@10 =	0.42

Figure 3.4: Example average precision calculation. Evaluation is performed to depth ten. The number of relevant documents R is assumed to be six, of which the system has returned four within the ten documents that are in the ranking.

recall–precision curve. More formally, let k be the cutoff depth; then:

$$\text{AP@}k(r) = \frac{1}{R} \sum_{d=1}^{k} r_d \frac{\sum_{i=1}^{d} r_i}{d} \tag{3.5}$$

where the inner fraction calculates precision at each depth, and r_d acts as an indicator variable, only including precisions at depths that hold a relevant document. Figure 3.4 shows a sample calculation. Note that we have been careful in Equation 3.5 to include the cutoff depth k in the notation for the metric. Frequently, the cutoff depth is not explicitly specified, especially when it is relatively deep. Practice at TREC is to evaluate to depth $1{,}000$; when we refer in later chapters to AP without specifying a cutoff depth, the depth of $1{,}000$ is assumed.

Average precision gives greater weight to higher-ranked relevant documents, since they contribute to the precision at lower-ranked relevant documents, but not vice versa. The exact weight, however, is determined not for a single document, but for a document pair, and is not fixed for a given rank, but depends on R. Specifically, from Equation 3.4, it can be seen that for a pair r_d, r_c of relevant documents, with $c < d$, the weight is $1/(R \cdot d)$. These entanglements make analysis and estimation of the metric quite complex (Aslam et al., 2006; Carterette et al., 2006). Average precision adjusts for topic difficulty, or at least for R; a score of 1 is always and only achievable by returning the set $\mathcal{R}$, in any order, at the head of the ranking. But the incorporation of R requires that the number of relevant document be known or estimated, which is challenging for large collections (see Section 3.4). Some have also criticized recall-based metrics like average precision as not reflecting user experience, arguing that a user's satisfaction with the results they see is not affected by how many relevant documents have (unknown to the the user) not been returned (Cooper, 1973; Moffat and Zobel, 2008). The very concept of recall has also been questioned (Zobel et al., 2009). Nevertheless, average precision has been the most widely used evaluation metric over the past two decades, and therefore is employed in many of the experiments here.

One other recall-based metric deserves description, and that is precision at R documents (RPrec). As the name suggests, RPrec is a precision at cutoff metric, but instead of having a fixed cutoff, the depth is set to R, and hences varies from topic to topic. More formally:

$$\text{RPrec}(r) = \frac{1}{R} \sum_{i=1}^{R} r_i \tag{3.6}$$

	Rank									
	1	2	3	4	5	6	7	8	9	10
r_i	1	0	0	1	1	0	0	0	1	0
$w(i)$	1.0	1.0	0.63	0.5	0.43	0.39	0.36	0.33	0.32	0.30
$r_i \cdot w(i)$	1.0	0	0	0.5	0.43	0	0	0	0.32	0

$$\text{DCG@10}(r) = 1.0 + 0.5 + 0.43 + 0.32 = 2.25$$

Figure 3.5: Example discounted cumulative gain calculation. Evaluation is performed to depth ten. The b parameter is set to 2.

As with average precision, a precision at R score of 1 is achievable and only achievable by returning the set $\mathcal{R}$ at the head of the ranking. Precision at R documents is not rank-weighted, though. Despite this, and despite the metric's simplicity, the correlation between RPrec and average precision is remarkably close (Buckley and Voorhees, 2005; Carterette, 2009). A geometric explanation for this relationship is given by Aslam and Yilmaz (2005).

3.2.3 Rank-weighted metrics

Another objection to P@k, beside its ignoring recall, is that it is not sensitive to the rank at which relevant documents are returned. Retrieval systems aim to return documents by decreasing probability of relevance (Robertson, 1977), and users examine results in rank order (Joachims et al., 2005); therefore, the higher the rank of a relevant document, the greater should be the system's score. Average precision achieves rank-weightedness through the way it incorporates recall. It is, however, possible, and perhaps preferable, to weight ranks explicitly, and orthogonally to other metric features.

A family of rank-weighted metrics can be defined of the form:

$$\text{RWM@}k(r) = \sum_{i=1}^{k} r_i \cdot w(i) \tag{3.7}$$

where $w(i)$ is the weight assigned to rank i. Here, r_i, the relevance of the document at rank i, can take on any range of values, which was one of the major motivations behind the first such rank-weighted metric, DCG (described below); we will, however, only be using binary relevance values in this thesis. The ease with which graded relevance values are supported in rank-weighted metrics is a result of making rank weighting orthogonal; incorporating graded relevance into recall-based metrics like average precision is far more complicated (De Beer and Moens, 2006). Different metrics are realized by choosing different weighting functions w. Precision at cutoff k can itself be viewed as a member of this family, with $w(i) = 1/k$ for $i \leq k$, and 0 otherwise.

The first explicitly rank-weighted metric to be proposed was *discounted cumulative gain* (DCG) (Järvelin and Kekäläinen, 2000, 2002). The metric sets out to calculate the value of a ranked list to the user. Each relevant document represents a (possibly graded) gain (G) in value. The further down the list a document occurs, the more its value to the user is discounted (D), since the user is less likely to look at it. Finally, the total

	Rank									
	1	2	3	4	5	6	7	8	9	10
r_i	1	0	0	1	1	0	0	0	1	0
$w(i) \cdot 10^3$	200	160	128	102	082	066	052	042	034	027
$r_i \cdot w(i) \cdot 10^3$	200	0	0	102	082	0	0	0	0	034

$$\text{RBP}(r) = 0.200 + 0.102 + 0.082 + 0.034 = 0.418$$

Figure 3.6: Example rank-biased precision calculation, with the persistence parameter p set to 0.8. Evaluation is performed to depth ten. The RBP score at this depth gives a lower bound on the RBP score achievable at greater, including infinite, depths.

value of the ranking to the user is cumulated (C) from the discounted individual gains. These elements correspond precisely to the elements of the rank-weighted metric in Equation 3.7: the gain is r_i, the cumulation is the sum $\sum$, and the discount is the weighting function $w(i)$. The weighting function proposed by Järvelin and Kekäläinen, tuned by the parameter $b > 1$, is:

$$w_{\text{DCG}}(i) = \begin{cases} 1/\log_b(i) & \text{if } i > b\,; \\ 1 & \text{otherwise} \end{cases} \tag{3.8}$$

Järvelin and Kekäläinen suggest $b = 2$ as the default parameter choice. An example calculation of DCG using Equation 3.8, with $b = 2$, is given in Figure 3.5. As with AP, the cutoff depth of DCG is frequently not reported if it is particularly deep; again, the TREC standard is evaluation to depth $1{,}000$, and that will be followed in this thesis.

Under the original DCG weighting scheme given in Figure 3.5, all ranks up to and including rank b receive the same weight; in particular, for $b = 2$, a ranking that starts $\langle$`0 1 ...`$\rangle$ is scored as highly as a ranking that starts $\langle$`1 0 ...`$\rangle$. Moreover, the equality $(1/\log_b c)/(1/\log_b d) = \log_c d$ means that, aside from the flatness of initial weights, the steepness of the decline in weights is the same, whatever value of b is chosen. The minor and counter-intuitive influence of the choice of b in Equation 3.8, and the complexity that the equation adds both in expression and analysis (Sakai, 2007a), have led to an alternative formulation of the DCG weighting scheme:

$$w_{\text{MSDCG}}(i) = \frac{1}{\log_2(i+1)}\,, \tag{3.9}$$

sometimes informally referred to as "Microsoft DCG", as it seems to have originated from MS Research (Burges et al., 2005). Although the MS-DCG weighting scheme has much to recommend it in simplicity and intuition, we will use the original DCG weighting in this thesis, with $b = 2$, as it still appears to be the more widely used variant in evaluation research.

Another rank-weighted metric is rank-biased precision (RBP) (Moffat and Zobel, 2008). Like DCG, RBP is based on a simple user model, in this case that of how users peruse rankings. The central concept is the user's *persistence*, modelled as a parameter p, which is the probability that a user, having reached a given rank in the results, will proceed to the next rank. The probability that a user will reach rank i is then p^{i-1}; the user is assumed always to look at the first rank. These probabilities form a geometric

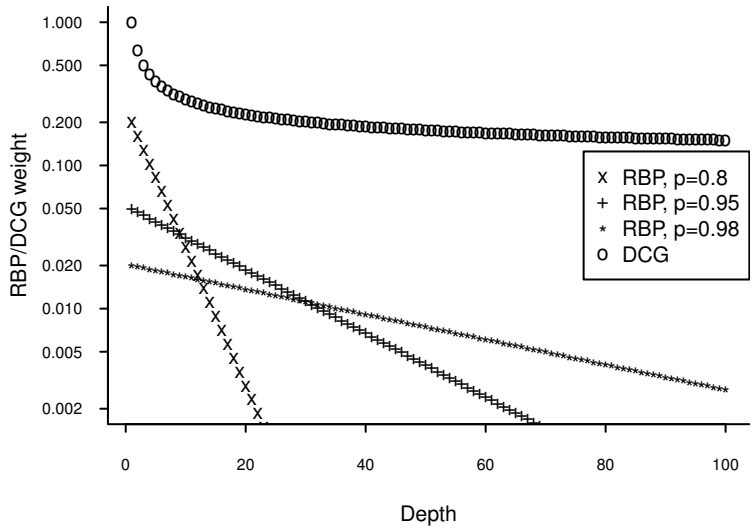


Figure 3.7: Rank weightings, of RBP with different p, and of DCG. Each point gives the contribution that a document at that rank makes to the run's score if that document is (binary) relevant. Note the logarithmic y axis.

sequence, summing to $1/(1-p)$; multiplying by $(1-p)$ makes the weights sum to 1. Thus, the weighting function for RBP is:

$$w_{\mathrm{RBP}}(i;p) = (1-p)p^{(i-1)} \tag{3.10}$$

The parameter p is used to control the degree of top-weightedness of the metric; the lower p is, the less persistent the user, and hence the greater the top-weighting. An example working of RBP is given in Figure 3.6.

While RBP is a member of the family of rank-weighted metrics, its weighting function has the important feature that it is convergent, summing to 1, whereas DCG's logarithmic weights are divergent, their sum going to infinity as the ranking is extended indefinitely. Thus, RBP scores are bounded in the range $[0, 1]$ (provided relevance values are also in this range), while DCG scores are unbounded. For an infinite ranking under DCG, the weight of the tail always dominates the weight of the head, as suggested by Figure 3.7. Therefore, DCG requires an explicit evaluation depth cutoff, whereas RBP naturally converges, even if evaluation is carried out to an indefinite depth.

A particularly useful effect of RBP's convergent weights is that a partial evaluation sets bounds on a full one. Each rank has a fixed weight, so the residual uncertainty of an evaluation is the sum of the rank weights of unassessed documents in the ranking. The base RBP score is the score achieved on the assessed documents; the maximum is the base plus the residual, representing the score that the system would achieve if all unassessed documents turned out to be relevant. As more documents are assessed, the residual is monotonically decreasing, the base monotonically non-decreasing, and the maximum monotonically non-increasing. These features are especially helpful when, as is generally the case, the qrels are incomplete, with many documents not assessed for relevance, a situation discussed further in Section 3.4.

One other metric, rank-weighted but not a member of the RWM family, deserves mention here, and that is *reciprocal rank* (RR). The reciprocal rank score for a ranking

is the inverse of the rank of the most highly-ranked relevant document. More formally:

$$\text{RR@}k(r) = \begin{cases} 1/(\min\{i : r_i = 1\}) & \text{if } \min\{i : r_i = 1\} \leq k \\ 0 & \text{otherwise} \end{cases} \tag{3.11}$$

The user model behind reciprocal rank is one in which the user scans the ranking from the top, and stops when they find a relevant document. A probabilistic version of the metric has been shown to correlate with user click behaviour (Chapelle et al., 2009). Reciprocal rank is often used as a metric in retrieval tasks in which there is only one relevant document, or at most a handful which are equivalent, such as the named page and home page tasks of the TREC Web track (Hawking and Craswell, 2001).

3.2.4 Metric normalization

Evaluation metrics typically give scores in the range $[0, 1]$, and it is at least aesthetically desirable that newly-proposed metrics also follow this convention. An example of a metric which does not is DCG. How scores should be distributed in the $[0, 1]$ range depends on how the metric is understood. If the metric attempts to directly quantify some independent property of the result, such as user satisfaction, then scores are tied to that property; users may find a result unsatisfactory even if no better result is achievable for that topic. If, however, the metric is designed to measure system performance, then it desirable that a full score of 1 be achievable for every topic. Precision-at-cutoff-k, for instance, fails this requirement for topics having less than k relevant documents. It is also desirable, though difficult to achieve, that metric scores are independent of topic difficulty, because the score a system achieves on a topic should measure system quality, not topic difficulty (see Moffat (2010) for a discussion of a number of desirable—but not all simultaneously achievable—metric properties).

The most direct way to bound a metric in the range $[0, 1]$ is to divide a system's score by the maximum score achievable on that topic. This technique will be referred to here as *normalization*. The maximum score for a topic is found by forming an ideal ranking, namely, one that places the most highly-relevant documents first, followed by the next most-relevant documents, and so forth, with the irrelevant documents last. For binary relevance, there are only two relevance classes to consider. The score of the ideal ranking is calculated, and observed scores on the topic are normalized by dividing them by the ideal, maximum score. Discounted cumulative gain, for instance, can be normalized to produce *normalized discounted cumulative gain* or nDCG (Järvelin and Kekäläinen, 2002); and indeed this is the more commonly used form of the metric. If binary relevance is assumed, with R relevant documents for a query, then the nDCG formula is:

$$\text{nDCG@}k(r) = \sum_{i=1}^{k} r_i \cdot w_{\text{DCG}}(i) \Big/ \sum_{i=1}^{\min\{k,R\}} w_{\text{DCG}}(i) \tag{3.12}$$

where $w_{\text{DCG}}(i)$ is the DCG rank weight, given in Equation 3.8; MS-DCG from Equation 3.9 (or indeed another weighting scheme) can be substituted instead. Note that the numerator in Equation 3.12 is the DCG score of the observed ranking, while the denominator is the DCG score of the ideal ranking. Average precision can also be defined as a normalized metric: normalized *sum of precisions* (SP) (Yilmaz and Aslam, 2006). Sum of precisions is defined as:

$$\text{SP@}k(r) = \sum_{d=1}^{k} r_d \frac{\sum_{i=1}^{d} r_i}{d} \,. \tag{3.13}$$

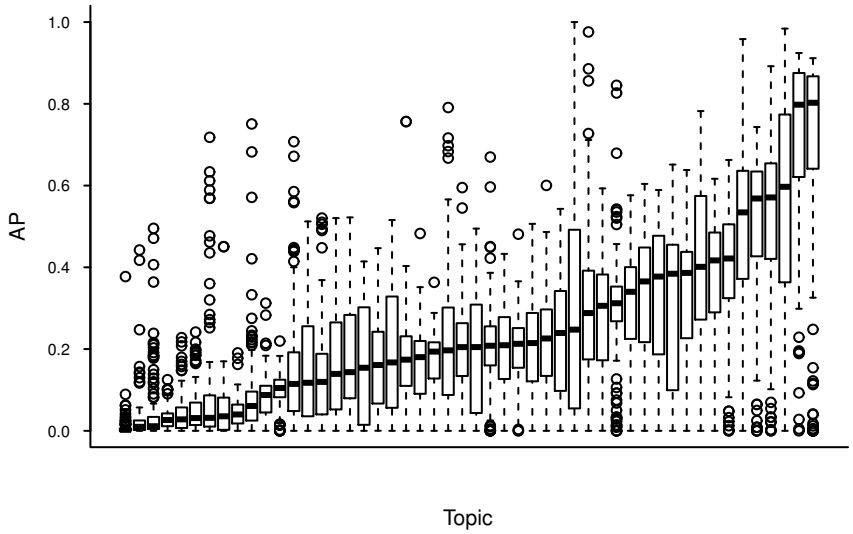


Figure 3.8: Per-topic distributions of AP scores for the systems participating in the TREC 8 Ad-Hoc Track, on all 50 topics in the track's collection. Each column represents a topic, ordered by the median AP score. The solid line is the median AP score achieved by the set of systems on that topic; box edges gives quartiles; whiskers extend to the outermost point within 1.5 times the interquartile range; dots show outliers.

An ideal ranking places all R relevant documents at the top, achieving an SP score of R, assuming that $R \leq k$. Therefore, normalized sum of precisions or nSP is $\text{SP}(r)/R$; and comparing Equations 3.13 and 3.5 shows that $\text{nSP@}k(r) = \text{AP@}k(r)$. Thus, recall-based metrics incorporate a form of normalization; but making normalization a separate, orthogonal component results in greater flexibility in metric construction.

A perfect score is always achievable under a normalized metric, by returning an ideal ranking—although the correct treatment of topics that have no relevant documents is ambiguous (Moffat, 2010). Normalization is also an approach to making scores independent of topic difficulty, in so far as difficulty is measured by the number of relevant documents, R. On the other hand, normalization requires a knowledge or estimate of R, as with recall-based metrics. Also, topic difficulty depends on more than the number of relevant documents alone, as can be seen from Figure 3.8: even though average precision is a normalized metric, the distribution of AP scores differs greatly between topics. We return to making scores independent of topic difficulty in Chapter 4, where we propose a more reliable method; namely, to *standardize* scores by the results that reference systems achieve on each topic.

3.2.5 Metric meta-evaluation

Several evaluation metrics have been discussed above; many more are described in the literature. How are we to choose between them? What criteria should be used to evaluate evaluation metrics?

A natural first question is, does choice of metric matter? Do different metrics give noticeably different results? Table 3.1 compares system rankings under different metrics on one TREC experiment. The measure is Kendall's τ, which is described in Sec-

	RBP.95	nDCG	P@10	RR
AP	0.81	0.88	0.74	0.52
RBP.95		0.79	0.87	0.61
nDCG			0.75	0.54
P@10				0.65

Table 3.1: Correlation between different metrics. The Kendall's τ between TREC 8 AdHoc track system rankings under different metrics is shown. The 27 systems which came in the bottom quartile for every metric are excluded, leaving 102 systems in the comparison.

System	Ranking	RR	AP
A	$\langle$10000 00000$\rangle$	1.0	0.11
B	$\langle$01111 11111$\rangle$	0.5	0.79

Figure 3.9: Example document rankings and their RR and AP scores. The number of relevant documents R is assumed to be 9.

tion 3.3.6. One rule of thumb is that a τ above 0.9 indicates that system rankings are effectively equivalent (Voorhees, 2001). By this rule, no two of the metrics reported give equivalent rankings. The choice of metric is therefore of practical significance, and we proceed to consider several proposed criteria for making this choice.

Correlation with user experience

It was observed in Section 3.1 that the test collection method is an abstraction of the user search experience. This suggests that the best metric is the one that correlates most closely with user satisfaction or utility. The results of overall comparisons between metric scores and user experience have, however, been mixed. Some have found a reasonable correlation (Huffman and Hochster, 2007); others have failed to detect one except in extreme cases (Turpin and Scholer, 2006); and others still have had ambiguous results (Al-Maskari et al., 2007). As it has proved less than straightforward to detect correlations between any metric and user experience, it would be even more demanding to discriminate metrics by their correlation. There has, though, been some promising work with click-through data, which, though difficult to interpret, is (for operational systems) voluminous (Chapelle et al., 2009; Zhang et al., 2010).

Metric predictivity

Even if the correlation between metric and user experience could be accurately measured, it does not necessarily follow that the metric with the strongest correlation is the most reliable one. We have argued this case in Webber, Moffat, Zobel, and Sakai (2008c); space limitations allow only a brief summary here. When evaluating a system against a set of topics, what is sought in not the system's performance on those topics as such, but rather a prediction of the system's performance on all topics; and it is possible that a metric m might be a better predictor than a metric n of user satisfaction on other queries, even if metric n correlates more strongly with satisfaction on the particular evaluated queries.

	P@10	RR	RBP.95	AP	nDCG
P@10	0.64	0.48	0.64	0.64	0.64
RR		0.36	0.48	0.47	0.47
RBP.95			0.68	0.70	0.69
AP				0.80	0.79
nDCG					0.80

Table 3.2: Predictivity of different metrics on the top 75% of TREC 2004 Terabyte track systems. Predictive power is the mean Kendall's τ on system rankings resulting from 2,000 random partitionings of the 50-topic set into two 25-topic subsets. Higher values mean greater predictivity.

Take, for instance, the reciprocal rank (RR) metric, Assume that the metric perfectly represents the user experience. Consider the rankings given in Figure 3.9. System A has returned only one of the nine relevant documents, but at rank 1; System B has returned all nine of the relevant documents, but beginning from rank 2. System A outscores in RR, System B in AP. By our assumption, RR measures user satisfaction more accurately than AP for this query. It appears, however, that (on this very slight amount of evidence) System B is the more reliable system, given the weight of relevant documents it finds, as reflected in the AP score. Therefore, System B is more likely to outperform System A on other queries, even as measured by RR, and therefore to give better overall user satisfaction. This can be expressed by saying that while RR might be a good *user metric*, AP is a better *system metric*.

In Webber et al. (2008c), we examine the *predictivity* of the simple metrics P@10 and RR, and of the more complex metrics RBP, AP, and nDCG. Predictivity is measured as the correlation of system rankings between two different topic sets, under either the same metric or different metrics. The results for the TREC 2004 Terabyte collection are given in Table 3.2; similar results were observed on the TREC 8 Ad-Hoc collection. The more complex metrics are more predictive of RR than RR is of itself, and as predictive of P@10. Even if one metric correlates perfectly with user satisfaction, there can be circumstances where another metric is preferable.

Plausibility of user model

Rather than empirically correlating user satisfaction and metric score, a metric can be assessed by the plausibility of the user model underlying it. We have observed that RBP is based on such a user model, one of utility modulated by persistence; and DCG has a (less clearly formulated) user model of gain and effort behind it, too. Recall-based metrics like AP have been criticized for their lack of a plausible user model, as observed in Section 3.2.2, while others have sought to defend AP by identifying a user model behind it (Robertson, 2008b). Similarly, precision at ten can be viewed as modelling a user that looks at every entry on the first page of search results, and no entries beyond, while a user model for reciprocal rank has been suggested in the previous section. Basing metrics on user models is a natural extension of the modelling approach on which the test collection method is based. Plausibility is not, however, a well-defined measurement; it is difficult objectively to determine that one metric's user model is more plausible than another's.

Statistical characteristics

Correlation with user satisfaction and plausibility of user model are both user-centric criteria for metric meta-evaluation. Other criteria are more system-centric or statistical in nature. Metric predictivity, discussed above, is one such; we review others briefly here. Aslam et al. (2005) suggest using maximum entropy, wherein the best metric is the one whose score places the strongest constraint on the possible document rankings it is derived from. Not surprisingly, the more complex the metric, the greater the constraints it imposes, with (for instance) average precision being more constraining than precision at ten.

Other meta-evaluation approaches focus on statistical stability and predictivity. Sakai (2006) proposes that metrics should be assessed on their discriminative power; that is, the proportion of differences between systems that are statistically significant under a metric (see Section 3.3 for a discussion of statistical significance). A more special-purpose measure of stability, called the swap rate, is proposed by Buckley and Voorhees (2000). The swap rate (described in more detail in Section 3.3.6) measures how often an ordering of two systems on one set of topics is swapped on another. Discriminative power and swap rate both measure metric consistency, and report AP and nDCG as stabler than the simpler P@k metrics.

Related to metric stability is the score variability between different systems (the *system effect*) compared to the variability between different topics (the *topic effect*) (Tague-Sutcliffe and Blustein, 1994; Banks et al., 1999). It is desirable to maximize system effect and minimize topic effect, so that differences in system scores stand out. An analytical framework for addressing this question, that of *generalizability theory*, was introduced to the field by Bodoff and Li (2007), and applied by Kanoulas and Aslam (2009) to derive empirical gain and discount factors for use in nDCG. Score standardization, discussed in Chapter 4, eliminates the topic effect on absolute scores (though not on score deltas) for a closed set of systems, and greatly reduces it in an open set.

An issue of metric meta-evaluation that has attracted relatively little attention is the tradeoff between accuracy and effort in evaluation. Sanderson and Zobel (2005) address this question, using a combination of the swap rate (Buckley and Voorhees, 2000) and statistical significance to investigate whether it is more stable, given a fixed assessment budget, to assess a few topics deeply, or a large number shallowly. Their finding is that a broader, shallower evaluation gives a more stable evaluation. A more general approach to planning the amount and assignment of assessment effort for an experiment requires the use of the statistical technique of *power analysis*. We examine the application of power analysis to information retrieval evaluation in Chapter 5. Other approaches to stretching assessment budgets are discussed in Section 3.4.2.

3.2.6 Scoring the system

We have described several evaluation metrics, and examined criteria that can be used in choosing between them. The purpose of an evaluation metric is to convert the relevance vector for a ranking into a numerical score. These per-topic scores are then aggregated into a single score for the system against the test collection, typically by taking the arithmetic mean. Figure 3.10 illustrates the process of taking the thousands of individual relevance scores, themselves a radical reduction of the complex relationship between a document and an information need; reducing each relevance vector to a single summary score, which for most metrics is essentially a weighted average of the ranking's relevance mass; then further reducing these topic scores to a single collection

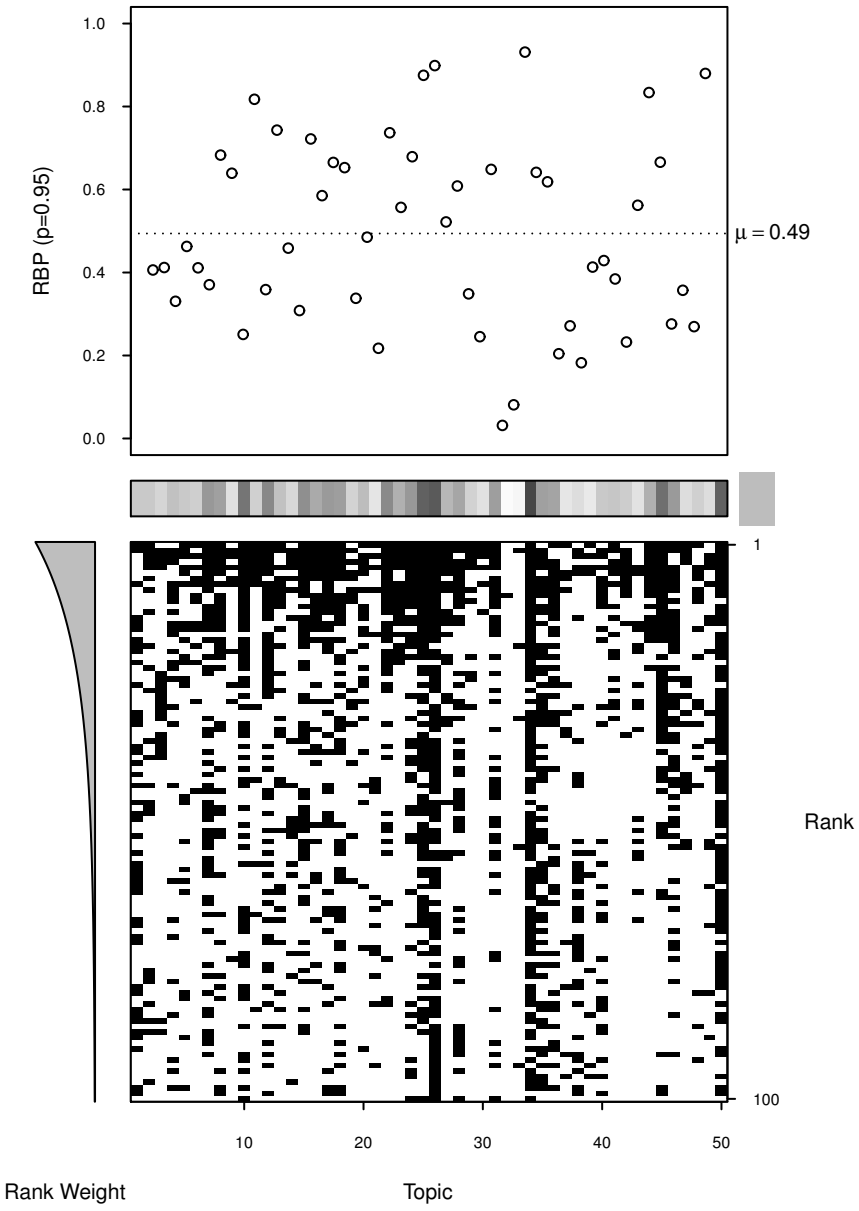


Figure 3.10: Scoring a system. The system is `CL99XT` from the TREC 8 AdHoc Track. The bottom part shows the relevance vectors; rows are ranks, columns are topics. Relevant documents are marked in black, irrelevant in white. The evaluation metric applied is RBP, $p = 0.95$. The gradient on the left gives the weight of each rank under the metric. In the middle of the figure, the relevant vectors have been scored; the shade of the box indicates the score achieved, with darker shades representing higher scores. As with most precision-based metrics, RBP is a weighted summary of the relevance proportion in the relevance vectors. On the right is the shade representing the system's mean score. On the top part of the figure, the per-topic RBP scores have been plotted, and the mean score of 0.49 marked.

score, which represents the system's performance.

Alternative topic score aggregations than the arithmetic mean have been proposed in the literature. The most common is the geometric mean (Robertson, 2006), suitably adjusted to handle zero scores, although the harmonic mean or the median are also candidates (Ravana and Moffat, 2009). The theoretical choice between them depends on whether metric scores are on an interval or ratio scale (Stevens, 1946). No compelling general answer has been given, but it is plausible that a given score difference has more significance at low scores than at high ones; that outscoring another system 0.05 to 0.02, for instance, means more than outscoring it 0.85 to 0.82. The geometric mean places greater weight on absolute differences amongst lower scores, and has been adopted as a way of emphasizing hard queries compared to easy ones, for instance in the Robust track of TREC (Voorhees, 2004). Score standardization, presented in Chapter 4, is a more direct and consistent solution (Ravana and Moffat, 2008).

A system's aggregate score is used primarily to compare one system's performance against that of other systems. The variability of topic and therefore collection difficulty means that this comparison must be made on the same collection, unless score standardization is employed (Chapter 4). Even on the one collection, though, it is misleading to simply compare the aggregate scores. What is required instead is a prediction of how reliably the observed outperformance of one system over another can be generalized from the limited sample of topics in the test collection to the broader population of topics that would be encountered in operation. Measuring this generalizability requires the tools of statistical analysis; we turn to this topic next.

3.3 Statistical analysis

We have discussed the model and method of system evaluation through test collections (Section 3.1), and the use of evaluation metrics to calculate a system's effectiveness score (Section 3.2). Two retrieval systems are compared by the effectiveness scores they achieve on the one collection; but the comparison only applies directly to the collection's topics, which are a small subset of all the topics that the systems would have to process when deployed. The question then becomes how confidently the performance comparison can be generalized to the full population of topics. We consider the question of generalization in this section.

3.3.1 Particular results, general conclusions

The top half of Figure 3.11 shows the result of a typical comparative retrieval experiment. Two systems, A and B, have been run against the one test collection, and their average precision scores calculated on each topic. Because the same topic set has been used for each system, the scores can be paired by topic. Each system's topic scores are averaged to produce a system score against the collection. System A has achieved a mean score of 0.37, compared to 0.31 for System B. System A has therefore outperformed System B on this collection, as measured by mean average precision.

An alternative view of these results is to look at the score differences, or *deltas*, between the two systems on each topic, shown in the bottom half of Figure 3.11. Examining deltas clarifies the differences between the two systems, both visually and statistically, but loses information about absolute scores. We might, for instance, want to interpret the near-zero delta of the lowest-scoring topic differently from the near-zero delta of the sixth-highest scoring one; this is the sort of question that taking the

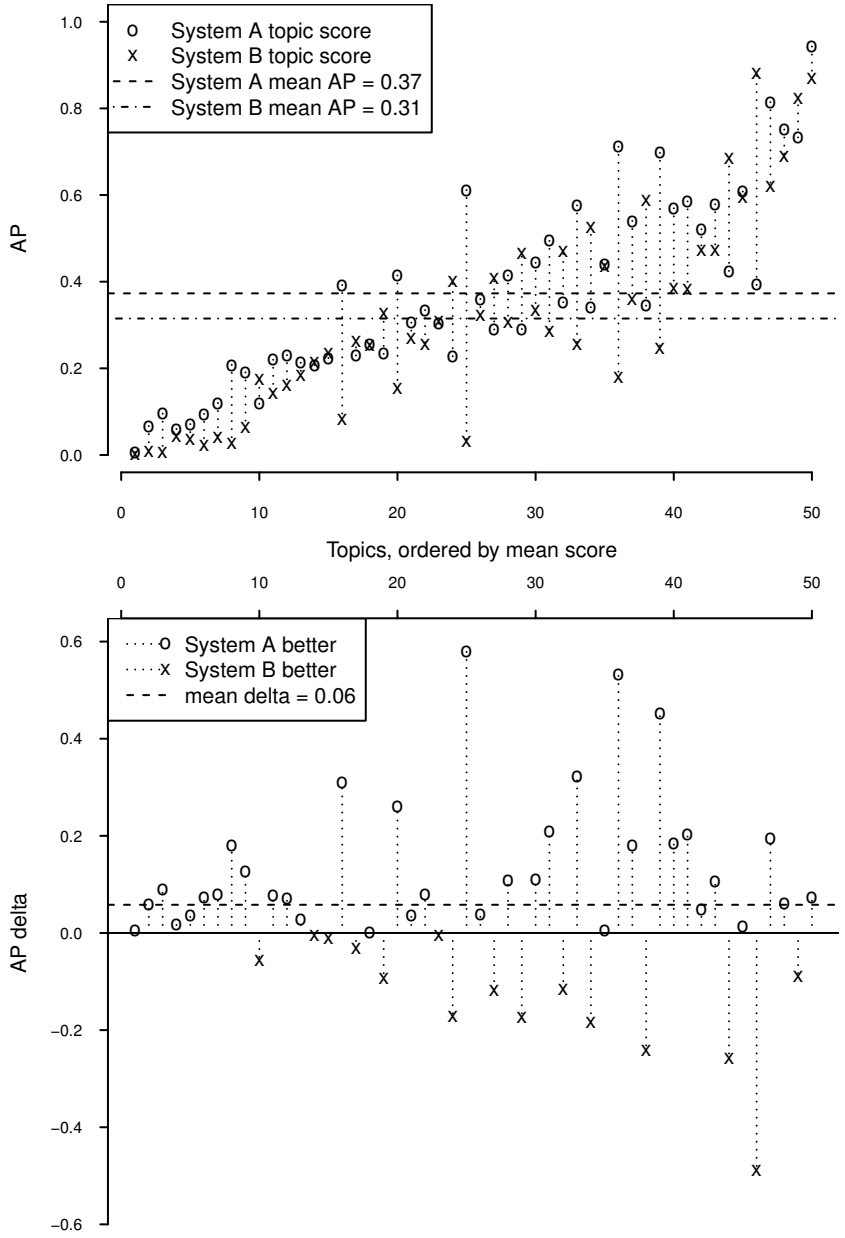


Figure 3.11: Per-topic and mean AP scores achieved by two participant systems for the TREC 8 AdHoc track (top), and the per-topic differences between those scores (bottom). Topics have been ordered by the mean of the per-topic scores for the two systems. In the lower figure, a positive delta indicates System A outperformed System B, a negative delta the reverse. System A is `CL99XT`; System B is `apl8c221`.

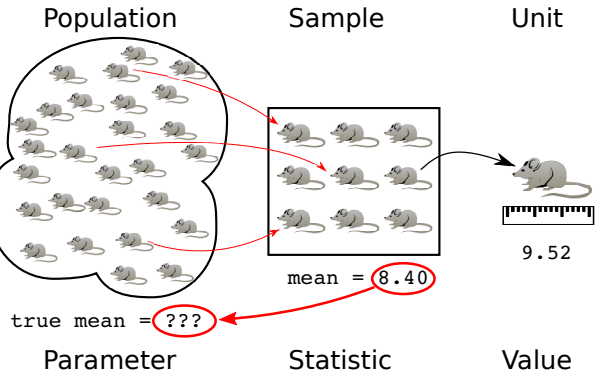


Figure 3.12: The standard model of statistical inference. The experimental units are regarded as a sample from a larger population. The mean of the observed values, or statistic, on the sample is used to infer the mean, or parameter, on the full population.

geometric mean partially addresses.

If the topics and documents in the test collection were the only topics and documents that these two retrieval systems were ever to be run on, then the experimental evaluation would be complete. In reality, however, retrieval systems operate on a wide range of queries and documents, of which the particular instances contained in the test collection are merely representative. As mentioned previously, the interest is not so much in how well the systems perform on the test collection in particular, but rather what these results predict of retrieval performance in general.

Generalization of retrieval scores can be considered along many dimensions. Robertson (1981) lists several (for instance, generalizing from current documents to documents in five years' time), and Cormack and Lynam (2006) calculate confidence intervals for retrieval scores over different document samples. In practice, though, query generalization is the most studied dimension, since the query set is, due to resource constraints, far more limited in size than the document corpus. Therefore, the key question is how confidently the comparative effectiveness achieved on the collection's query set can be generalized to all queries.

Generalization is tied to variability: the greater the variability in the results, the less confidently they can be generalized. We may have more confidence in a small margin, if it is consistent across topics, than in a larger one, if the consistency is lacking. A form of variability obvious in the top part of Figure 3.11 is in absolute topic scores. This topic variability is partially controlled by pairing scores and taking deltas, as shown in the bottom part of Figure 3.11; it can more generally be managed by score standardization, described in Chapter 4. Even with paired scores, however, much variability remains. For instance, although System A has a higher score overall, System B still manages to beat it on fifteen of the fifty topics. The magnitude of the score deltas is also highly uneven; if signs were switched on just the three topics with the highest positive score delta, then System B would have the higher mean score. Given these variabilities, how confident are we that System A's higher score has not occurred from chance in the choice of topics?

3.3.2 Population, sample, statistic, parameter

Questions of the generalization of experimental results are addressed using the tools of statistical inference; but first, the problem must be mapped to the appropriate model. In generalizing from the test query set to other queries, the test set is modelled as a *sample* from the full *population* of queries. Where queries are taken from an operational query stream, then the population is the query stream. Where queries have been explicitly created, the population can (rather unsatisfactorily) be defined as "all queries that might have been created by a similar process", in the hope that the process is representative of the formulation of authentic user queries. Each query in the sample is an experimental *unit*. It is assumed that the sample has been drawn by (uniform) random sampling. This assumption is crucial, because it allows probabilistic inferences to be made from sample back to population.

For each unit, or query, in the sample, an experimental *value* is observed. For comparative evaluations between two systems on the same collection, the value is the score delta on the query. A summary function or *statistic* is calculated over the unit values. Figure 3.12 summarizes this inferential model. In Figure 3.11, if the statistic were the arithmetic mean, then the statistic's value would be $+0.06$, the mean score delta. Alternatively, if the statistic were the proportion of queries on which System A outscored System B, then the statistic's value would be 0.7 (35 out 50). The quantity of interest to the experimenter is the value of some summary function or *parameter* on the population. Typically, but not always, the statistic on the sample and the parameter on the population are the same function. The statistic on the sample is then used to infer the parameter on the population. If the parameter were the mean, then the value sought would be the *true mean delta*, and the informal question "is System A really better than System B?" translates to the statistical question "is the true mean delta between System A and System B positive?".

3.3.3 Statistical significance tests

It is not possible to determine with certainty the parameter on the population from the statistic on the sample (unless the whole population is sampled), nor yet definitely to answer the question of whether one system is really better than another. In our running example, we may, by extreme chance, have chosen the only thirty-five queries in the whole population on which System A outperforms System B. It is, however, possible to give a probabilistic answer. The manner of doing so is analogous to the proof by the *reductio ad absurdum* of a contradiction in mathematical logic: we propose a hypothesis, called the *null hypothesis*, that denies the assertion we wish to test, and then see how probable it is that the observed statistic (or greater) would have occurred if the null hypothesis were true. This latter probability is termed the p value of the test; if it is below some threshold α, then we reject the null hypothesis, not as disproven, but as implausible, and conclude that the result is *statistically significant* at the α level. Conventional values of α are from the set $\{0.05, 0.01, 0.001, \ldots\}$; the smaller the value of α, the more stringent the test.

Sign test

The workings of statistical significance can be illustrated with a simple but elegant hypothesis test called the *sign test* (Gibbons and Chakraborti, 2003, Chapter 5). The sign test is a test of proportions; specifically, of the proportion of values in the sample

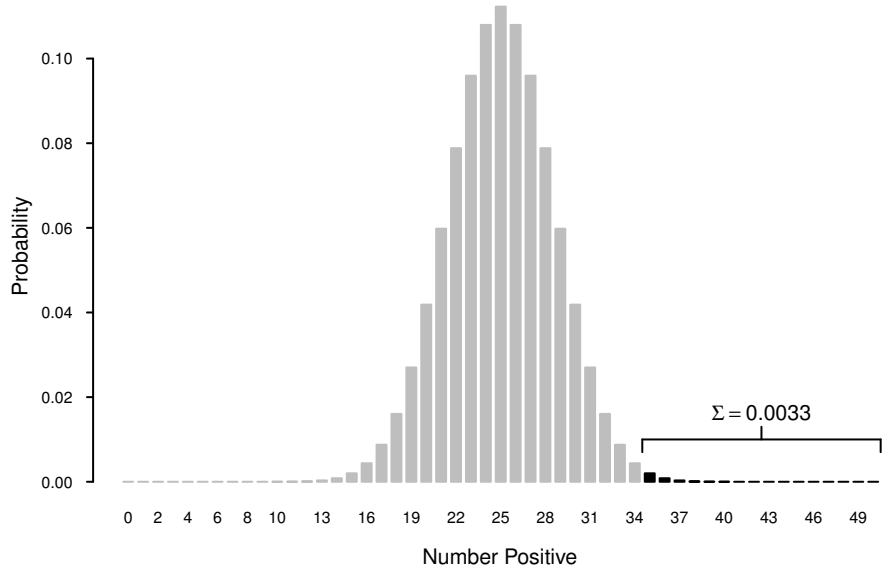


Figure 3.13: Binomial distribution for sampling 50 elements from a population with proportion 0.5 positive. Samples having 35 or more positive elements are marked.

and population that have a given (positive or negative) sign. Let q be the (unknown to us) proportion of positively signed values in the population; in the running example, the proportion of queries on which System A gets a higher score than System B, with ties ignored. A randomly drawn value V then has probability q of being positive. Therefore, V is a random variable following a Bernoulli distribution with parameter q—colloquially, a biased coin with probability q of turning up heads (positive), and $1 - q$ of turning up tails (negative). Now, if we randomly draw a sample of size n from the population (flip the coin n times, or select n queries at random from our query stream), then the probability of getting m positive values is $\binom{n}{m} q^m (1-q)^{n-m}$. In other words, the number of positive values follows a Binomial(n, q) distribution. This is the *sampling distribution* of the statistic; that is, the distribution giving the probability that a statistic (here, proportion) of a random sample falls on a particular value or (for continuous values) within a certain range.

The observed statistic in the running example is that System A outscores System B on 35 out of 50 queries. What then is to be tested is whether System A is truly better than System B on a majority of queries in the population. The null hypothesis follows immediately: it is that the two systems are as good as each other; System A outperforms on half the queries in the population, System B on the other. The null hypothesis is tested by determining the probability, having sampled 50 values from a population with 50% positive, of at least 35 values in the sample being positive. This probability gives the p value of the test, and can be directly determined from the Binomial sampling distribution as $p = 0.0033$, as illustrated in Figure 3.13, making the test significant at the $\alpha = 0.01$ level. If that level is regarded as sufficiently stringent, then the null hypothesis is rejected, and the result found to be statistically significant.

The test as carried out above is a *one-tailed* test: only the upper tail of the sampling distribution, namely that of majority positive values, is considered. The alternative, *two-tailed* test examines the probability that at least 35 of the values would be of the

same sign, positive or negative. The two-tailed test is more stringent than the one-tailed test. For a symmetrical distribution, the p value of the former is twice that of the latter; here, that is $p = 0.0066$, still significant at the $\alpha = 0.01$ level.

Bootstrap test

The sign test is easy to understand and calculate, but it is a test only on proportions, not on the score mean; it ignores information about the magnitude and variability of score deltas. Performing a significance test on the mean requires estimating the sampling distribution of the mean. Unlike the proportion, the mean's sampling distribution does not follow directly from the null hypothesis; not merely the hypothesized mean, but also the shape and dispersion of the population is required. One way to estimate the sampling distribution of the mean, and of other statistics as well, is a resampling method known as the *bootstrap test* (Efron and Tibshirani, 1993; Savoy, 1997).

The intuition behind bootstrapping is as follows. If the distribution of the population were known, then repeated samples could be drawn from it to empirically approximate the sampling distribution of the mean. Unfortunately, the distribution of the population is not known; absent other information, the best estimate of it is given by the distribution of the observed sample. Therefore, sampling from the population can be simulated by resampling, with replacement, from the sample. The null hypothesis for tests of the mean is that the true mean of the population (here, of score deltas) is zero, so the distribution of resampled means is shifted to centre around zero. Then, the proportion of resampled means that are higher (one-tailed) or more extreme (two-tailed) than the observed mean gives the p value of the significance test.[2]

Figure 3.14 displays the estimated sampling distribution of the mean, derived from $5{,}000$ resamplings of the topic AP deltas between System A and System B from the running example. The size of each resample is 50, the same size as the original sample. The estimated distribution centres around zero; this follows from the null hypothesis of a true mean population delta of zero. A two-tailed test is performed by counting the proportion of resampled means whose absolute value is equal to or greater than the mean of the original sample; this proportion is the p value of the test. Here, 68 of the $5{,}000$ resamples have means less than or equal to the negative of the observed mean, and 81 have means greater than or equal to the observed mean. Therefore, the overall p value is $(68 + 81)/5000 = 0.0298$, making the result significant at the $\alpha = 0.05$ level, but not at the $\alpha = 0.01$ level. The achieved significance level is weaker for the bootstrap than for the sign test; the variability in delta magnitudes in Figure 3.11 makes it more plausible that the observed positive mean delta for System A occurred by chance. The bootstrapped sampling distribution here is not symmetric, but has a slight positive skew; this reflects the positive skew in the sample (the sample mean is 0.058, higher than the median of 0.053).

The bootstrap test is straightforward to implement, can be applied to other statistics than the mean, makes no assumptions about the distribution of the population, and can be used with any sample size. It is, however, computationally intensive, which made it impracticable until recent decades, and is still an issue where very many significance tests need to be performed (for instance, in the inner loop of a simulation). Additionally, as a randomized method, the precise p values, and in marginal cases the significance levels, vary from resample to resample.

[2] Greater stability can be achieved by using a *studentized statistic* rather than the plain mean; see Efron and Tibshirani (1993, Chapter 16) and Davison and Hinkley (1997, Chapter 4) for details.

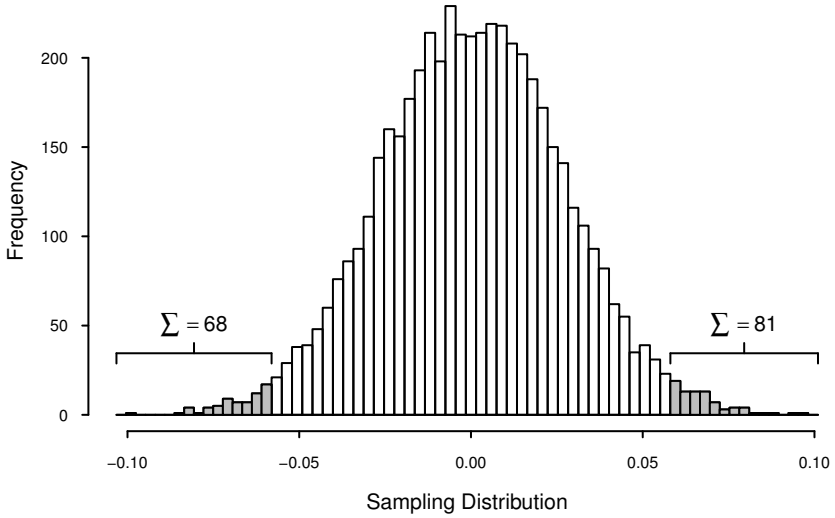


Figure 3.14: Bootstrap distribution and two-tailed significance test for the two systems shown in Figure 3.11. A total of $5{,}000$ bootstrap resamples have been taken to derive the sampling distribution. Of the resamples, 68 have a mean less than the negative of the observed mean, and 81 have a mean greater than the observed mean. The p value of the test is therefore $(68 + 81)/5000 = 0.0298$.

The t test

For the sign test, the sampling distribution is fully determined by the null hypothesis; for bootstrapping, the distribution is simulated by resampling. It is also possible in some circumstances to determine the sampling distribution of the mean by theoretical methods. We examine two closely related methods. Both make use of the *normal distribution*; the shape of this often-encountered distribution is displayed in Figure 3.15.

It can be shown that the mean of a random sample taken from a normal population follows a particular distribution, known as the t distribution. The t distribution is parameterized by its *degrees of freedom*. It resembles a fat-tailed normal distribution; the shape becomes more normal as the degrees of freedom increase. Figure 3.15 gives sample t distributions, along with a standard normal distribution. Stated more precisely, if $\overline{X}$ is the mean of a random sample of n elements, with sample standard deviation S, drawn from a population that is normally distributed with mean μ, then the statistic:

$$T = \frac{\overline{X} - \mu}{S/\sqrt{n}} \tag{3.14}$$

follows the t distribution with $n - 1$ degrees of freedom (Wasserman, 2004, Chapter 10). As the sample size n increases, the sampling distribution of the mean becomes less fat-tailed, essentially because the estimate of the standard deviation becomes more accurate. The relationship between the normal and t distributions can be directly used to test significance, provided we know or can reasonably assume that the population is normally distributed. Values of $\overline{X}$ and S from the sample are fed into Equation 3.14, and μ is set to 0, as dictated by the null hypothesis. The resulting value of the T statistic is then compared against the cumulative t distribution, with the appropriate degrees of

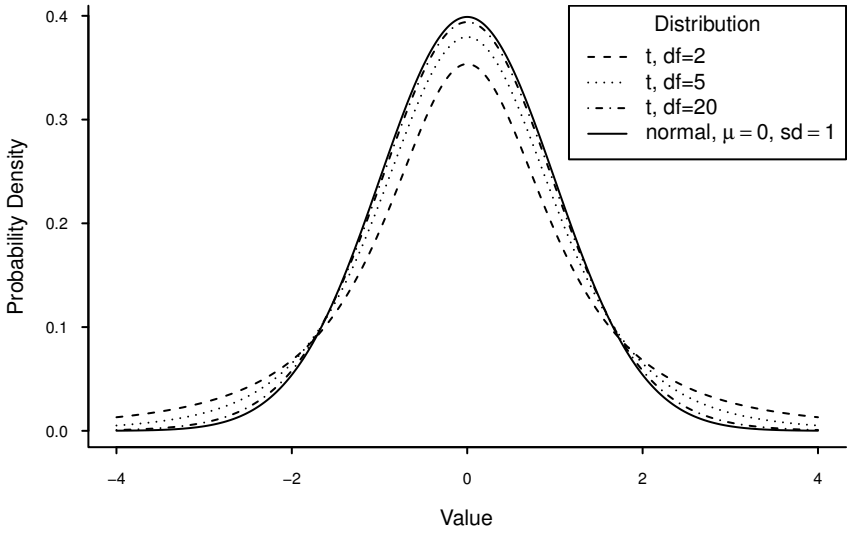


Figure 3.15: Example t distributions with various degrees of freedom, and the standard normal distribution. The area under each of these curves is 1. If the probability density function of one of these distributions is $f(x)$ (the "Probability Density" given on the y axis), then the probability of a random value x falling in the range (a, b) is $\int_a^b f(x)\mathrm{d}x$.

freedom. The p value of the test is, as usual, the proportion of this distribution with values equal to or more extreme than that observed in the sample; the more extreme the T statistic, the small the p value. This method constitutes the t test for statistical significance.

Unfortunately, the underlying population cannot always be assumed normal; and there is no reason to believe that evaluation metric score deltas are normal, making the above justification for using the t test invalid. Another feature of the sampling distribution of the mean can, however, be applied; namely, that the mean of a sufficiently large sample is approximately normal in its distribution, whatever the distribution of the population. This is known as the *central limit theorem* (CLT), and can be more formally stated as (Wasserman, 2004, Chapter 5):

Theorem 3.6 ***Central Limit Theorem****: the distribution of the mean of a random sample of n independent and identically distributed random variables with finite mean μ and variance σ^2 becomes, for large enough n, approximately normal in its distribution, with mean μ and variance σ^2/n.*

A significance test on the mean can be derived from the central limit theorem, provided the sample is sufficiently large; a common rule of thumb for sufficiently large is a sample size of 30 or more. The test statistic is identical to that for the t test, given in Equation 3.14. Formally, the p value is calculated from the normal distribution; but since the normal and t distributions are almost identical for large samples, in practice the t test is frequently employed.

Alternative tests

Other hypothesis tests are available. In the Wilcoxon sign-rank test, absolute score deltas are ranked; a test statistic is then calculated over the signed ranks, and compare to a theoretical distribution under the null hypothesis to determine significance. Both the sign test and the Wilcoxon test are simplified forms of *permutation tests*; they compute every combination of signs in the sign test, or rank in the Wilcoxon test, and observe what proportion give a statistic (proportion or sum of ranks) greater than that observed in the sample. A permutation test could in theory be applied directly to the score deltas; but for large sample sizes, the number of possible permutations is too great, over 10^{15} for a sample size of 50. Instead, in the *randomized permutation test*, a certain number of permutations are randomly generated, and significance estimated based on these sub-samples (Smucker et al., 2007).

Several hypothesis tests have been described above. The randomized and bootstrap tests are attractive for the minimal assumptions they make. In practice, though, test collections usually have at least 50 queries, which is sufficient for applying the t test under the central limit theorem. Smucker et al. (2007) compare the bootstrap, randomized, t, and Wilcoxon tests on TREC data. They find the former three to give very similar results, and the Wilcoxon test to be less reliable. Because of its simplicity of calculation, and the determinacy of its results, the t test is the significance test employed here, while resampling methods similar to the bootstrap are used for experiments that go beyond simple significance testing.

3.3.4 Achieving significance

The formulation of the T statistic in Equation 3.14 shows three factors in achieving significance: the mean, standard deviation, and size of the sample. The sample size is the factor most directly under the experimenter's control. A doubling of the standard deviation, though, requires a quadrupling of the sample size to compensate, so reducing sample variability is crucial, if it can be done. We have already observed in Figure 3.11 on page 50 an important technique for variability reduction; namely, pairing topic scores and taking the deltas. A *paired test* helps to control the variability in difficulty between topics, which Figure 3.8 on page 44 shows to be very high. If the systems were run against different topic sets, then a *two-sample test* would be required. In this case, the enormous variability in topic difficulties, unrelated to the quality of the systems, would increase the apparent variability in the sampled scores, making significance much more difficult to achieve. For instance, a paired t test on the systems shown in Figure 3.11 gives a p value of 0.035; but if the scores are not paired, then a two-sample t test gives a p value of 0.202, well short of significance. Topic variability therefore makes comparing scores between different collections difficult; controlling variability through score standardization, described in Chapter 4, addresses this problem. In addition, Chapter 5 examines the estimation of likely significance during retrieval experiment design.

3.3.5 Confidence intervals

Significance tests of the mean give the probability that the observed result could have happened by chance. Also of interest is determining the interval within which the true mean falls, with a given probability. Such an interval is known as a *confidence interval*.

The calculation of a confidence interval is most easily illustrated using the bootstrap method. Consider again the distribution of bootstrapped means in Figure 3.14. It is straightforward to find the 2.5th and 97.5th percentiles of these means. This, centered around the observed mean, might seem to give us a 95% confidence interval, but there is a subtlety: if the true mean is above the observed mean, then it is the lower tail of the displayed distribution that must be tested against; if below, the upper tail. The 2.5th percentile of Figure 3.14 is -0.051, the 97.5th percentile 0.053, which gives a 95% confidence interval of $(0.005, 0.109)$ on the true mean delta between System A and System B. The full range is positive, which accords with the finding that System A is significantly better than System B; the calculations of significance and of the confidence interval, though, are not identical.[3] Confidence intervals can be similarly calculated based on the t distribution, provided the standard requirements (normal population or large sample) are met.

The above confidence intervals are on score deltas between system pairs. If there are n systems, then there are $n(n-1)/2$ pairwise confidence intervals to consider. A confidence interval can also be calculated on the mean score of a single system. It would be convenient to use the overlap between these n confidence intervals as at least a rough indication of the pairwise confidence intervals, and indeed of the pairwise significances. The confidence intervals on the mean, however, are greatly widened by topic variability, compared to those on the (paired) scores deltas, just as the two-sample significance test is far weaker than the paired test. So, for instance, the 95% interval on the t distribution for the mean of System B in the running example is $(0.249, 0.381)$, while for System A it is $(0.311, 0.435)$. These intervals overlap by a wide margin, with each system's mean score falling within the other's confidence interval, even though the systems are significantly different in a paired test. In Chapter 4, confidence intervals on mean scores are narrowed to a width more indicative of the paired intervals through score standardization.

3.3.6 Rank similarity measures

In meta-evaluative studies, we frequently wish to compare test environments for consistency or stability, by changing the evaluation metric, say, or the set of topics. One ground for comparison is between the rankings that each environment induces over a set of systems via the systems' effectiveness scores. Such analysis requires a measurement of the similarity between a pair of rankings. Several such rank similarity metrics are described in this section.

Pearson's correlation coefficient

In some cases, not just ranks are available, but also the scores that determined those ranks; this is generally the case with systems ranked by effectiveness scores. It is then possible to measure the *correlation* between scores, rather than just the similarity between rankings. The standard measure of correlation is Pearson's product-moment correlation coefficient, denoted r. Let $i \in \{1, \ldots, n\}$ identify the ranked items, and X and Y the two rankings, such that X_i is the score that item i achieves in ranking X,

[3] There are a range of methods to improve the accuracy of this basic bootstrap interval; see Efron and Tibshirani (1993, Chapters 12–14) and Davison and Hinkley (1997, Chapter 5).

and Y_i its score in ranking Y. Then the correlation coefficient is:

$$r = \frac{\text{cov}(X,Y)}{\sigma(X)\sigma(Y)} = \frac{\sum_{i=1}^{n}(\overline{X} - X_i)(\overline{Y} - Y_i)}{\sqrt{\sum_{i=1}^{n}(\overline{X} - X_i)^2}\sqrt{\sum_{i=1}^{n}(\overline{Y} - Y_i)^2}}\ , \tag{3.15}$$

that is, the covariance of the scores of the two rankings divided by the product of their standard deviations ($\sigma(\cdot)$, read "sigma"). If $Y_i = a \cdot X_i + b$ for $a > 0$ and any b, then $r = 1$; graphically, the scores lie on a straight, upward-sloping line, and the two rankings are said to have perfect positive correlation. If $Y_i = -a \cdot X_i + b$, then the rankings have perfect negative correlation. The degree to which r is less than 1 and more than -1 measures the degree to which the scores of the two rankings deviate from this perfect linear relationship. If the scores were chosen randomly, then the expected correlation would be 0.

Unlike purely rank-based measures, Pearson's correlation is sensitive to the magnitude of score deltas, penalizing swaps against large deltas more than against small, and penalizing delta changes when no swap occurs, too. The coefficient assumes, however, that perfect correlation is linear; that is, that the scores of one ranking are a constant multiple plus a constant factor of the other. Experimenters are often reluctant to make this linear assumption between system rankings—perhaps two metrics have a perfect but curvilinear relationship. Also, experimenters sometimes care only about actual changes in rank, not in score. Finally, there are frequent cases in which only ranks, not scores, are available.

Kendall's τ

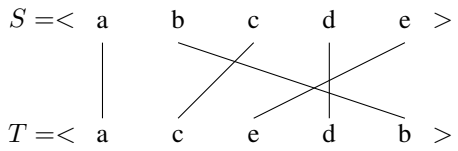
Similarity measures that make use only of rank information are *rank similarity measures* in the proper sense. One such measure is Spearman's ρ (read "rho"), based on the square of the distance between the ranks of an item; this measure is equivalent to Pearson's correlation calculated over the ranks, rather than the raw scores, of the items. An alternative, known as Spearman's footrule, instead measures the unsquared or L1 distance between ranks.

A more popular rank similarity measure in retrieval evaluation is Kendall's τ (read "tau"). Kendall's τ is calculated by counting the number of concordant and discordant pairs between the two rankings. A *concordant pair* is where two items i and j are placed in the same relative order in both rankings (i above j in both, or i below j in both); a *discordant pair* is one where the order differs. Ties need to be handled specially, and will be assumed in this exposition not to occur. Let t_c be the number of concordant pairs, t_d of discordant ones. Then, Kendall's τ is:

$$\tau = \frac{t_c - t_d}{t_c + t_d}\ . \tag{3.16}$$

If all pairs are concordant, then $\tau = 1$, and the rankings are identical; if all pairs are discordant, then $\tau = -1$, and the rankings are reversed. If concordant and discordant pairs are evenly balanced, then $\tau = 0$. The latter would be expected if the observed items were randomly sampled from a population itself with $\tau = 0$; a significance test is available that tests this null hypothesis.

A working of Kendall's τ on two example rankings S and T is given in Figure 3.16. The set of concordant pairs is enumerated in $\mathcal{C}_{ST}$, while $\mathcal{D}_{ST}$ lists the discordant pairs. Discordant pairs can be found graphically by drawing a straight line between each item in S and the corresponding item in T, as is done in the figure; whenever two of these



$$
\begin{aligned}
\mathcal{C}_{ST} &= \{(a,b),(a,c),(a,d),(a,e),(c,d),(c,e)\} \\
\mathcal{D}_{ST} &= \{(c,b),(d,b),(e,b),(e,d)\} \\
C &= |\mathcal{C}_{ST}| = 6 \\
D &= |\mathcal{D}_{ST}| = 4 \\
M &= C - D = 2 \\
\tau &= M/P(n) = 2/(C+D) = 2/10 = 0.2
\end{aligned}
$$

Figure 3.16: Example working of Kendall's τ.

lines cross, the ordering of the respective items is discordant. The total number of pairs is the sum of C, the number of concordant pairs, and D, the number of discordant pairs, so τ is the proportion of these pairs that are concordant, linearly adjusted to the range $[-1, 1]$.

If a pair of items are selected at random from two rankings with a given τ, then the probability that the pair is concordant in the two rankings is:

$$P(\text{concord}) = \frac{t_c}{t_c + t_d} = \frac{\tau + 1}{2} \ . \tag{3.17}$$

The probabilistic understanding aside, though, interpreting a τ score in isolation is not straightforward. The significance test for positive relationship is not meaningful for most system rankings; it is implausible that two evaluation metrics, or system performance on two topics sets, would have no relationship. A proposed rule of thumb is that system rankings with a τ above 0.9 should be considered equivalent, whereas those with a τ below 0.8 display noticeable differences (Voorhees, 2001); but such rules are at best rough guides, affected as they are by the size and characteristics of the system set (for instance, defective systems with consistently low scores increase τ). Kendall's τ is more easily interpreted as a comparative measure; for instance, to determine which of two topic subsets gives a system ranking closer to that of the full topic set. The question of significance recurs, though: is the ranking on one subset significantly closer to the full ranking than on the other? While there are tests of significance for this case (Cliff, 1996), they are not widely used, and in any case set wide bounds, making significance difficult to achieve. Additionally, such significance tests assume that it is the systems that are randomly sampled, whereas in practice it is the topics that the experimenter wants to generalize over (Carterette, 2009).

Alternative measures

System rankings are compared so frequently in retrieval meta-evaluation that the field has produced several new measures. One family of these, the *swap rate*, counts how frequently system pairs swap order when evaluated against different query sets. In Buckley and Voorhees (2000), the swap rate is computed over parallel query formu-

lations. Voorhees and Buckley (2002) randomly partition the one topic set, and aggregate swap rates between the partitions into bins defined by system score deltas. Sanderson and Zobel (2005) only record swaps between systems with significantly different score deltas, and argue that the sampled query sets should not be disjoint.

A more specialized rank similarity metric is proposed in Carterette (2009): not only must the two rankings be of retrieval systems, but they must be over the same query sets; only the evaluation metric can differ. The proposed measure, denoted d_{rank}, takes account of the topic score correlations between systems. Swaps between dissimilar systems are more heavily penalized than between similar systems, as are swaps that separate systems with highly correlated topic scores. The measure provides a significance test over the population of topics.

When comparing system rankings, it is frequently more important that highly-ranked systems are similarly ordered than that lowly-ranked systems are. The τ_{AP} measure proposed by Yilmaz, Aslam, and Robertson (2008a) is *top-weighted* in this way. Like Kendall's τ, the measure penalizes discordant pairs, but the penalty for discordance is weighted by the reciprocal of the rank of the lower-ordered item—the same weighting scheme as average precision. The measure is not symmetric, since one of the rankings must be nominated as determining item ranks for weighting purposes. Melucci (2009) generalizes τ_{AP} to a family of measures, τ_*, in which arbitrary weights can be assigned to the lower of the pair of ranks in the objective ranking. Melucci further demonstrates that τ_*, and therefore τ_{AP}, are specializations of the class of weighted τ variants, τ_w, analyzed by Shieh (1998), in which each pair of ranks can be assigned its own weighting.

The rank similarity measures described above are primarily used for comparing system rankings. An even more common ranking in information retrieval is the ranking of documents returned by a retrieval system. Document rankings are non-conjoint, so the above measures cannot be applied. In Chapter 7, we combine the properties of non-conjointness, top-weightedness, and arbitrary continuability, to identify a class of *indefinite rankings*, and propose a metric of similarity between indefinite rankings, called *rank-biased overlap*.

3.3.7 Kernel density estimates

Various graphical data representations are used in this thesis, such as dot plots, line graphs, box and whisker plots, and so forth. It will be assumed either that the reader is familiar with these representations, or that their meaning can be worked out with the help of the summary notes provided. There is, however, one form of representation that will be used throughout the thesis which may not be so familiar; namely, the plot of a *kernel density estimate* (Silverman, 1986), which offers a kind of smoothed histogram.

A *histogram* is a plot of the distribution of values in a one-dimensional data set. The range of the values is divided up into equally-spaced bins, and the number of items falling into a bin determines the height of that bin's bar. The two parameters of a histogram are the width of each bin, and the offset of the first bin. Bin width affects the jaggedness or smoothness of the plot, while the choice of bin offset can have a major effect on the plot's shape. The effect of offset choice is illustrated in Figure 3.17, which displays the per-topic AP scores achieved by a particular TREC 8 AdHoc run, and Figure 3.18, which shows two histograms of the data set, differing only in the choice of offset. One offset results in an almost level histogram, shown in solid lines, up to an average precision of 0.5, while the other offset produces three peaks.

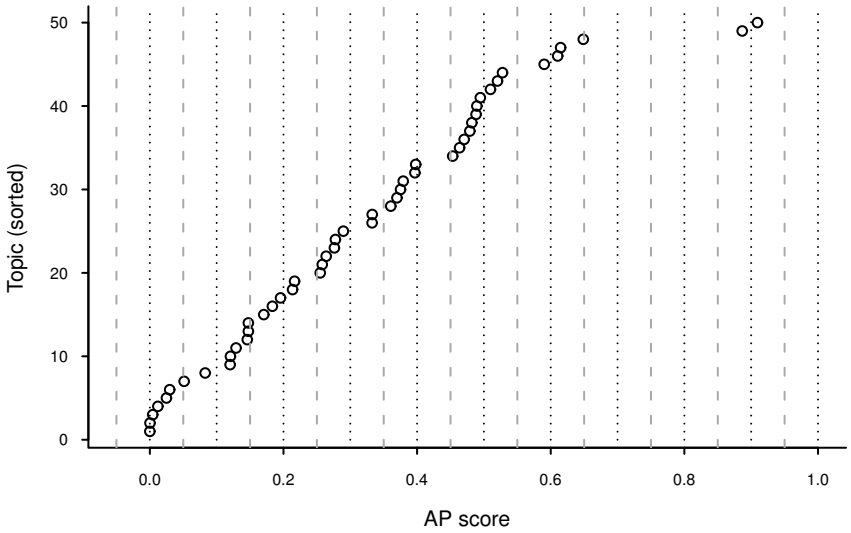


Figure 3.17: Individual AP topic scores for the system `Flab8atdn` run against the TREC 8 AdHoc collection. The dotted lines show the bin boundaries for the histogram shown in solid bordered, unfilled bars in Figure 3.18; the dashed lines show the alternative bin boundaries, marked in dashed, shaded bars used in Figure 3.18. Topics are sorted by increasing AP score.

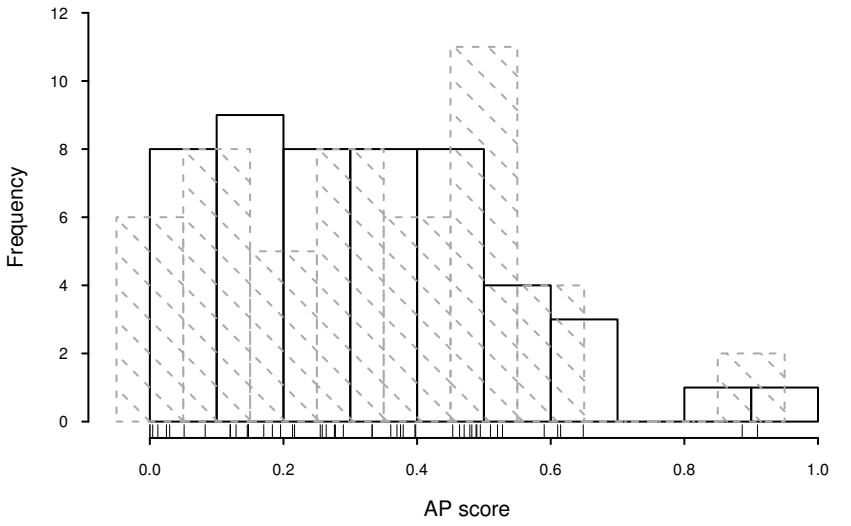


Figure 3.18: Score histograms, derived from Figure 3.17. The solid line, unfilled bars are for bins aligned to $0.1x$ boundaries, for $x \in \{0, \ldots, 10\}$; these correspond to the dotted dividing lines in Figure 3.17. The dashed line, shaded bars are for bins shifted to the left by 0.05; these correspond to the dashed dividing lines in Figure 3.17. The data points are shown at the bottom of the histogram as a 1-d plot.

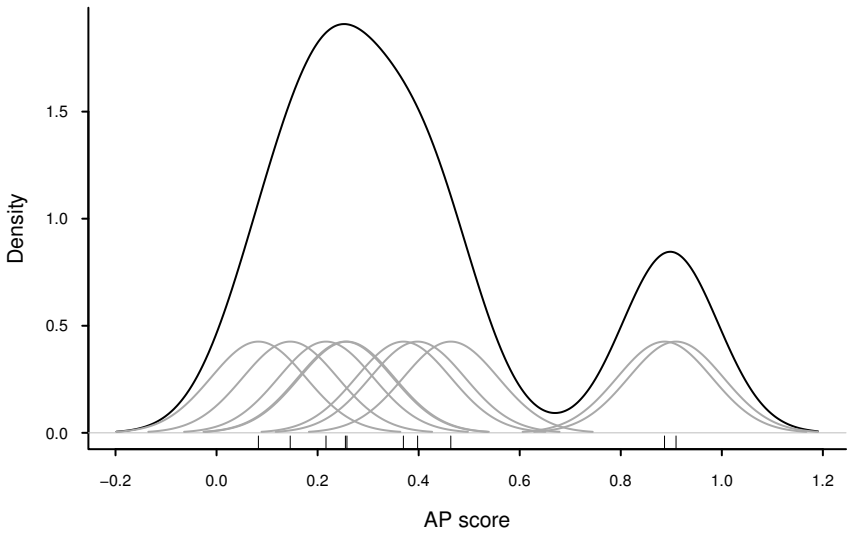


Figure 3.19: Density estimate over the per-topic AP scores of the system `Flab8atdn` on the first ten topics by topic id of the TREC 8 AdHoc collection. The individual Gaussian kernels are also plotted; note that there are two data points close together around 0.25.

An alternative way to present of the distribution of a data set is to use a kernel density estimate. This method gives each value in the data set a continuous contribution. The shape of this contribution is determined by the chosen *kernel function*, and its width by the selected *bandwidth*. The contributions are then summed to form a continuous curve. A common kernel function is a normal or Gaussian distribution. A choice of bandwidth must be made, just as a bin width must be chosen for a histogram; but there is no need to select offsets. Figure 3.19 illustrates a density estimate, on a subset of ten topic scores, using a Gaussian kernel, with the standard deviation of each kernel set to $\sigma = 0.9$ by a rule-of-thumb method (Silverman, 1986, Equation 3.31).

A problem with kernel estimates is that, in their straightforward form, they do not respect bounds on the range of the data values. Most evaluation metric scores take values in the range $[0, 1]$; yet the estimated distribution in Figure 3.19 allocates positive density beyond these limits. An obvious solution to the problem is to truncate the density at the boundaries. But then data values near the boundaries lose some of their density contribution. The problem is significant with evaluation metric scores, since there is generally a cluster of them that are close to the 0 boundary.

Several methods of *boundary correction* for kernel density estimates have been proposed. The more complex of them involve applying transformations to the data set (Marron and Ruppert, 1994), or employing special adaptive kernels at the boundaries (Jones, 1993). We employ the simpler *reflection method* (Silverman, 1986; Cline and Hart, 1991). Under this method, density falling beyond a boundary is reflected back at the boundary into the valid range. Equivalently, if the lower bound is a (here, 0), and the upper bound is b (here, 1), then the dataset is augmented by reflection in each boundary, to the form $X^* = \{X_1, 2a - X_1, 2b - X_1, \ldots, X_n, 2a - X_n, 2b - X_n\}$, and only then is the density truncated at the bounds.

Figure 3.20 shows the unbounded kernel density estimate (dashed line) of the topic

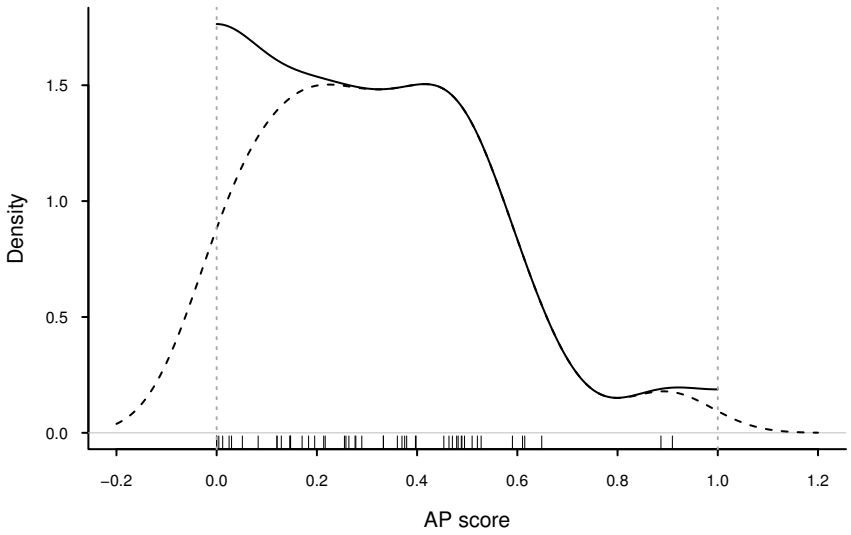


Figure 3.20: Density estimate over the per-topic AP scores of the system `Flab8atdn` for all fifty of the TREC 8 AdHoc collection topics. The dashed lines show the unbounded density estimate; the solid line shows the density estimate bounded to the $[0, 1]$ range, with density outside this range reflected back at the boundaries.

scores for the system previously examined, along with the bounded estimate derived using the reflection method (solid line). If simple truncation were used, the distribution would fall off at either end. This fall would be particularly misleading at the lower bound, since a number of scores cluster near this limit. The bounded density under the reflection method correctly accounts for this cluster of low-scoring topics.

In density estimation graphs such as Figure 3.20, and more generally in graphs of the density of continuous probability function such as Figure 3.15, the probability of a random value falling within a certain range is given by the area under the curve for that range, with the total area under the curve being 1. The y axis (marked "Density" in Figure 3.20) provides the scale for calculating the area under the curve. So, for instance, in Figure 3.20, the density for AP values between 0.2 and 0.4 is roughly 1.5; therefore, we can estimate that (under the density estimate, if not necessarily among the original scores themselves) $(0.4 - 0.2) * 1.5 = 0.3$ of AP values fall within this range.

3.4 Test collection construction

Building a test collection involves creating each of its components: a corpus of documents; a set of topics; and assessments of document-query relevance. The chief issue in corpus creation is that of coherence between documents, while in topic set formation, there are conflicting imperatives for using real queries on the one hand, yet having queriers also act as assessors on the other. But the main difficulties are met in producing the relevance assessments. Exhaustive assessment is infeasible, but incomplete assessment is subject to inaccuracy and bias. Methods for selecting documents for relevance assessment, and for dealing with incompleteness in the set of assessments, have

attracted considerable research interest in recent years. These issues of test collection construction are discussed next.

3.4.1 Corpus, queries, qrels

The document corpus is, from a technical perspective, the most straightforward component to collect. Documents may be sourced from a third party, say from the archives of a newswire service, or else crawled from the web. The chief technical question is of corpus coherence. If a corpus is being created from a subset of the web, for instance, then maintaining a realistic link structure requires some care; it would not be adequate to sample documents at random (Bailey et al., 2003; Soboroff, 2002). Issues of coherence aside, large corpora can be cheaply created, and corpus size is the least constrained of the test collection dimensions.

In creating the topic set, there are two conflicting requirements, which are hard to satisfy simultaneously. On the one hand, topics should be a random sample of real queries; on the other, documents should be judged for relevance by the person who issued the query. Queries can be sampled from a query log, but then assessors must attempt to recreate the underlying information need. Conversely, assessors can write the topics, but then queries are no longer a true random sample. Having the experimental subjects manually create topics risks artificiality in the queries, but sampling queries from a log ensures artificiality in the relevance assessments. In general, TREC has chosen the former method, whereas operational systems appear to rely on the latter (Harman, 2005a; (Pseudo-)Google, 2007).

The most challenging of a test collection's components to create, however, is the relevance assessments or qrels. For one thing, a document's relevance to a topic is subjective, especially when not assessed by the query formulator. Even amongst professional assessors of the same background, two-way inter-assessor relevance set overlap of under 50% has been reported (Voorhees, 2000). The greater assessment challenge, though, is the expense. Documents can be crawled, and queries sampled, but relevance judgments must be performed by humans. Assessing every document for relevance to each query is not feasible for a corpus of more than trivial size. But leaving documents unassessed threatens the reusability of the collection; if those documents are returned by retrieval systems, how are the systems to be evaluated? Qrel incompleteness has been a recurrent issue in information retrieval, and one that has attracted strong research interest recently.

3.4.2 Pooling, incompleteness, and bias

Only a tiny proportion a corpus will be relevant to any one query. For instance, there are over $500{,}000$ documents in the TREC 8 AdHoc corpus, but only 200 or so are estimated to be relevant to the average query (Voorhees and Harman, 1999).[4] One line of thought holds that, if assessment effort can be focused on the documents most likely to be relevant, then a good proportion of the relevant documents should be identified.

The standard method of focusing assessment effort is *pooling* (Spärck Jones and van Rijsbergen, 1975). A set of systems is run against the corpus for each topic, and the top d documents of each ranking are pooled for assessment (Harman, 2005a). The assumption is that a system placing a document at a high rank is evidence in favour of the document's

[4] The average number of judged relevant documents is 95, and Zobel (1998) estimates that around 50% of relevant documents are located by pooling.

relevance; and if the pooled systems are numerous and diverse enough, and pooling is deep enough, then the relevant documents should be substantially covered by the pool (Cormack et al., 1998).

Pooling, by design, leaves the majority of documents in the corpus unassessed. For instance, pooling 71 systems to pool depth 100 at the TREC 8 AdHoc TREC led to an average pool size of 1,736, or around a third of a percent of the full corpus. Unpooled systems therefore can readily return unassessed documents at ranks above the rank that pooled systems were pooled to. The standard approach is to assume that an unpooled, unassessed document is irrelevant. Not all relevant documents will be identified through pooling, though, making the relevance assessments *incomplete* (Buckley and Voorhees, 2004). Since an unassessed document may be relevant, and since only unpooled systems can return unassessed documents by pool depth, qrel incompleteness leads to a *pooling bias* in favour of pooled systems, and of unpooled systems that are similar to pooled ones (Zobel, 1998).

The reality of incompleteness and the potential for bias were first examined systematically by Zobel (1998). Zobel estimates that, for the early TREC AdHoc collections, at best 50% to 70% of relevant documents were found by pooling. Nevertheless, by removing individual systems from the pool and observing the change in their scores (known as a *leave one system out* experiment[5]), Zobel concludes that pooling bias is minimal. Voorhees and Harman (1999) confirmed this finding, using the more stringent test of removing all systems submitted by the one group (a *leave one team out* experiment). As a result of these studies, pooling bias was not held to be a serious issue for the TREC AdHoc collections (Voorhees and Harman, 1999; Harman, 2005a). These experiments, however, only cover the set of systems actually participating in each TREC. It is still possible that an innovative system would find many unique, actually relevant but formally unassessed, documents, and suffer a heavy pooling bias. Indeed, in the TREC 2005 Robust track, a pooled system that used an unusual pseudo-routing approach was found to suffer a 23% drop in score when removed from the pool (Buckley et al., 2007).

Corpus size has continued to grow over the years. The recent ClueWeb09 corpus, for instance, contains over a billion documents (Callan and Hoy, 2009). It is reasonable to assume that as collection size increases, the number of documents relevant to each query will also increase, though perhaps not linearly. A particular concern is that the pools may become filled with easy to find documents rich in query keywords, disadvantaging new systems that attempt to go beyond keyword matching (Buckley et al., 2007). At the same time, even pooling is an expensive process, and it would be attractive to find a more efficient solution, particularly as that would allow an increase in the topic set size, frequently felt to be inadequate (Carterette et al., 2008).

Several approaches have recently been proposed for resolving pooling bias, or replacing the pooling method altogether. Some methods retain pooling while attempting to reduce pooling bias; these are discussed in Section 3.4.3. Another group of methods, described in Section 3.4.4, attempt to increase the proportion of relevant documents identified. Score estimation methods based on random sampling have also been proposed; we examine these in Section 3.4.5. Section 3.4.6 describes another approach, which chooses documents to maximize score deltas, based on a probability of relevance model. There have also been proposals for pool-based evaluation without relevance assessments, for instance by randomly marking pooled documents as relevant (Soboroff et al., 2001) or scoring documents based on how many different systems re-

[5] *System ablation* would be a more concise term, but it has not come into standard usage.

turn them (Wu and Crestani, 2003); such methods, however, are chiefly curiosities, rewarding systems that are similar to others in the pool, rather than ones that are intrinsically of better quality, and serving to underline the dangers of pooling bias and (in the case of Soboroff et al. (2001)) of metrics that evaluate beyond pooling depth.

3.4.3 Metric adjustment for qrel incompleteness

One approach to qrel incompleteness and the retrieval of unassessed documents is to adjust the evaluation metric. Buckley and Voorhees (2004) propose the Bpref metric, which is calculated only over assessed documents; unassessed documents are simply ignored. To test their metric experimentally, Buckley and Voorhees create qrel sets with different degrees of incompleteness, by random sampling from a pooled qrel set. They demonstrate that Bpref is more robust to this form of incompleteness than existing metrics: both absolute scores and system rankings are more stable as incompleteness increases.

Rather than use a special-purpose metric, Yilmaz and Aslam (2006) propose the use of average precision with unassessed documents removed from the ranking, a variant they term *induced AP*. Induced AP and Bpref differ only in the latter using a less top-weighted normalization of document pair contributions. Sakai (2007b) suggests that rankings purged of unassessed documents, which he calls "condensed lists", can be used with any metric. Again using randomly-sampled qrels, Sakai demonstrates that nDCG on condensed lists is more stable than Bpref.

Both Yilmaz and Aslam (2006) and Sakai (2007b) follow Buckley and Voorhees (2004) in creating incomplete experimental qrel sets by random sampling. But such sampling is unbiased, whereas many of the causes of incompleteness, such as pooling, are not. Sakai (2008) re-examines condensed list metrics under partial pooling, and finds them biased in favour of unpooled systems. Unpooled documents are less likely to be relevant than pooled ones; removing them from the ranking of an unpooled system allows lower-ranked, pooled documents to take their place, to the system's favour.

In Chapter 6, we propose a more principled solution to qrel incompleteness than either assuming unassessed documents to be irrelevant, or removing them from the ranking. Our approach is to directly estimate the degree of pooling bias in a qrel set via a leave one out experiment, either on the pooled systems, or on a subset of topics for which the unpooled system is fully assessed, and adjust the scores of the unpooled system accordingly.

3.4.4 Relevance-greedy and strategic selection

Pooling is a simple method for focusing assessment effort on documents that are likely to be relevant. Other methods have been proposed for achieving an even higher proportion of relevant documents amongst those assessed. The goal is to gain maximum coverage of the relevance set with minimum effort; during evaluation, unassessed documents are still treated as irrelevant.

Cormack et al. (1998) propose a dynamic, relevance-greedy method of choosing documents for assessment, called move-to-front (MTF) pooling. The submitted runs are held in a priority queue, with the next document for assessment being the highest-ranked, as-yet-unselected document of the highest priority run. If the selected document is relevant, then that run's priority is set to the maximum value; otherwise, the run's priority is decreased. Assessment effort is therefore focused on runs that return

relevant documents more frequently. Cormack et al. find that MTF pooling locates relevant documents at a rate over 50% higher than regular pooling, and as a result achieves a stable system ordering with fewer assessments.

A more sophisticated algorithm is proposed by Aslam et al. (2003). They treat document selection as an aggregation of expert opinion problem, with each retrieval system as an expert. The next document to assess is the one that gets the strongest combined recommendation from the systems, where the strength of each system's recommendation is derived from the rank it returns a document, weighted by an estimate of the system's reliability. A system's estimated reliability increases when it recommends relevant documents, and decreases when it recommends irrelevant ones. Aslam et al. demonstrate that their method finds relevant documents at a rate roughly 50% higher than plain pooling; a direct comparison to the MTF method of Cormack et al. (1998) is not given.

The interactive searching and judging (ISJ) method proposed by Cormack et al. (1998) does away with submitted runs altogether. Instead, the assessor or assessors actively seek out relevance judgments, through query reformulation performed on a live search system. This process was used to form a manual TREC 6 AdHoc run, which subsequent analysis showed to have located 59% of the officially relevant pooled documents, though the ISJ assessors only regarded 71% of these as actually relevant. The proportion of the documents located through ISJ which were relevant was 30%, five times the density of the 6% for pooling. Controlling for assessor disagreement, the ISJ qrels gave a Kendall's τ of 0.96 with the official qrels on system ranking, indicating virtually identical rankings.

Having the assessor search for documents through explicit query reformulation is arguably a confusion of roles, one liable to a narrow and biased interpretation of relevance. An alternative is to use *true relevance feedback*, where the query is automatically and iteratively extended, based on the relevance assessments made by the assessor. The qrels for the 2003 Filtering track of TREC were created using iterative true relevance feedback (Robertson and Soboroff, 2002; Soboroff and Robertson, 2003). To provide diversity, four retrieval systems were employed, with their rankings merged, and the top hundred presented for assessment. The assessments were then fed back into the relevance feedback system, and a new merged ranking produced and assessed. The loop was repeated until no new relevant documents were found, or five iterations had elapsed. An average of 433 documents were judged per topic, with 82 (19%) found relevant; in comparison, the TREC 8 AdHoc pool had an average of 1,763 document per topic, with 94 (5%) relevant. An additional round of pooled judging found more than fifty new relevant documents for only seven of the fifty topics. Sanderson and Joho (2004) use only a single feedback system in the loop, with similar results.

Moffat, Webber, and Zobel (2007) present methods for a range of strategic goals, besides maximizing the relevance proportion. These methods are specific to the RBP metric, and are built upon its monotonically decreasing error residual, described in Section 3.2.3. Algorithms are provided for achieving three different strategic goals: minimizing the mean residual across systems; equalizing residuals between systems; and dynamically weighting score residuals to achieve greater fidelity in separating high-performing systems.

The relevance-greedy methods identify relevant documents more efficiently than pooling, but they leave unaddressed the question of pooling bias, and potentially exacerbate it; the aggregation of expert opinion method of Aslam et al. (2003), for instance, seems likely to bias assessments in favour of clusters of similar systems. Additionally,

no matter how efficient the methods are, budget constraints mean that only a fraction of relevant documents in very large contemporary collections can be assessed. The sampling methods described next address the issue of bias, and require only a fraction of the runs to be assessed.

3.4.5 Random sampling

In relevance assessment, we are faced with trying to determine a value (relevance) over a large set of items, while only having the resources to examine a small subset of the items. The standard approach to such problems is *sampling*, and it is natural that sampling has been applied to score estimation in retrieval evaluation, too.

Yilmaz and Aslam (2006) propose the uniform random sampling of documents from runs, up to evaluation depth. Calculating scores upon the assessed sample is based on the key insight that an evaluation metric can be modelled as the expectation of a random variable. For instance, precision at ten can be modelled as the expectation of picking one of the top ten documents at random and assessing it for relevance. Average precision is more complex, because it is pairs of documents, not individual documents, that are scored; therefore, the sampling also has to be over pairs. But purely paired sampling is inefficient, because pairings induced between documents from different sampled pairs are ignored (sampling ij and kl induces ik and jl). In practice, the method is to sample documents individually, assess them for relevance, and then treat the induced pairs as if they had been produced by a pairwise sampling method.

The uniform random sampling approach can be made more efficient. Intuitively, greater sampling probability should be given to documents with more weight in the metric, such as documents highly ranked by multiple systems. One also (perhaps less intuitively) wants to give greater sampling probability to documents that are more likely to be relevant. Such an approach is proposed by Aslam et al. (2006). Again, the authors tackle the difficult case of estimating average precision, deploying an arsenal of techniques for minimizing variance under unequal sampling (Thompson, 2002, Chapter 6). Average precision weights pairs of ranks, so unequal sampling must be performed on rank pairs. Sampling weights are combined from the full set of systems, with values rescaled to fit each run's distribution. A distribution over pairs is derived from marginal distributions over single documents, to avoid losing induced document pairs. A prior over probability of relevance is applied, itself based on average precision weights (Aslam et al., 2003). Finally, marginals for each ranking are averaged to provide a marginal sampling distribution over all documents.

The complexity of the unequal sampling method discourages its use in practice, and makes it difficult to apply where hybrid document selection methods are used or more than one evaluation metric is employed. Instead, Yilmaz, Kanoulas, and Aslam (2008b) propose a simpler method based on stratified sampling (Thompson, 2002, Chapter 11). Different proportions are sampled from different strata of the document rankings, using uniform random sampling within each stratum, and adjusting sampled values to give an unbiased estimator. The stratified model supports in particular pooling to a certain depth, and sampling beyond that depth. Again, adjustment of values is complicated by the pairing of documents under average precision, and it is not clear that Yilmaz et al. deal in a theoretically unbiased way with pairs of assessed documents induced in one ranking because one or other of the pair have been pooled in another ranking. Nevertheless, the resulting sampling method is flexible and straightforward.

Sampling requires that the full population be available for sampling from; in retrieval evaluation, that means all the runs that are to be evaluated. New systems can be

scored as if the assessed documents they return had been randomly sampled from their runs (Yilmaz, Kanoulas, and Aslam, 2008b); the purported sampling, however, has not in fact occurred, suggesting that the new system will still be biased against. The degree of bias has not been determined; it seems plausible that the bias of sampling from a pool is similar to the bias of full assessment of the pool.

3.4.6 Probabilistic delta determination

The sampling approach aims to estimate an absolute score. Frequently, though, the experimenter is more concerned with determining whether one system outscores another; in other words, has a positive delta compared to it. Documents can be selected to maximize this delta. Assessment could continue until a positive delta was certain; but these methods become much more efficient if joined with a probability of relevance model, to calculate confidence in a positive delta.

The maximization of delta confidence is proposed by Carterette et al. (2006) in their minimal test collection (MTC) method (see also Carterette and Allan (2005) for an earlier, more heuristic approach). Different documents have different impacts on score deltas between two rankings. If a document is returned at the same rank by both rankings, then, under a rank-weighted metric such as RBP, its relevance can have no impact on their delta. The situation under AP is, again, complicated by its scoring of document pairs, but a similar logic applies. The first step of the MTC method is to select the documents that have the greatest impact on score delta, and a deterministic implementation of MTC would continue until it is certain that one system is better than the other. For non-convergent metrics like AP, this involves the assumption that evaluation is only carried out to a certain depth, as otherwise the tail always has infinite potential weight.

As evaluation continues towards the point of certainty, it becomes increasingly likely that one system is better than the other, and one would like to be able to cut short the assessment once this likelihood had reached a certain level. To support the calculation of this likelihood, Carterette et al. (2006) develop a model of the probability of relevance for individual documents, from which they derive an estimator of the delta, and a variance on that estimator. Their initial model is very simple: each unassessed document has an independent 0.5 probability of relevance. Carterette et al. demonstrate that, given this (unrealistic) probability assumption, AP is normally distributed, allowing a confidence interval and achieved significance level to be calculated on the delta. Thus, assessment need only continue until sufficient confidence is achieved. Carterette et al. find that, given the above assumptions, a 95% confidence in a system ranking (estimated as the average of the pairwise confidences) can be achieved with around a tenth of the number of relevance assessments that a depth 100 pooling would require.

A more realistic probability of relevance model is fitted to the MTC framework in Carterette (2007). The new model is one of an aggregation of expert opinion, with the retrieval systems as experts, and probabilities of relevance for each unassessed documents the output. A three-level logistic regression is performed. The first infers the probability of relevance that a system is stating by returning a document at a given rank; the second calibrates the stated probability of relevance by the system's observed reliability; and the third calculates an aggregated probability of relevance, weighted by system correlations. The next document for assessment at each step is chosen according to the delta-maximization principle, and the result of the assessment is used to update the model. Carterette asserts that the updated method is robust for use on new

systems, without additional assessment effort; the new runs can be fitted into the same probability of relevance model, and an appropriate (non-zero) probability of relevance assigned to their unassessed documents. Experimental evaluation demonstrates this robustness, at least on the experimental data employed.

The updated MTC model and an early variant of the sampling method of Yilmaz, Kanoulas, and Aslam (2008b), called statMAP, were employed in the 2007 Million Query track of TREC (Allan et al., 2007; Carterette et al., 2008). The two methods gave rankings similar to each other, and to that produced by pooled assessments on older topics. The absolute estimated MAP scores produced by MTC, though, were only a third of those for statMAP or pooling. It seems that MTC is getting its absolute predictions of probability of relevance wrong, even though the relative values may be correct. This suggests a possible bias if one system in a pair happened to be fully assessed, and so required no estimation. On the other hand, for the one system that MTC differed markedly from statMAP and pooling on, a system which used the distinctive method of query expansion via an external search engine (Webber, Anh, and Moffat, 2007a), additional relevance assessments indicated that MTC was in the right, suggesting that it may be more robust than the other methods.

3.5 Materials

The key experimental data for our investigations is provided by the TREC effort. An historical overview of TREC has been given in Chapter 2. Here, we discuss the TREC tracks, collections, and particularly runs in more detail.

3.5.1 The TREC effort

The TREC effort began in 1992, and is still running at the time of writing. The naming convention for the annual efforts is inconsistent. From 1992 to 2000, each year's TREC is known by its number, TREC 1 through to TREC 9; then, starting in 2001, TRECs started to be commonly referred to by their date, TREC 2001, TREC 2002, and so forth; or, more briefly, TREC-01, TREC-02, and so on. While this convention is liable to confuse, it is so deeply ingrained that we shall follow it here.

There are two parts to TREC: the annual conference itself, and the collaborative retrieval experiments that lead up to it. The conference is held in November of each year. Towards the beginning of the year, around March, the test corpus (if new) is made available to participants. Topics are released around July, and runs are submitted by participants a few weeks later. Relevance assessments are then performed on the pooled runs, with results released in late September. Each year's task involves a new collection: either the topics or the corpus, or both, change. In a few cases, though, where the corpus is the same, participants are asked to run their systems over previous topic sets as well (Voorhees, 2004; Clarke et al., 2005). These instances are particularly valuable for meta-evaluation: they create larger combined topic sets, as well as repeated runs over earlier topic sets.

Proceedings, topics, qrels, and tools are freely available from the TREC website, `http://trec.nist.gov/`. Runsets from participating groups are also available from TREC, subject to a data usage agreement. The test corpora used in TREC collections are typically distributed by third parties, but information on where to obtain them is contained on the TREC site.

The success of the TREC effort has inspired a number of other collaborative retrieval experiments. The NTCIR project (http://research.nii.ac.jp/ntcir/) began in 1999, initially as a Japanese-language task but spreading to include other East Asian languages and cross-lingual retrieval. The Cross Language Evaluation Forum (CLEF, http://www.clef-campaign.org/) was founded in 2000, with a special emphasis on multi-lingual, cross-lingual, and mono-lingual retrieval of European languages. The Initiative for the Evaluation of XML Retrieval (INEX, http://inex.is.informatik.uni-duisburg.
launched in 2002, to work on the retrieval of semi-structured text. In 2008, the Forum for Information Retrieval Evaluation (FIRE, http://www.isical.ac.in/~fire/index.html) was initiated, with a focus on South Asian languages. In this thesis, though, we use the TREC data exclusively.

3.5.2 TREC tracks and collections

The TREC effort began with two related tasks: the AdHoc and Routing tasks (Harman, 1992b). In the AdHoc task[6], a topic is provided, and the system has to find documents relevant to that topic (Harman, 2005b). The Routing task involves a form of batch document filtering: the system is provided with relevance judgments over an existing corpus, and must identify similarly relevant documents in a new corpus (Robertson and Callan, 2005). Starting with TREC 4 in 1995, additional *tracks* were added to the TREC program, each dealing with a different retrieval environment or problem. By TREC 2008, 28 different tracks had been run at TREC at one time or another (Voorhees, 2007; Buckley and Robertson, 2008), although at no one TREC were there more than nine tracks running concurrently.

The motivation for the different tracks beyond AdHoc are various. They can be roughly categorized as:

- Different corpus types and sizes: Very Large Collection, Web, and Terabyte (Hawking and Craswell, 2005; Clarke et al., 2004).
- Different retrieval modes: Routing, Filtering, Interactive, Question Answering (Robertson and Callan, 2005; Dumais and Belkin, 2005; Voorhees, 2005c)
- Different natural languages: Spanish, Chinese, Cross-Lingual (Harman, 2005c)
- Different media: Confusion (OCR data), Spoken Document, Video (Voorhees and Garofolo, 2005; Smeaton et al., 2001)
- Specialized domains: Genomics, Blog, Enterprise, Legal (Hersh and Bhupatiraju, 2003; Ounis et al., 2006; Soboroff et al., 2006; Baron et al., 2006)
- Special retrieval aspects: Novelty, Robust, High Precision, Query, Million Query (Harman, 2002; Voorhees, 2003; Buckley, 1997, 1998; Allan et al., 2007)

The retrieval type focused on in this thesis is ad hoc retrieval; that is, retrieval of topically relevant text documents from a broad-domain corpus in response to one-off user queries. The chief TREC tracks dealing with ad hoc retrieval are the AdHoc track itself and the Robust track, which used the same corpus and many of the same topics; and, to a lesser extent, the Web and Terabyte tracks. We describe these tracks next.

[6]The naming convention at TREC changed slightly. Up to TREC 4, the task was referred to as the "adhoc" task, without a space; this is even footnoted in the overview to TREC 4 as the received TREC spelling. From TREC 5 onwards, however, it is referred to as the "ad hoc" task, with a space. In this thesis, we will refer to the task itself as the "AdHoc task", and to the type of retrieval the task (and, say, the Robust track after it) involved as "ad hoc retrieval".

AdHoc track

The AdHoc track ran from the first TREC 1 in 1992, to TREC 8 in 1999; it was then discontinued, as it was believed that retrieval effectiveness on the task had plateaued (Voorhees and Harman, 1999). The initial data corpus for the AdHoc track was the 2GB TIPSTER corpus (Harman, 1992b), known as TREC Disks 1 and 2. The corpus was supplemented in later years with a further three disks of material, with old disks being retired as new ones were added. From TREC 6, the AdHoc corpus settled down to what are known as TREC Disks 4 and 5; the same corpus was used not only in TREC AdHoc 6, 7 and 8, but also the TREC Robust track in 2003 and 2004. The AdHoc corpora consist primarily of newswire data, with some other document types included in earlier years (Harman, 2005a).

A new set of 50 topics was formed for each year's AdHoc collection. The AdHoc topics are numbered sequentially, starting with Topics 51–100 for TREC 1 (Topics 1–50 were used for pre-TREC trials and training, and also in the Routing track in the first year). The nature of the topics changed over time. Topics 51–150, from TRECs 1 and 2, are highly detailed, containing concept categories, term definitions, and lists of topic factors. The topics were also designed to return a large number of relevant documents, at least 25 based on an initial sample run (Harman, 1992b). It came to be felt that these topics specified too much information and were not demanding enough, and over time there was a conscious effort to make topics harder, in part by reducing the amount of information contained in the topic statements. Topics 201–250 from TREC 4 were reduced to single-sentence descriptions (Harman, 1995; Spärck Jones, 2000). By TREC 5, the format for AdHoc topics had settled down to title, description, and narrative, as described below in Section 3.5.3.

Robust track

The next track of interest to the research described in this thesis is the Robust track, run at TREC from 2003 to 2005. The purpose of the track was to increase the consistency of system performance, by focusing on topics that the systems performed poorly on (Voorhees, 2003). The track promoted work on query difficulty measurement and prediction, and introduced the use of the geometric mean in score aggregation, in order to emphasize the contribution of low-scoring topics.

The main attraction of the Robust track for meta-evaluation is its re-use of corpus and topics from the AdHoc track. The TREC 2003 and TREC 2004 Robust collections use the same document corpus as the TREC 6 through 8 AdHoc collections, namely TREC Disks 4 and 5 (minus the Congressional Record sub-corpus). Furthermore, the TREC 2004 Robust topic set includes the TREC 6 through 8 AdHoc and TREC 2003 Robust topics, in addition to its own new topics. This comes to a total of 249 topics on the one document corpus, making it the largest fully-pooled topic set amongst TREC collections. Care has to be taken, though, because the TREC 6 topic descriptions are unusual in lacking title keywords and hence providing lower retrieval performance, and also the Robust pools include a narrower range of automatic runs than the AdHoc pools, and (aside from 2005) no manual runs at all. Nevertheless, the reuse of earlier topic sets on the same corpus means that TREC 2004 Robust runs can be directly compared to earlier TREC runs. These features enable a wealth of meta-evaluative experiments.

Web track

The Web track ran from TREC 8 in 1999 through to TREC 2004 (Hawking and Craswell, 2005). Initially, the task was to run ad hoc style queries over web data (Hawking et al., 1999). Topics statements followed the AdHoc form, although title-only queries were emphasized as the only realistic queries for a web environment (Hawking, 2000). The topical relevance model of ad hoc retrieval, though, came to be seen as unrepresentative of much web search behaviour (Hawking and Craswell, 2005). In response, starting in TREC 2001, the web-centric tasks of home page and named-page finding were introduced (Hawking and Craswell, 2001; Craswell and Hawking, 2002). For these tasks, only a single page is relevant; since this page is known in advance to the collection creators, the relevance assessment of submitted results is unnecessary. In TREC 2002, ad hoc retrieval was replaced by the topic distillation task, which was to find pages that were not merely relevant, but which served as key resources or entries into relevant sites (Craswell and Hawking, 2002; Craswell et al., 2003). Thus, concepts of document quality and location in the link graph were introduced. The track continued in this form until its cessation in 2004. It has recently been revived, with a much larger collection, in 2009 (Callan and Hoy, 2009).

Terabyte track

The final track considered here is the Terabyte track, which ran from TREC 2004 to TREC 2006. The corpus was a 426 GB crawl of the US government web domain. The core task was topical relevance ad hoc retrieval; named-page and efficiency tasks were also run.

One of the track's central goals was to explore whether TREC-style evaluation methods scale to terabyte-size collections; in particular, whether pooling is reliable on such large collections (Clarke et al., 2004). Leave-one-team-out experiments (Section 3.4.2) on the 2004 runs discovered an average drop of 9.6% in AP scores; for 2005, it was 3.9%, but one system still suffered a 17.7% fall (Clarke et al., 2005). The frequency of query keywords in relevant documents was found to be significantly higher than for the AdHoc and even Web collections, suggesting an increasingly narrow and biased pool (Clarke et al., 2005; Buckley et al., 2007). In response, the selection of documents for assessment in the 2006 track was performed by a mixture of standard pooling to depth 50, late pooling from depth 400 onwards, and random sampling. Manual runs were also strongly encouraged, to increase pool diversity. Issues of incompleteness, bias, and assessment efficiency were taken up the following year, on the same corpus but with very different methods, as part of the new Million Query track (Allan et al., 2007; Carterette et al., 2008).

The scale of the Terabyte collection, its focus on evaluation issues, and its inclusion (unlike much of the Web track) of an ad hoc, topical relevance task, make it superficially attractive for our investigations. But the very fact that it is, almost by design, highly incomplete, makes it less suitable for meta-evaluative studies, as establishing the ground truth of full evaluation is more problematic. For these reasons, and because of the particular attractions of the combined late-AdHoc and Robust collections, the AdHoc and Robust datasets form the backbone of our experimental structure here. Unless otherwise noted, it is the features of these collections that are discussed in the following sections.

```
<top>

<num> Number: 408
<title> tropical storms

<desc> Description:
What tropical storms (hurricanes and typhoons) have
caused significant property damage and loss of life?

<narr> Narrative:
The date of the storm, the area affected, and the extent
of damage/casualties are all of interest. Documents that
describe the damage caused by a tropical storm as
"slight", "limited", or "small" are not relevant.

</top>
```

Figure 3.21: Topic statement from the TREC 8 AdHoc test collection.

3.5.3 TREC topics

The topics in the TREC AdHoc collections (from TREC-3 onwards) were formulated by the same NIST assessors who subsequently performed the relevance assessments for those topics. Assessors were asked to come up with a range of information needs based on their own interests; these were then explored in the test collection, using a NIST search engine, to ensure that each topic had some, but not too many, relevant documents (Harman, 2005b).

Up to TREC 5, the fields of the AdHoc topic varied from year to year. From TREC 5 onwards, they were standardized to the form illustrated in Figure 3.21. Aside from the topic number, each topic has three fields: title, description, and narrative. In earlier TRECs, the title field was simply a title. From TREC 6 onwards, the title also served as a potential, keyword-based query. The description is a single-sentence statement of the information need underlying the topic; the narrative expands upon this information need, and specifies criteria for a document to be judged relevant or irrelevant.

Submitted runs are divided into *automatic* runs, made without human involvement in query formulation, and *manual* runs, made with such involvement. Queries for automatic runs had to be extracted from one or more of the topic fields, and automatic runs are sub-classified by the topic fields used. Common choices are title-only; description-only; title and description; and title, description, and narrative. No training on test topics that involved human assessments was permitted (NIST, 1999).

Manual runs, on the other hand, did permit human involvement. Initially, this was restricted to manually-written queries. However, from TREC 5 onwards, no limits were placed on the degree of human interaction in manual runs (Voorhees and Harman, 1996; Harman, 2005b). Query re-formulation based on iterative runs was a common technique; and in at least one case, a manual run was constructed by privately assessing retrieved documents for relevance, and hand-crafting the submitted rankings accordingly (Cormack et al., 1998).

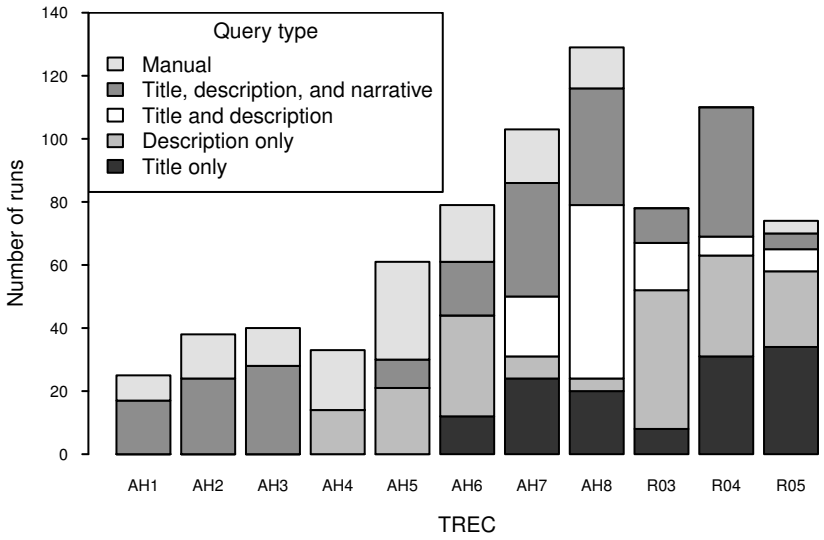


Figure 3.22: Composition of official TREC runs by query type for TREC 1 through TREC 8 AdHoc track, and TREC 2003 through TREC 2005 Robust track.

3.5.4 TREC runsets

The key part of each year's TREC is the collaborative experiment. Guidelines differ for different tracks; here, we describe those that applied to the AdHoc and Robust tracks. Each participating research group was required to submit one or (typically) more runs against that year's test collection. The maximum number of automatic runs a group could submit varied from 2 to 5, except for the TREC 2004 Robust track, in which no limit was set, with one group submitting 11 runs. A run submitted to TREC for evaluation contains one document ranking for each topic in the topic set. Document rankings were generally made to depth $1{,}000$. In the TREC 6 Robust track, each group was required to submit at least one description-only automatic run, if they submitted any automatic runs at all (Voorhees and Harman, 1997); this requirement was also enforced in the TREC 2003 Robust track (Voorhees, 2003). In TREC 8, the requirement was for a title and description run (Voorhees and Harman, 1999), whereas the TREC 2004 Robust track required both a title-only, and a title and description run (Voorhees, 2004). Apart from that, groups were not constrained in what runs to submit; commonly, though, the different runs from the one group were based on modified versions of the one system. In addition to automatic runs, manual runs were encouraged.

Importantly, the submitted TREC runs have been archived by NIST (apart from TREC 1), and made available to researchers. The availability of this dataset has been crucial in spurring research on evaluation over the past decade, and the dataset is core to the experiments reported in this thesis.

Figure 3.22 shows the number and type of runs submitted to each TREC AdHoc and Robust track. In some years, participants were permitted to run on a subset of the corpus; these runs have been excluded. The variety in the composition of run types is evident. The concept of using different topic fields for automatic queries was introduced in TREC 5, and then expanded in following years. A significant number of manual runs were submitted to each of the AdHoc tracks, but none for the first two Robust tracks and only a handful for the third. Title-only runs began in TREC 6, and

make up a minority of runs until the TREC 2005 Robust track, even though from the current perspective these might seem the most natural form of user queries.

3.5.5 TREC qrels

As emphasized in Section 3.4, creating the relevance assessments is the crucial task of test collection construction. The method used for the TREC AdHoc and Robust collections was pooling, described in Section 3.4.2. Pooling was performed to depth 100 for all tracks except for TREC 3 AdHoc (200), the new topics for TREC 2003 Robust (125), and TREC 2005 Robust (55, plus assessments from the parallel HARD track) (Harman, 1994; Voorhees, 2003, 2005b). A certain number of runs from each group, selected by the group itself, formed the pool. In the TREC 2003 and 2004 Robust tracks, assessment pools were only formed for the new topics; the old topics reused the relevance assessments from the original collections.

One of the useful contributions of manual runs is the wide range of unique relevant documents they uncover. In the TREC 7 AdHoc track, for instance, 24% of relevant documents were included in the pool solely because they were returned by the 17 manual runs, compared to just 9% solely by the 86 automatic runs. That these figures show automatic runs to be almost redundant for pool formation, given a reasonable set of manual runs, has not been lost on those seeking more efficient modes of pool creation, as we have seen (Cormack et al., 1998; Sanderson and Joho, 2004). Conversely, where manual runs were not submitted, as for the TREC 2003 and 2004 Robust runsets (see Figure 3.22), the depth and diversity of the set of pooled (and therefore assessed) documents is open to question.

Relevance assessment for the TREC AdHoc and Robust collections was performed by NIST assessors, who are predominantly retired intelligence analysts. For the AdHoc collections, relevance assessment was binary; that is, documents were assessed as either relevant or irrelevant to a topic. The threshold for relevance was low: a document was judged relevant if it contained any information that could be included in a report on the topic (Harman, 2005a). For the TREC Robust collections (as well as for the Web and Terabyte collections), a three-level relevance assessment scale was used: irrelevant, relevant, and highly relevant (Voorhees, 2003; Hawking, 2000; Clarke et al., 2004). These judgments are frequently converted to binary values, by treating both relevant and highly relevant documents as (binary) relevant; this convention will be followed in the experiments described later in this thesis.

Figure 3.23 displays the average number of documents per topic assessed relevant and irrelevant in the TREC AdHoc and Robust test collections. The way the qrels were formed varied in some years, affecting both the absolute number of judgments and the proportion of documents assessed relevant. For instance, in TREC 3, pooling was done to depth 200, rather than 100 (Harman, 1994); in TREC 5, a high proportion of pooled systems ran on only a fraction of the corpus (so-called "Category B" runs), pooling a disproportionate number of mostly irrelevant documents from this fraction (Voorhees and Harman, 1996); while in TREC 2005, pooling was only to depth 55, but also included runs from the quasi-feedback HARD track, leading to a higher proportion of relevant documents (Voorhees, 2005b). Nevertheless, two important trends can be noted. The first is the decrease in the number of relevant documents over the first six years of TREC, the result of a deliberate policy to make the topics more difficult. And the second is the much more superficial pools of the TREC 2003 and TREC 2004 Robust tracks, particularly in the number of relevant documents, caused a smaller mix of participating groups, and especially by an absence of manual runs.

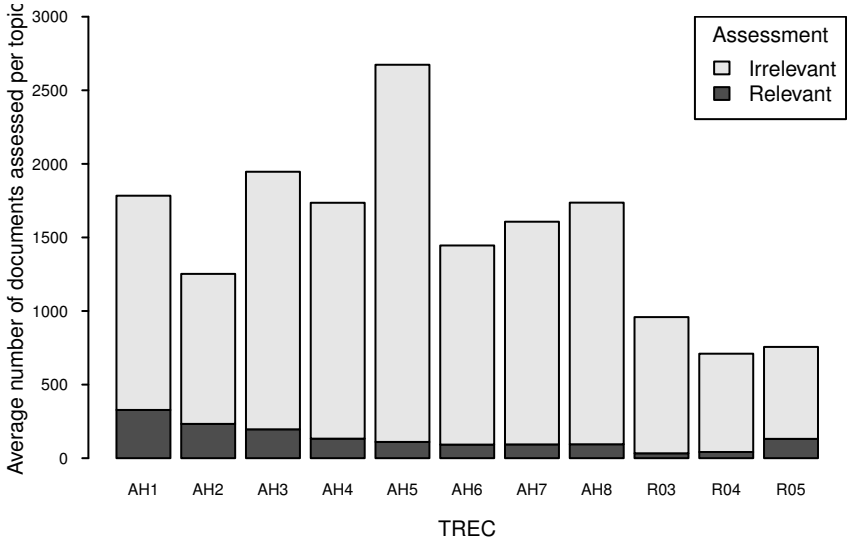


Figure 3.23: Mean documents assessed relevant and irrelevant per topic for TREC 1 through TREC 8 AdHoc track, and TREC 2003 through TREC 2005 Robust track.

The variability in the number of relevant documents per topic within each collection is far greater than the variation between collections. This variability is plotted in Figure 3.24 for the TREC 8 AdHoc Track collection; the figures for other collections are similar. The mean number of relevant documents per topic for this collection is 95, but the range is from 6 up to 347. This variation makes it difficult to meaningfully compare scores between different topics. We explore this issue in more depth in Chapter 4, where we propose the use of score standardization.

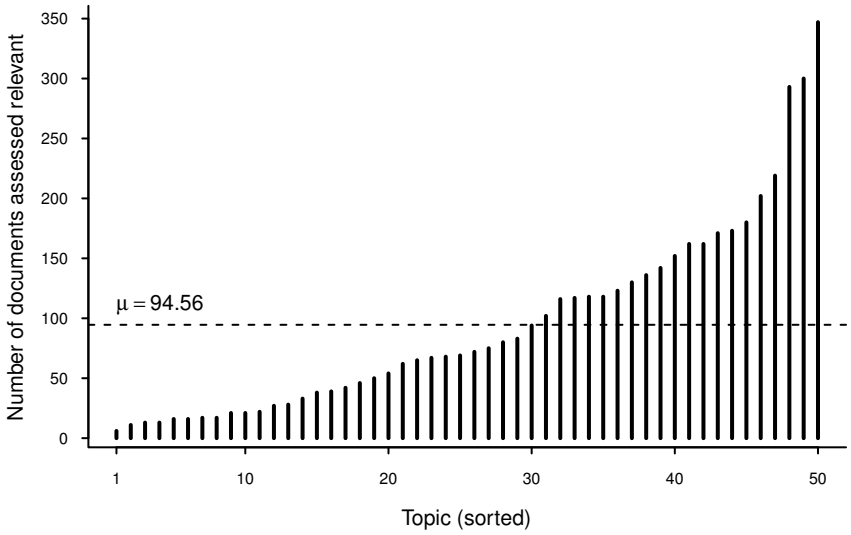


Figure 3.24: Documents assessed relevant per topic in the TREC 8 AdHoc collection.

3.6 Thesis plan

Chapter 2 provided an historical overview of information retrieval evaluation, describing its origins in the Cranfield experiments, the shift towards user studies in the 1970s and 1980s, and the emphatic re-instatement of the system-centric methodology that came with TREC. In this chapter, the technical foundation has been provided. We have summarized the model underlying topical relevance evaluation using test collections, and the methodology built upon this model (Section 3.1). Several metrics for evaluating retrieval effectiveness have been introduced, along with criteria for comparing them (Section 3.2). Statistical methods, essential for summarizing, analyzing, and verifying evaluation results, have been described, with a particular emphasis on tests of statistical significance (Section 3.3). The main issue in the construction of test collections is how to perform relevance assessment in an efficient and unbiased way, and recent work on the effect on assessment incompleteness and on alternative methods of assessment selection has been summarized (Section 3.4). Finally, the TREC test collections and runsets, which form core test data for the thesis, have been introduced (Section 3.5).

The historical and technical backgrounds having been filled in, the scene is set for the presentation of the original contributions of the thesis. The variability in topic difficulty has been observed; this variability renders score comparisons unreliable and inexact, not only between topics, but also between collections. In Chapter 4, we propose *score standardization* as a direct approach to balancing variability in topic difficulty. Under standardization, the mean and standard deviation of the scores of a set of reference systems are observed for each topic. Future scores on that topic, and the scores of the reference systems themselves, are then standardized by subtracting that mean and dividing by that standard deviation. Between-topic variability is eliminated for the reference systems, and greatly reduced for new systems, allowing scores to be directly compared between different topics and different collections.

We have underlined the importance of testing the statistical significance of evaluation results. In planning an experiment, a researcher wants to be confident that a practically important improvement in effectiveness is found to be statistically significant; otherwise, not only is an experiment wasted, but a valuable idea might be neglected. The main factor under the experimenter's control to boost significance is the number of topics; but adding extra topics increases the expense of assessment. A similar question is whether existing test collections have enough topics in them to reliably detect meaningful differences. Questions of this nature are addressed using the statistical tool of *power analysis*. In Chapter 5, we introduce the use of power analysis in deciding the necessary topic set size for an evaluation experiment, describe some of the difficulties involved in deploying this tool in the retrieval evaluation setting, and propose pragmatic solutions to these difficulties. We also use power analysis to ask how sensitive existing collections are to detecting differences in performance, concluding that the standard TREC topic set size of 50 is sufficient only for detecting quite large differences in retrieval effectiveness.

The problem of how to deal with assessment set incompleteness is a particularly important one, given the increasing scale of test corpora and the need to find more efficient ways of targetting relevance assessment effort. The traditional approach of assuming unassessed documents to be irrelevant is biased against unpooled systems, but the recently proposed alternative of removing unassessed documents from rankings is biased in unpooled systems' favour. In Chapter 6, we propose that the degree of bias against an unpooled system be directly estimated, and scores adjusted accordingly. Estimation can be done via a leave one out experiment on the fully-pooled systems, as-

suming that the new system is similar in nature to the pooled ones. Or alternatively, it can be performed by pooling the new and existing systems on a small set of common queries. Our proposed solution is particularly suitable for dynamic evaluation environments, deployed over a regularly increasing set of test queries.

So far, we have only considered the comparison of document rankings by their effectiveness scores. There are, however, many scenarios in which document rankings can or must be compared without reference to relevance or effectiveness. Such a comparison could be made as a cheap proxy for a much more expensive evaluation of effectiveness; but it can also be made for its own reasons, for instance where a search provider wants to monitor its rivals, or even itself, to measure the degree of change in rankings over time. A number of similarity measures between rankings exist, such as Kendall's τ, but they do not deal with the particular needs of document ranking comparison: specifically, top-weightedness, nonconjointness, and incompleteness. We define rankings having these qualities as *indefinite rankings*, and in Chapter 7 we propose a new (and, we argue, the first suitable) indefinite rank similarity measure, called *rank-biased overlap* (RBO). We demonstrate the usefulness of RBO in comparing document rankings; indefinite rankings, though, are quite widely found, and we suggest that RBO has a wider field of application than information retrieval alone.

We have seen in Chapter 2 that information retrieval evaluation has had a long history. The last two decades of this history have been dominated by the TREC effort, and have seen a period of standardization and dissemination of TREC-style evaluation methods. Such standardized methods and materials facilitate the improvement of information retrieval technology. They also allow us to ask whether information retrieval technology has in fact been improving over the past decade or more, at least as embodied in public research. This is the question we tackle in Chapter 8, first by looking at the retrieval scores achieved at TREC over time, and then at the scores published subsequent to TREC on TREC collections. We find that, in the AdHoc and subsequent Robust Track, improvement in results appears to have plateaued quite early, perhaps as early as TREC 3 in 1994. Similarly, in examining published results, we find few systems outperforming the original TREC runs, and no evidence of an upwards trend over time. These are interesting and concerning findings, and we examine a number of possible explanations for them.

That the TREC methodology can be turned to critically analyze its own use by the research community is a sign not of its weakness, but its strength, and indeed of the rigour of the information retrieval community. The purpose of this thesis is to refine the methods and extend the applicability of the TREC methodology: to give evaluation scores some independence from collections and pairwise comparisons via standardization; to encourage individual research groups to develop new, special-purpose test collections by providing the tools of power analysis as a guide in experimental design; to make the TREC method more flexible in dynamic evaluation environments through score adjustment for the compensation of evaluation bias; to add the similarity comparison of document rankings as an adjunct tool to effectiveness evaluation; and to emphasize the importance not just of rigorously evaluating retrieval effectiveness in individual experiments, but tracking the effectiveness of retrieval technology over time. Certainly, the topical-relevance test collection model needs to be extended and supplemented to cope with the richer, more interactive, more complex world of search and exploration on the web. But the enormous success of the TREC model, and the prevalence of the methods it has popularized, suggests that extended or alternative evaluation methodologies will not gain traction in the research community unless they offer similar traits of automation, replicability, and rigour.

Chapter 4

Score Standardization

Test collections provide a means for measuring the effectiveness of retrieval systems. Effectiveness is quantified, using an evaluation metric, as a score for each topic, and then a mean score for the collection as a whole. These topic and mean scores by themselves convey little information about system performance, however. They are only informative when used to compare one system with another. The reason for this lack of absolute meaning is that topics scores are highly variable, with some topics receiving mean scores (across a set of systems) that are ten times that of others. In other words, some topics are easy to achieve high scores on under a given metric, others hard. The score that a system achieves on a collection depends, then, on how hard the topics included in that collection are; and scores achieved on different collections cannot usefully be compared. Additionally, the dispersion of scores differs between topics, making some much more discriminative than others in practice, without it being clear that they should be more discriminative in principle.

We have seen in Chapter 3 that some metrics, such as AP and nDCG, normalize scores based on the number of relevant documents for each topic, thus making perfect scores at least theoretically obtainable. Normalization can be seen as a method of addressing the variability in topic difficulty. It is, however, one that has limited success, as will be seen. In this chapter, we present a more direct solution, that of *score standardization*. The idea of standardization is to infer topic difficulty, not from the number of relevant documents for the topic, but from the observed scores achieved on the topic by a set of *reference systems*. Specifically, the mean and standard deviation of the reference scores on the topic are taken as *standardization factors*. Scores for that topic, both of the reference and of other systems, have the mean subtracted from them, and are then divided by the standard deviation, to derive a *standardized score*.

Standardization greatly reduces topic variability. The standardized scores for different topics have by construction the same mean and standard deviation for the reference systems, and much diminished variance for non-reference systems, too, provided the reference set is sufficiently representative. As a result, standardized scores carry more information about system effectiveness than unstandardized ones, since the reference points of average and exceptional performance are fixed. Moreover, standardization substantially increases comparability between different collections. It is even possible to perform a significance test between the scores that one system achieves on one collection and another system achieves on a different collection.

This chapter is laid out as follows. We begin in Section 4.1 by introducing the tools of components of variance analysis, which are useful for analyzing and summarizing

the variability of systems, topics, and interactions between them. In Section 4.2, we examine the nature and extent of inter-topic score variability, and the problems it causes for the interpretation and comparison of effectiveness scores. Section 4.3 introduces score standardization, and compares it with metric normalization, as described in Chapter 3. In Section 4.4, we look at the simplest case, self-standardization, in which the set of systems whose scores are standardized is the same set that the information used to calculate the standardization factors is drawn from. Section 4.5 extends the analysis to the case where the standardized and standardizing systems are different, while Section 4.6 examines standardization's claim to permit score comparison between different collections. Finally, the chapter concludes in Section 4.7 by exploring methods for reducing the effect of outlier scores and the dependence of standardized scores on the reference set.

4.1 Measuring score variability

A collaborative evaluation experiment such as TREC involves running a set of systems against the one collection. The set of participant systems, $\mathcal{S}$, is run against each topic in the collection's topic set, $\mathcal{T}$. The document ranking produced by system $s \in \mathcal{S}$ on topic $t \in \mathcal{T}$ is then scored by an evaluation metric, such as average precision, to produce the *system–topic score*, X_{st}. If there are $S = |\mathcal{S}|$ participating systems, and $T = |\mathcal{T}|$ topics, then the system–topic scores from the collaborative experiment can be regarded as an $S \times T$ matrix. (One could also think of the k-depth rankings as forming a three-dimensional array of $S \times T \times k$, which the evaluation metric reduces to two dimensions.) Let $\mathbf{M}$ denote this score matrix. The system–topic score X_{st} that system s achieves on topic t is the element $\mathbf{M}_{s,t}$, and the collection score $\overline{X}_s$ for system s is the mean of its per-topic scores against each topic, $\overline{\mathbf{M}}_{s,*}$.

The system–topic score X_{st} that system s achieves on topic t can be represented by the following linear model:

$$X_{st} = \mu + \nu_s + \nu_t + \nu_{st} \ . \tag{4.1}$$

The value μ (read "mu") is the mean across all systems and topics. Each ν (read "nu") is an effect: ν_s the system effect for system s, ν_t the topic effect for topic t, and ν_{st} the system–topic interaction effect for system s on topic t. A positive ν_s indicates that system s is strong, relative to other systems; a positive ν_t indicates that topic t is easy, relative to other topics; and a positive ν_{st} indicates that the performance of system s on topic t is high compared to other systems on this topic, or this system on other topics, or both. Negative effects have the reverse meaning.

The system, topic, and system–topic interaction components each have a variance, $\sigma^2(s)$, $\sigma^2(t)$, and $\sigma^2(st)$. (Note that s and t refer here not to a specific system or topic, but to the variance over systems and over topics.) These variances sum to the total variance of the system–topic scores in the runset:

$$\sigma^2(X_{st}) = \sigma^2(s) + \sigma^2(t) + \sigma^2(st) \ . \tag{4.2}$$

The variances can be calculated from the mean squares of an ANOVA computation; see Brennan (2001, Chapter 2) for details. We refer to this as a *components of variance analysis*. Informally, $\sigma^2(s)$ expresses how much the systems differ from each other in performance, $\sigma^2(t)$ how much the topics differ from each other in difficulty, and $\sigma^2(st)$ how consistently topics score systems; a $\sigma^2(st)$ of 0 would, for instance, mean that every topic ranks the systems in the same order.

We can think of $\sigma^2(s)$ as the signal of the evaluation, separating out systems by their true performance, whereas $\sigma^2(t)$ and $\sigma^2(st)$ are the noise, confusing the analysis of absolute performance by topic-related effects. From these, we define two measures of the reliability of topic scores. The first of these is the *index of absolute comparability*, φ (read "phi"):

$$\varphi = \frac{\sigma^2(s)}{\sigma^2(s) + \sigma^2(t) + \sigma^2(st)} \ . \tag{4.3}$$

The index φ measures the comparability of absolute scores; that is, the information conveyed by the score alone, and the ability to compare scores between topics and collections. The second measure is the *index of relative comparability*, ρ_I (read "rho");

$$\rho_I = \frac{\sigma^2(s)}{\sigma^2(s) + \sigma^2(st)} \ . \tag{4.4}$$

The index ρ_I measures how comparable scores are on individual topics and within the one collection. It does not include the topic variance component, $\sigma^2(t)$. When comparing systems on the same topics, it is the score deltas that matter, and the score deltas are not affected by changes in mean topic scores.

The indexes φ and ρ_I are similar to the measures Φ and $\mathbf{E}\rho^2$ from generalizability theory (Brennan, 2001; Bodoff and Li, 2007). The difference is that the former two are expressed per topic, the latter two over a certain number of topics. Increasing the number of topics proportionally dampens the topic and topic–system interaction effects. For instance, Φ, the *index of dependability* from generalizability theory, is defined as follows:

$$\begin{aligned} \Phi &= \frac{\sigma^2(s)}{\sigma^2(s) + \sigma^2(T) + \sigma^2(sT)} \\ &= \frac{\sigma^2(s)}{\sigma^2(s) + \left(\sigma^2(t) + \sigma^2(st)\right)/n_t} \end{aligned} \tag{4.5}$$

where the capital T indicates that the variance is amortized over a number of topics, and n_t is the number of these topics. In this chapter, we work with the per-topic versions defined above, to avoid the dependency on the number of topics.

4.2 Topic variability

Section 4.1 introduced summary statistics for analyzing the different components of variance in system–topic scores. From the point of view of deriving reliable, absolute scores, the most undesirable of these components is topic variance; it adds to score variability but conveys no information about system performance. Topic variance—the variance in mean topic scores—is, however, only one aspect of topic score variability. There is also variability in the dispersion of scores, as measured by standard deviation: for some topics, scores tend to cluster together, whereas for others, they are more spread out. Topic scores also differ in the shape of their distributions: some topics give balanced distributions, others give skewed ones. In the current section, we examine the symptoms of topic variability (Section 4.2.1), and describe its consequences (Section 4.2.2).

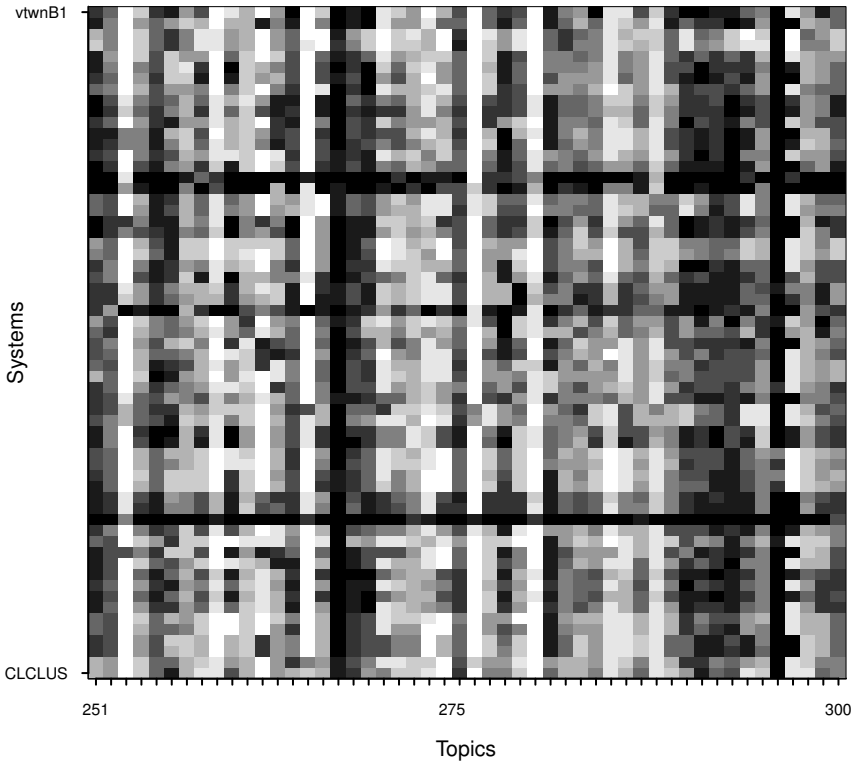
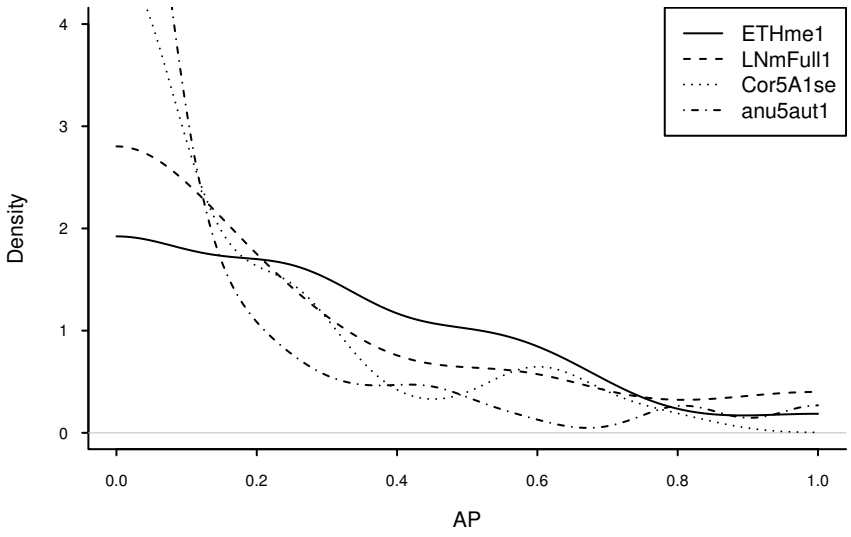


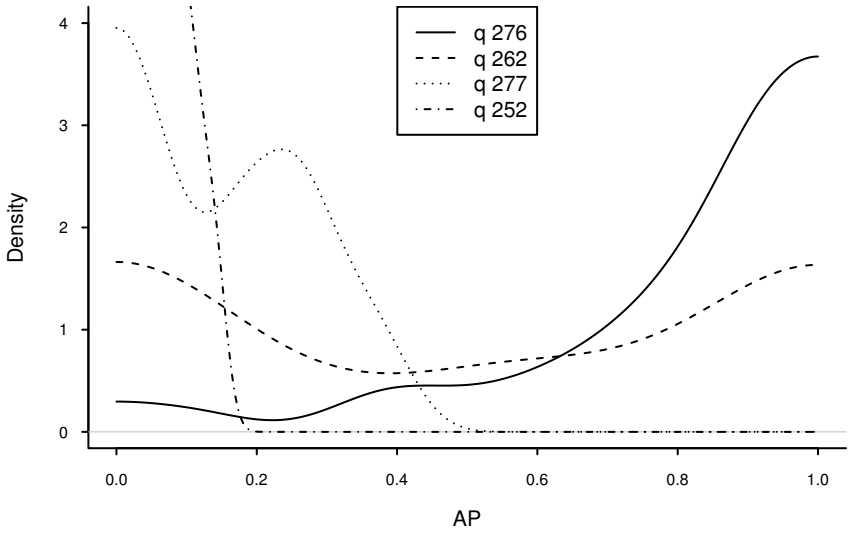
Figure 4.1: Intensity visualization of unstandardized AP scores from the TREC 5 Adhoc Track. The columns represent topics, ordered by topic number, and the rows represent systems, in ASCII order of system name, from `CLCLUS` at the bottom to `vtwnB1` at the top. Each cell represents the AP system–topic score of a document ranking. Lighter shades indicate higher scores. An equal number of scores are assigned to each colour, such that white represents AP scores from 1 and 0.57, lightest grey from 0.57 to 0.38, and so forth, with black indicating scores below 0.0015. Although failed systems are obvious, easy topics (light vertical stripes), such as Topic 253, stand out more clearly than good systems (light horizontal stripes).

4.2.1 The incidence of topic variability

Visualizations provide an intuitive, high-level view of the patterns in a dataset. Figure 4.1 visualizes the AP scores achieved on the TREC 5 Adhoc Track topics by systems participating in that year's track. The scores are represented as an $S \times T$ matrix, in which topics are columns, systems are rows, and each cell represents a system–topic score, with lighter cells indicating higher scores. The easy topics, those against which systems generally achieve high scores, stand out strongly in the visualization, producing vertical white lines. But good systems do not produce bright horizontal lines, making them difficult to discern. Only the very worst systems, those that fail to find any relevant documents for almost every topic, stand out as dark horizontal lines. This simple visualization illustrates an important point: a system–topic evaluation score holds more information about the difficulty of the topic than it does about the quality of the system.



(a) System AP score density



(b) Topic AP score density

Figure 4.2: Kernel density estimates of AP scores for the (a) systems, and (b) topics, at the first, tenth, twentieth, and seventy-fifth percentile (top to bottom in the legend) when ordered by mean (a) system, or (b) topic, AP scores. Systems behave more like each other than do topics. All data is from the the TREC 5 Adhoc Track.

The scores that system s achieves against the collection's topic set constitute the row $\mathbf{M}_{s,*}$ of the matrix visualized in Figure 4.1, and the scores that are achieved by the set of systems against a topic t make up the column $\mathbf{M}_{*,t}$. Each such row or column can be treated as a distribution of scores. These distributions are displayed in Figure 4.2 for the first, tenth, twentieth, and seventy-fifth percentile topic and system from the TREC 5 Adhoc Track, as ordered by topic or system mean AP score. The system

	System AP			
	ETHme1	LNmFull1	Cor5A1se	anu5aut1
mean	0.317	0.282	0.206	0.154
st.dev	0.231	0.271	0.220	0.232

	Topic AP			
	q276	q262	q277	q252
mean	0.771	0.506	0.175	0.056
st.dev	0.235	0.383	0.118	0.039

Table 4.1: Mean and standard deviation of AP scores for sample systems and topics from the TREC 5 Adhoc Track. The kernel density estimates for the same data are plotted in Figure 4.2.

Metric	TREC 5 AdH				TREC 01 Web				TREC 06 TB			
	s,φ	t	st	ρ_I	s,φ	t	st	ρ_I	s,φ	t	st	ρ_I
AP	8	62	30	22	12	46	42	23	19	49	32	37
P@10	12	47	41	22	12	48	39	23	19	41	40	32
RBP.95	11	55	34	24	10	60	30	25	19	51	30	39
nDCG	16	55	29	36	18	44	38	33	30	41	29	51

Table 4.2: Variance components and comparability measures as percentages (%) for different metrics and TREC runsets. The components are: s, system; t, topic; st, system–topic interaction. The measures are the indexes of absolute (φ), and relative (ρ_I) comparability. Higher scores for these indexes mean greater comparability. The percentages of the first three columns in each row of each block add up to 100, rounding effects aside.

score distributions have a similar, right-skewed unimodal shape, similar dispersions, and even relatively similar locations. The topic scores, in contrast, differ from each other in shape, dispersion, and location.

Table 4.1 summarizes the locality and dispersions of the system and topic distributions plotted in Figure 4.2, in terms of their means and standard deviations. All systems have similar standard deviations, and the mean of the best system is only twice that of the seventy-fifth percentile. In contrast, topic means and standard deviations each vary tenfold from seventy-fifth percentile to highest value. Thus, topic scores show much greater diversity than system scores. The standardization transformation that we are about to describe will fix dispersion and location; shape, though, will remain variable.

The variance components and reliability measures (Section 4.1) are displayed as proportions in Table 4.2 for a number of different TREC runsets and metrics. For AP, P@10, and RBP with $p = 0.95$, the proportion of score variance attributable to actual differences in system performance is mostly around 10%, and always below 20%. The nDCG metric performs noticeably better, but its maximum system effect is 30%. The proportion of variance due purely to topic effects is always above 40%, and reaches as high as 60%. That is, around half the variance in individual system–topic

Systems	Significance test	
	Paired	Two-sample
All	0.636	0.364
Auto	0.495	0.130

Table 4.3: Proportion of system pairs from the TREC 5 Adhoc Track found to have significantly different average precision at $\alpha = 0.05$ in a two-tailed t test, either paired or two-sample, and including either all 61 systems or only the 28 automatic systems scoring more than 0.05.

scores is attributable purely to differences in topic difficulty, another third or more to peculiar interactions between topics and systems, and a fifth or less to consistent differences in system performance. As a result, the reliability of the absolute scores, as measured by φ, is around half that of the relative scores, as measured by ρ_I. These are, admittedly, figures for individual topics; where scores are averaged across n topics, each of the topic and system–topic variance components will reduce by $1/n$ (refer back to Equation 4.5). Even so, for the typical 50-topic TREC collection, some 10% of the variance in mean system scores is due to factors other than system performance.

4.2.2 The impact of topic variability

The variability in topic difficulty observed in Section 4.2.1 has several effects. One is to make two-sample significance tests much weaker than paired significance tests. A second is that confidence intervals on mean system scores are very wide, indicating the weak information that a mean score provides. These two problems mostly concern comparisons between test collections; a third, though, applies even within test collections, and that is that individual topics have different impacts on mean score deltas. We discuss these issues in turn.

The first effect of high topic variability is that two-sample tests achieve far fewer significant findings than paired tests (Section 3.3). Using a two-sample test, as if the systems were run on distinct topic sets, is strongly subject to topic variance. This variance is controlled in paired tests by taking paired system score deltas. Table 4.3 contrasts the proportion of the $1{,}830$ system pairs, between each of the 61 the TREC 5 Adhoc Track systems, that are found significantly different under paired and two-sample t tests. The paired test finds significance for almost two-thirds of system pairs, but only a little over a third for the two-sample test. If the mostly high-performing manual runs are removed, as well as four faulty systems scoring below 0.05 (two of which are manual, two automatic), then the outcome is even more marked. Half of the 378 pairings between the 28 remaining automatic systems are found significantly different by the paired test, but only one eighth by the two-sample test. These results illustrate the difficulty of comparing scores between collections (even similar ones), where the paired test cannot be employed.

Figure 4.3 displays another effect of high topic variability. Here, 95% confidence intervals have been plotted on the "true" mean system AP scores of the TREC 5 Adhoc Track systems. These confidence intervals are related to (though not identical with) the results of a two-sample significance test. The intervals on the scores are wide, leading to high degrees of overlap. The lower bound of the first system overlaps with the upper bound of the median (thirty-first) system, and the mean score of the median

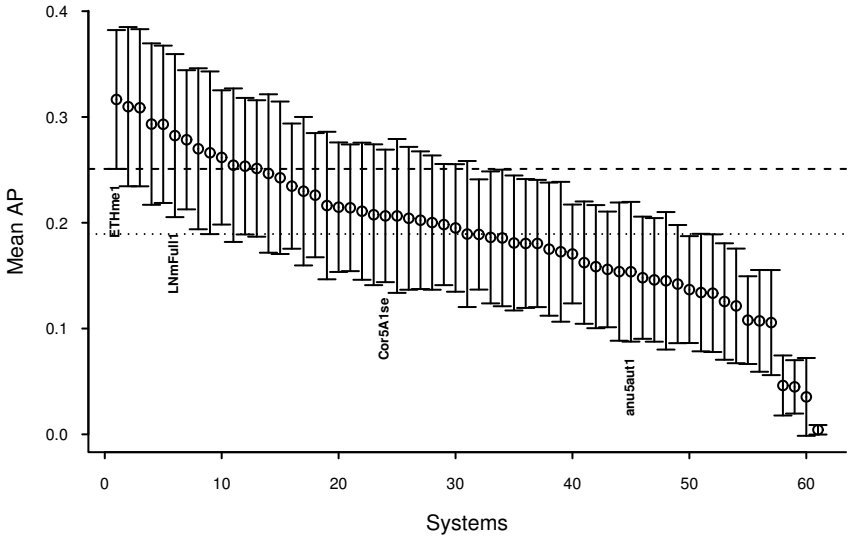


Figure 4.3: The 95% confidence intervals on mean AP scores for the TREC 5 Adhoc Track systems, using a t distribution. Systems are ordered by their mean AP score. The first, tenth, twentieth, and seventy-fifth percentile systems (previously reported in Figure 4.2) are labelled. The lower bound of the first system and the mean of the median system are marked with horizontal lines.

system overlaps with the bounds of the ninth at one end, and the fifty-first at the other. One way of interpreting these values is as the range of mean scores the systems could achieve on equivalently sampled collections. The width of these intervals underlines the weak information that a system score, quoted in isolation, conveys. A mediocre automatic system on an easy collection—that is, a collection that (by chance) consists of easy topics—can outscore a top-class manual run on a hard collection.

Even for evaluation on the one test collection, topic variability causes distortions. Consider Table 4.1 again. The standard deviation of system AP scores for Topic q262 is ten times that for q252 (0.383 to 0.039). Topic q262 therefore has on average ten times the impact on the mean score delta between any two systems as Topic q252. It happens that Topic q262 has the highest standard deviation in TREC 5; but even the twenty-fifth percentile topic by deviation has two and a half times the standard deviation of the seventy-fifth percentile topic (0.170 against 0.067). It might be supposed that, for whatever reason, higher variance topics are more discriminating than lower variance topics. But the *item–total correlation* (that is, how reliably an item score reflects overall performance) (Bodoff and Li, 2007) between individual and aggregate topic scores is not significantly greater for high-variance topics than for low ones. The Pearson's correlation between item–total correlation and standard deviation under AP for TREC 5 is 0.277 for all runs, and 0.091 for non-faulty automatic runs; neither correlation achieves significance at level $\alpha = 0.05$. Although high variance topics have greater influence on system mean AP scores and on paired significance tests, they are not inherently more reliable. Their greater influence is neither deserved nor desirable.

4.3 Score standardization

One approach to controlling variability in topic difficulty has already been presented in Section 3.2.4, namely score *normalization*. Under normalization, a system's topic score is divided by the topic's maximum score, derived from R, the number of relevant documents for the topic. Let r_t be the maximum score achievable for a topic, and X_{st} the unnormalized score that a run achieves on that topic; then the normalized score X^*_{st} for that run is:

$$X^*_{st} = \frac{X_{st}}{r_t} \tag{4.6}$$

Average precision is an implicitly normalized metric, normalized discounted cumulative gain an explicitly normalized one. Table 4.2 indicates that normalization does a poor job of reducing topic variability for AP, and a better but still incomplete job for nDCG.

A more direct method for controlling topic variability derives from the observation that, since the problem is that different topics have different score means and standard deviations (as in Table 4.1), then the solution is to adjust topic scores so that means and standard deviations are the same for each topic. If a topic t produces a score mean of $\mu_t = \overline{\mathbf{M}}_{*,t}$ and a score standard deviation of $\sigma_t = \mathrm{sd}(\mathbf{M}_{*,t}) = (1/S)\sum_s \sqrt{(\mathbf{M}_{s,t} - \mu_t)^2}$, and if a run s receives the unadjusted score of X_{st} for topic t, then the adjusted score X'_{st} for that run is:

$$X'_{st} = \frac{X_{st} - \mu_t}{\sigma_t} \tag{4.7}$$

Such a value is known as a z *score*, and the process of deriving it is called *standardization* (Hays, 1991, Chapter 4). (In the unusual case that all reference systems receive the same score, so that σ_t is 0, then we set X'_{st} to 0, too.)

Standardization factors must be calculated across a set of *reference systems* that are run and evaluated against a collection. For instance, the systems participating in a TREC track might form the reference set for that track's collection. It is important to distinguish between standardization's effect on reference and on non-reference systems. Standardization directly controls topic variability for reference systems, taken as a set; it reduces the topic variability of non-reference systems, provided this variability is similar to that of the reference systems. These questions are examined experimentally in Section 4.5.

Standardization relies on there being an original retrieval experiment, whereas normalization could theoretically be performed in the absence of such an experiment, if documents for relevance assessment were chosen some other way. In fact, though, pooling experimental systems is the predominant method of forming qrel sets. Therefore, the requirements for standardization are in practice no greater than for normalization. How many experimental systems are required to derive reliable standardization factors, and how durable these factors remain over time, are considered in the Section 4.5; the same questions also apply to normalization, as is observed in Section 4.6.

Standardization and normalization are alternative methods of controlling topic score variability; either can be applied to most raw retrieval metrics. Standardization of normalized scores, though, is identical to standardization alone, as we proceed to prove:

Proposition 4.1 *Standardization of a normalized score gives the same numerical value as standardization of a raw score.*

Proof. As shown in Equation 4.6, normalization involves dividing each raw score X_{st} achieved on a topic by the maximum score r_t achievable for that topic. Then, from Equation 4.7, the standardized normalized score is:

$$X_{st}^{*\prime} = \frac{(X_{st}/r_t) - \sum_{i=1}^{S}(X_{it}/Sr_t)}{\sqrt{\frac{1}{S}\sum_{j=1}^{S}\left((X_{jt}/r_t) - \sum_{i=1}^{S}(X_{it}/Sr_t)\right)^2}} = X'_{st} \ . \tag{4.8}$$

□

The three choices for a base metric are therefore its raw form (for instance, DCG); its normalized form, denote by the prefix "n-" (for instance, nDCG); and its standardized form, denoted by the prefix "s-" (for instance, sDCG). This means that, since AP is normalized SP (nSP) (as shown in Section 3.2.4), standardized AP or sAP is the same as standardized SP or sSP. Since AP is familiar, and nSP is not, we will adopt the former usage here, making for AP the triplet of SP, AP, and sAP (corresponding, for instance, to DCG, nDCG, and sDCG). Note that sAP can be calculated without needing to know or estimate R, the number of relevant documents.

A standardized distribution has the same shape as the unstandardized one; standardization only affects locality and dispersion. After standardization, the mean score $\overline{\mathbf{M}}'_{*t}$ for each topic across the reference systems is zero, and the standard deviation is one. There are no absolute upper or lower bounds on standardized scores, but Chebyshev's inequality guarantees that at least 75% of standardized scores for a topic will be between -2.0 and 2.0, and in practice the proportion is much higher; for AP on the TREC 5 Adhoc Track, it is 96%, close to the proportion for a normal distribution. The finite size n of the reference set places a practical bound of $\pm\sqrt{n-1}$ on standardized reference scores, occurring when $n-1$ reference systems achieve the same score, and the nth one receives a different one (see Section A.1.1 for a proof). Thus, the impact of any one topic on a reference system's score is constrained. A non-reference system, however, can achieve a score beyond this bound. This issue is examined empirically in Section 4.5, and some methods for controlling outlier scores are suggested in Section 4.7.

Table 4.4 shows the unstandardized and standardized AP scores for the topics and systems used as illustrations in Section 4.2. The unstandardized scores are difficult to interpret and compare. For example, `ETHme1` scored 0.344 for Topic `q277` and 0.500 for Topic `q262`; but does the latter score represent a better result than the former, or was it simply an easier topic? Similarly, `Cor5A1se` scored 0.2 higher than `anu5aut1` on Topic `q277`, but 0.08 lower on Topic `q252`; is the former result more compelling than the latter? In contrast, the standardized results are directly informative. A positive score indicates the run outperformed the reference mean for that topic, a negative score that it underperformed it; a score of 1.0 means the run is one standard deviation above the mean, and so on. So, without examining any other figures, we can immediately see that `anu5aut1` has done well on topic `q252` with its standardized score of 1.340, and `Cor5A1se` poorly on topic `q262` with -1.277.

4.4 Standardizing reference systems

In this section, we examine the effect of standardization on reference systems and within the one collection. The distribution and variance characteristics of standardized scores are investigated first, followed by the effect of standardization on significance tests and confidence intervals. Section 4.5 examines the standardization of

Topic	Unstandardized AP			
	ETHme1	LNmFull1	Cor5A1se	anu5aut1
q276	0.968	1.000	0.615	0.814
q262	0.500	0.950	0.017	1.000
q277	0.344	0.301	0.256	0.059
q252	0.045	0.058	0.030	0.109

Topic	Standardized AP			
	ETHme1	LNmFull1	Cor5A1se	anu5aut1
q276	0.840	0.975	−0.667	0.180
q262	−0.015	1.161	−1.277	1.291
q277	1.434	1.067	0.689	−0.985
q252	−0.275	0.059	−0.665	1.340

Table 4.4: Selected topics from the TREC 5 Adhoc Track, and selected system AP scores, before and after standardization, where the reference set is the full set of TREC 5 systems. The standardization factors μ_t and σ_t for these four topics are the mean and standard deviation values in the bottom section of Table 4.1.

non-reference systems, and Section 4.6 looks at the application of standardization to cross-collection comparisons.

4.4.1 Characteristics of standardized scores

We start by comparing the system rankings produced by unstandardized and standardized scores. Figure 4.4 compares the mean standardized and unstandardized system AP scores for the TREC 5 AdHoc Track. The two metrics correlate closely, with a Pearson's r of 0.985, and a Kendall's τ of 0.903. By way of comparison, the correlation of nDCG with mean (unstandardized) AP is $r = 0.973$ and $\tau = 0.915$. The difference between the standardized and unstandardized variants of a single metric is similar to that between one unstandardized metric and another. Standardization does cause some local perturbations in the ranking, as it equalizes the score variability of different topics. These perturbations may represent improvements, if we accept the thesis that topics should have similar impacts.

Next, we examine the effect that standardization has upon the overall distribution of scores. Figure 4.5 shows the distribution of system–topic unstandardized and standardized AP scores across all TREC 5 AdHoc systems and topics. The raw AP scores, shown in part (a), are heavily skewed towards lower values, with half of the per-run scores being below 0.1. As the raw score intensities in Figure 4.1 and the analysis of topic variance in Section 4.2.1 suggest, the tail of high system–topic scores is caused by a few exceptionally easy queries, not a few exceptionally strong systems. The standardized values, shown in part (b), are clustered around their mean of 0, with a skewed bell-curve shape, inherited from the raw scores: a slight majority of values (57%) are below 0 (the median is -0.2), but there are a few (10 out of $3{,}050$) outlier values above 4. The distribution of standardized scores diverges from normality, but normality of distribution is not a goal of standardization: as mentioned before, the topic score

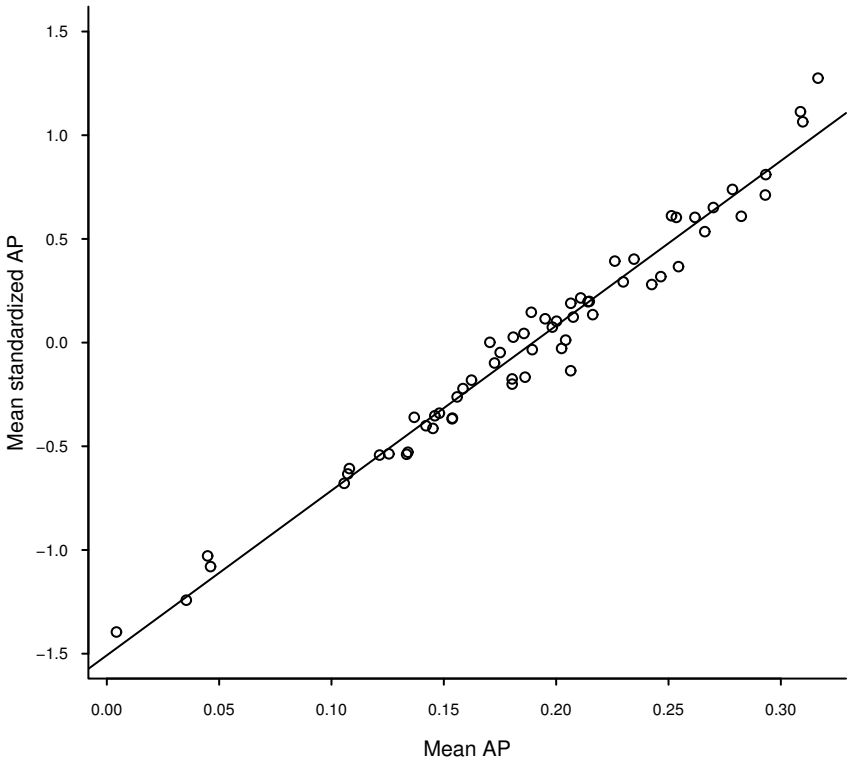
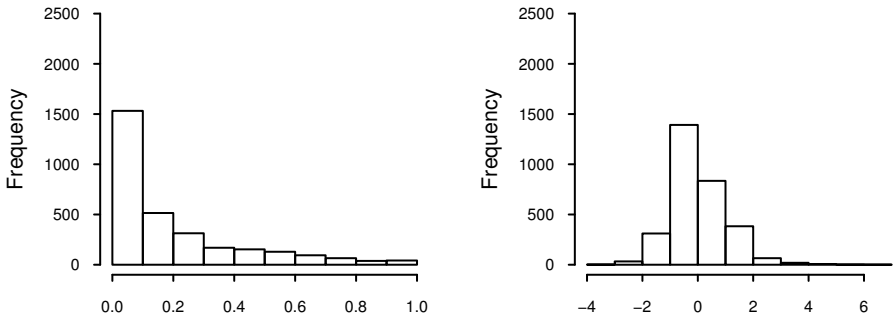


Figure 4.4: Mean unstandardized AP and mean standardized AP scores for the TREC 5 Adhoc Track systems, with the line of best fit, which has slope 7.9 and intercept -1.5. The Kendall's τ correlation is 0.903; Pearson's r is 0.985.

distributions maintain their (mostly skewed) original shapes, changing instead their locality and dispersions. Nevertheless, Figure 4.5 is suggestive of a more balanced distribution of scores after standardization than before.

The distribution of standardized scores across the $S \times T$ matrix of system–topic results is illustrated in Figure 4.6; this should be compared with the distribution of unstandardized scores shown in Figure 4.1. As expected, easy topics have disappeared: there are no vertical white lines. As with unstandardized scores, weak systems still stand out; but now, strong systems are also visible, as predominantly (but not entirely) light horizontal lines. We can easily make the judgment from this illustration that, even in relative terms, no system is good at all topics. Moreover, there is a visible clustering effect amongst systems, resulting from the alphabetical ordering of system IDs, which places runs submitted by the same group next to each other; different families of systems excel at different groups of topics. One question which jumps out of this visualization is how to combine the complementary strengths of the two University of Waterloo systems (third and fourth from the top) with those of the four Lexis-Nexis systems (eighteenth through twenty-first from the bottom) (though admittedly these are all manual runs). This simple visualization illustrates the richness and depth of comparisons that score standardization enables within a set of runs.

We observed in Table 4.2 that the topic variance component made up as much as 60% of the variance of system–topic scores. Table 4.5 reports the variance components



(a) AP (b) sAP

Figure 4.5: Distribution of system–topic (a) AP, and (b) standardized AP, scores for the TREC 5 Adhoc Track systems. The 3,050 scores achieved by the 61 TREC 5 systems against the 50 topics are summarized. The reference set for the standardized scores are the 61 TREC 5 systems themselves.

Metric	TREC 5 AdH			TREC 01 Web			TREC 06 TB		
	s,φ,ρ_I	t	st	s,φ,ρ_I	t	st	s,φ,ρ_I	t	st
sAP	29	0	71	28	0	72	39	0	61
sP@10	22	0	78	25	0	75	32	0	68
sRBP.95	29	0	71	30	0	70	41	0	59
snDCG	38	0	62	35	0	65	54	0	46

Table 4.5: Variance components and reliability measures as percentages (%) for different standardized metrics and TREC runsets. The components are: s, system; t, topic; st, system–topic interaction. The measures are the indexes of absolute (φ) and relative (ρ_I) comparability. The percentages in each row of the three columns of each block add up to 100, rounding effects aside. Compared to Table 4.2, topic variance (t) has been redistributed as system (s) and system–topic interaction (st) variance.

of the TREC 5 scores for four common metrics after standardization. As anticipated, the topic component of variance has been eliminated in all cases. The removed topic component is more or less proportionally shared by the system and system–topic interaction effects. As a result, the index of absolute comparability, φ, which is equivalent to the proportional s component, doubles or even triples; standardization causes a dramatic increase in absolute score dependability. In addition, since the topic variance component has been eliminated, φ is identical to the index of relative comparability ρ_I, meaning that absolute scores are as reliable as relative ones. The effect of standardization on relative score reliability is much smaller; still, the ρ_I scores here are in general slightly higher than, and never below, those for unstandardized scores, given in Table 4.2. This suggests that even within the one collection, relative scores are slightly more reliable (or at least more consistent) under standardization; this question is discussed further in Section 4.7.

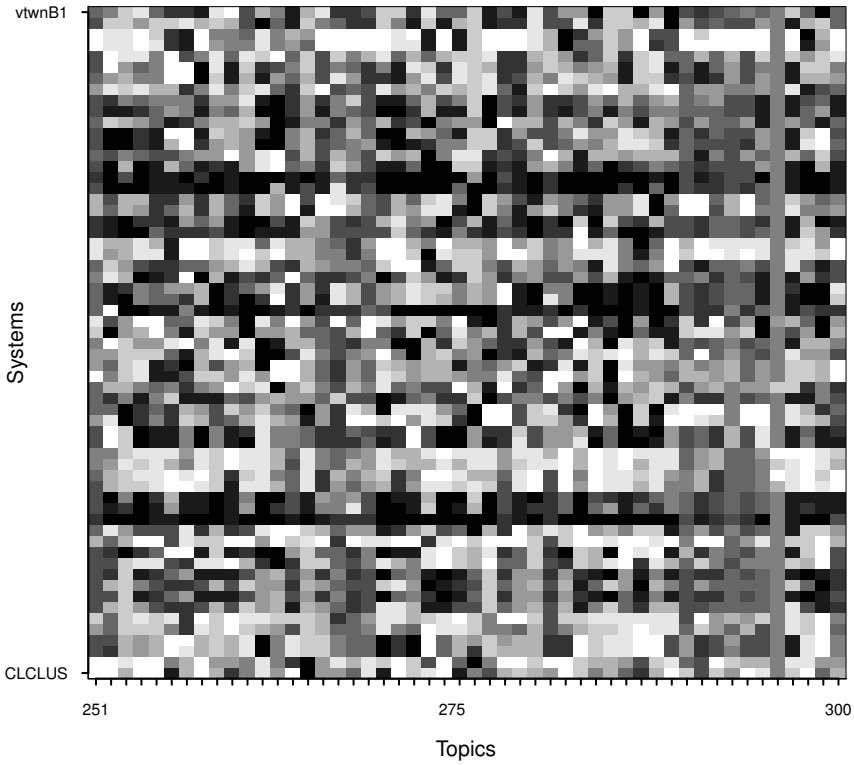


Figure 4.6: Intensity visualization of standardized AP scores from the TREC 5 Adhoc Track. The columns represent topics, ordered by topic number, and the rows represent systems, in ASCII order of system name, from `CLCLUS` at the bottom to `vtwnB1` at the top. Each cell represents the standardized AP (sAP) score of a single run. Lighter shades indicate higher scores. An equal number of scores are assigned to each colour, such that white represents sAP scores above 1.34, lightest grey scores from 0.89 to 1.34, and so forth, with black indicating scores below -1.07. Good systems (light horizontal stripes) stand out more clearly than for the unstandardized scores in Figure 4.1, and there are no easy queries (light vertical stripes).

4.4.2 Standardization, significance, and confidence

We saw previously, in Table 4.3, that the topic variance of unstandardized metrics makes two-sample significance tests much weaker than paired tests, and hence tests between collections much weaker than within collections. In contrast, Table 4.6 shows that standardization, by reducing topic variance, gives the two-sample significance test a discriminative power almost equal to that of the paired test—as it to be expected when $\varphi = \rho_I$ (absolute and relative comparability are the same). The level of agreement between the two is also high; the overlap (number of system pairs that both find significant divided by the number of system pairs that either finds significant) is 0.9 for the full system set, which is acceptable given the $p = 0.05$ significance level. These results show that under standardization, significance tests between collections potentially approach the power of those within the one collection, provided the collections are similar in makeup. (Inter-collection tests are explored directly in Section 4.6.) In

Systems	Significance test	
	Paired	Two-sample
All	0.683 *+0.047*	0.683 *+0.319*
Auto	0.561 *+0.066*	0.542 *+0.412*

Table 4.6: Proportion of system pairs from the TREC 5 Adhoc Track found to be significantly different using standardized AP at $p = 0.05$ in a two-tailed t test, either paired or two-sample, and including either all 61 systems or only the 30 automatic systems minus the 4 faulty ones with MAP < 0.05. The improvement over the results with unstandardized AP reported in Table 4.3 is shown in italics.

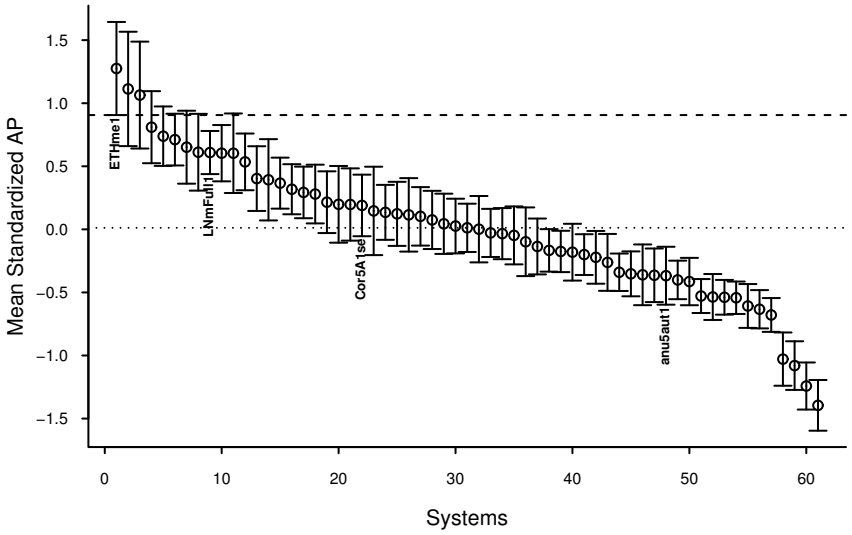


Figure 4.7: The 95% confidence intervals on mean standardized AP scores for the TREC 5 Adhoc Track systems, using a t distribution. Systems are ordered by their mean standardized AP scores. The first, tenth, twentieth, and seventy-fifth percentiles systems by AP (not necessarily sAP), previously reported in Figure 4.2, are labelled. The lower bound of the first system and the mean of the median system (by sAP) are marked with horizontal lines.

addition, for this evaluation environment at least, standardization also boosts the discriminative power of the paired test. This result suggests that standardization may also help to regularize per-topic score deltas between systems, making significance tests more reliable. See, however, the study of Voorhees (2009b), which failed to confirm this effect for standardization. More work is required in this direction.

Weak two-sample tests bring with them wide confidence intervals on true mean scores, and thus uninformative absolute scores; standardization's strengthening of two-sample tests should mean less overlap in intervals and more informative absolute scores, and Figure 4.7 demonstrates that this is indeed the case. The figure displays the 95% confidence intervals on the mean standardized AP scores for the TREC 5 Adhoc Track systems. These intervals overlap considerably less than for unstandardized AP (Figure 4.3). Before standardization, the top system's confidence interval overlapped with

	Collection				AP, T03.rob – T04.rob
	T6.adh	T7.adh	T8.adh	T03.rob	
Pearson's r	0.997	0.999	0.999	0.995	0.943
Kendall's τ	0.945	0.965	0.971	0.935	0.742

Table 4.7: Pearson's r and Kendall's τ correlations, on overall system orderings under standardized AP, for TREC 2004 Robust track systems on each of the earlier sub-collections. The comparison is between standardizing based on the original experimental systems and on the TREC 2004 Robust track systems themselves. As a reference point, the rightmost column gives the correlations between the ranking of the TREC 2004 Robust systems using unstandardized AP between the TREC 2003 Robust and TREC 2004 Robust sub-collections.

the median's; after standardization it is well clear by the twelfth system. Similarly, the median (thirty-first) system without standardization overlapped the ninth and fifty-first systems; with standardization, the range has halved to the nineteenth system at one end, and the fortieth at the other. It can be observed from the confidence intervals displayed in Figure 4.7 that, for instance, the standardized AP score for `ETHme1` is 1.28 ± 0.37, for `LNmFull1` 0.61 ± 0.18; from this, it can immediately be inferred that the two systems are significantly different (as indeed they are, at the $\alpha = 0.01$ level). The two systems' unstandardized mean AP score ranges of 0.32 ± 0.07 and 0.28 ± 0.08 are much less informative. In fact, a paired test on unstandardized scores narrowly fails to find significance; this is one of the cases where standardization has increased the discriminative power of paired tests.

4.5 Standardizing non-reference systems

So far, we have examined the situation in which the set of systems whose scores are standardized is also the set of reference systems, from which the standardization factors are drawn; that is, the system set is *self-standardized*. Such a scenario would apply to the original collaborative experiment in which a test collection is created. But the collection will then be re-used for many subsequent experiments. How reliable are the original standardization factors when used on new, non-reference systems?

The TREC effort has produced a data set well-suited for such experiments: the TREC 2004 Robust test set. As described in Section 3.5.2, the TREC 2004 Robust collection uses the same corpus, and includes the same topics, as the TREC 6, 7, and 8 AdHoc and TREC 2003 Robust tracks. (The TREC 6 sub-collection has an important peculiarity: the "description" component of many topics do not include all query keywords, leading to anomalously low scores for description-only runs.) Thus, each of these earlier sub-collections has two sets of runs against it: the original runs made at the time of the initial experiment, and the runs made in 2004 as part of that year's Robust track. We use this runset extensively in Section 4.6 for cross-collection comparisons; here, it is employed to investigate the reusability of standardization factors within the one collection.

The first question is how much of a difference the choice of reference systems makes on the outcome of standardization. This is explored by comparing the use of the original and the TREC 2004 runsets as reference sets for each of the pre-2004

Metric	Unstandardized				Self-std.			Orig-std.			
	s,φ	t	st	ρ_I	s,φ,ρ_I	t	st	s,φ	t	st	ρ_I
AP	7	66	27	20	25	0	75	20	10	70	22
P@10	4	60	36	11	12	0	88	10	8	82	11
RBP.95	5	66	29	14	18	0	82	15	8	77	16
nDCG	13	59	28	32	37	0	63	32	7	61	34

Table 4.8: Variance components and reliability measures as percentages (%) for different metrics and standardization types on the TREC 2004 Robust runsets, over Topics 301–450 and 601–650. The first block are unstandardized scores. In the second, scores are standardized by the TREC 2004 Robust runs themselves. In the third, scores are standardized by the original runsets for the TREC 6 through 8 and TREC 2003 sub-collections. The components are: s, system; t, topic; st, system–topic interaction. The measures are the indexes of absolute (φ) and relative (ρ_I) comparability. The percentages of the first three columns in each row of each block add up to 100, rounding effects aside.

sub-collections. We standardize the TREC 2004 runs on these sub-collections with the original runsets as reference systems on the one hand, and with the TREC 2004 runsets themselves as reference sets on the other. Table 4.7 gives the Pearson's r and Kendall's τ correlations between the standardized AP scores of the TREC 2004 systems under the two reference sets. The minimum r is 0.995; the minimum τ is 0.935. These correlations indicate near-equivalent rankings, even for the anomalous TREC 6 AdHoc collection. In comparison, ranking the TREC 2004 systems using unstandardized AP on the TREC 2003 Robust and TREC 2004 Robust sub-collections gives a Pearson's r of 0.943, and a Kendall's τ of 0.742. The latter figures indicate the noticeably different rankings that different sub-collections produce, even using the one metric. Standardization using different reference systems is thus far more reliable than evaluation using different topic sets.

The question of the re-usability of reference standardization factors can be further investigated by analyzing the components of variance. The analysis is run over the TREC 6 through TREC 8 AdHoc and TREC 2003 Robust sub-collections. Components of variance are calculated on the TREC 2004 Robust runset for three score types: the unstandardized scores; the scores self-standardized by the TREC 2004 runs; and the scores standardized by the original runset for the year the sub-collection was originally formed. The results are shown in Table 4.8. As before, the topic effect is very strong for the unstandardized scores, as much as two-thirds of the variance; and the topic effect is eliminated for the self-standardized score, with its variance split proportionately amongst the system and system–topic interaction components. Again, the index of absolute comparability, φ, increases greatly with standardization, and the index of relative comparability, ρ_I, very slightly, with the two being equal for self-standardized scores. The stronger system effect achieved by nDCG is once more notable. Switching to the original runsets as the reference set for standardization (shown in the rightmost block of Table 4.8) leads to the re-appearance of the topic effect. This means that the standardized TREC 2004 systems as a group find some topics easier, others harder, than the original runsets did. The re-appearing topic effect is still less than the system effect, though, and is only a seventh of the unstandardized topic effect. Absolute comparability (φ) is still much higher than for unstandardized scores, and even relative

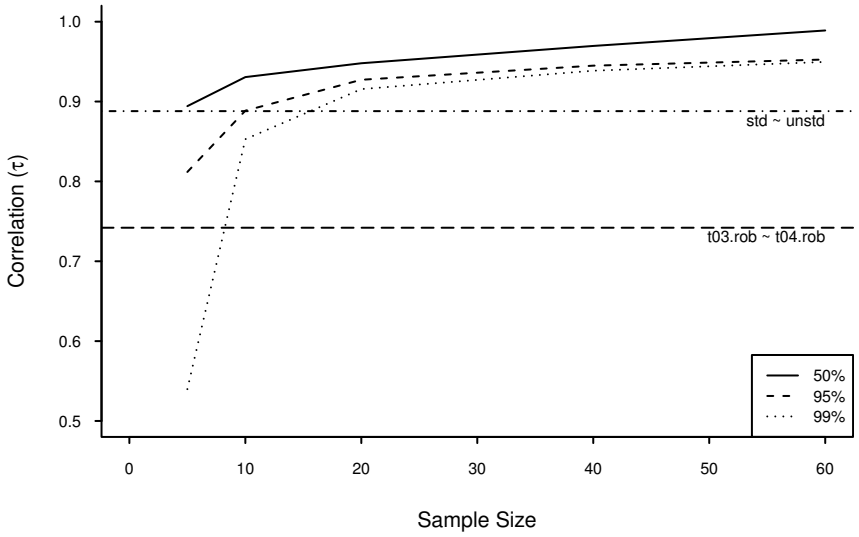


Figure 4.8: Percentiles of Kendall's τ between rankings on partial and full standardization system sets, using standardized AP. The test collection and standardization systems are from the Robust track of TREC 2003; the evaluated systems are those submitted to the Robust track of TREC 2004, scored against the TREC 2003 topics. There were $2{,}000$ random samples made for each sample size. The full standardization set has 78 systems. The Kendall's τ between the TREC 2003 and TREC 2004 sub-collections on unstandardized AP ($\tau = 0.742$), and between unstandardized and standardized AP on TREC 2003 ($\tau = 0.888$), are also shown; the latter standardized AP scores are TREC 2004 runs standardized with TREC 2003 systems.

comparability (ρ_I) continues to receive a marginal benefit in most cases. Where the reference set is as broad and diverse as those found in TREC experiments, standardization factors demonstrate considerable re-usability.

The reference systems can be considered as a (hopefully representative) sample of the full "population" of retrieval systems. Therefore, the standardization factors derived from the reference systems can be taken as estimates of the population (or "true") factors. A natural question then is how many systems are required to derive reliable standardization factors. We measure this as the correlation between the system ranking obtained with the full reference set on the one hand, and the system ranking obtained from a sub-sampled reference set on the other. (Note that the full qrel set is used in all cases; the effect of reducing the number of pooled systems is not examined.) The systems are the TREC 2004 runs, ranked on the TREC 2003 topics, using the TREC 2003 systems as the reference set for standardization; thus, the standardized and reference sets are distinct. The procedure is to sample from the reference systems; derive standardization factors from the sample; use these to standardize the scores of the evaluated systems; and calculate the Kendall's τ between the system ranking from the sampled standardization and from the full standardization. This is repeated multiple times for each sample size. The 50th (median), 95th and 99th percentile lowest Kendall's τ figures are recorded. The whole process is then repeated for other sample sizes. Figure 4.8 reports the results as a function of varying sample set sizes, using standardized AP; the unstandardized inter-collection and standardized to unstandardized intra-collection τ

are also given for comparison. On the line plotting the median correlation over the 2,000 samples, a set of as few as 5 reference systems gives ranking stability equivalent to the standardized–unstandardized comparison, though an unlucky choice of reference systems gives much less reliability, as the 99th percentile shows. Achieving the equivalent of standardized–unstandardized stability requires 10 reference systems at the 95th percentile, and 20 at the 99th percentile. Far fewer systems are needed to provide reliable standardization factors than are typically required for a reliable relevance assessment pool.

4.6 Cross-collection comparability

So far, we have examined the use of standardization within the one collection, first where systems are self-standardizing, and then where the reference and standardized systems sets are distinct. We have observed that standardization removes the topic component of variance, completely for self-standardization, and substantially where standardized and reference systems are distinct. We claimed that the removal of topic variance was of most benefit for comparisons between collections. Now it is time to empirically validate that claim.

In investigating the question of cross-collection comparability, two kinds of collection pairs need to be considered. The first are those where each collection has been randomly sampled from the same population; for instance, collections that have the same document corpus and that have topic sets randomly sampled from the same query stream. Such collections will be termed *randomly co-sampled*. The second type are collections which have not been randomly sampled in this way, and between which there may be some systematic difference; what will be referred to here as *natural* collections. As the name "natural" implies, most collection pairs fall into the latter category, while the former must be artificially created. Randomly co-sampled collections by definition differ only by chance, so that (for instance) the conditions for hypothesis testing between them are met. Statistical comparisons between natural collections must be more tentative or assumption-based, since the true nature of any systematic difference between them is generally unclear.

In our experiments, randomly co-sampled collection pairs are constructed from the TREC 2004 Robust test set by sampling from its topic set. Here, the 100 topics from the AdHoc tracks of TREC 7 and TREC 8, namely Topics 351–450, are used; these two sub-collections are chosen because they are the most mutually homogeneous. The runs are those from the TREC 2004 runset, with scores standardized using the original TREC 7 and TREC 8 systems as the reference sets for their respective sub-collections. The topics are randomly partitioned into two halves to form two randomly sampled collections. The random partitioning is repeated multiple times to generate a set of identically-sampled collections. The natural collections, in contrast, are simply the original sub-collections from which the TREC 2004 Robust collection was formed.

Note that in this section we are exploring a particular form of cross-collection comparison under standardization, that in which the reference set for each collection differs. An alternative form is one in which the same set of reference systems is used for both collections. The latter is preferable, other things being equal, because the reference systems are consistent; different reference sets could have different levels of performance, and hence produce standardization factors (in particular, means) of different stringencies; that this occurs in the data set under experiment here will be observed later (see Table 4.12 below). The collaborative TREC style of collection formation,

however, naturally leads to each collection having its own reference set. The situation of TREC 2004, where a large set of systems are collaboratively run across a number of collections, is quite unusual. A common reference set can be created by a third party, but such a reference set is likely to be narrower than a collaboratively-created one. These issues are discussed further in Section 4.7.

4.6.1 Measuring comparability

Score comparability between collections means that, if we run the same system against two collections, it should receive similar scores for each collection. Here, "similar" can be understood in the general sense of receiving aggregate system scores that are not too different; or, more narrowly, as receiving topic score sets that are not found significantly different under a significance test. Note that we explicitly are not measuring comparability by the correlation (rank or otherwise) between the scores that the one set of systems achieves on two different collections. Correlation only measures relative, not absolute, scores; what we are exploring here is whether absolute scores from different collections can be compared.

Comparability of mean scores

System score comparability can be measured by the root mean squared error (RMSE) between the mean scores that each system achieves on different collections. Continuing the notation of Section 4.1, let $\mathcal{S}$ be our set of evaluated systems, of size $S = |\mathcal{S}|$. Consider two collections, C and D. Let $\overline{\mathbf{M}^C}_{s*}$ be the score that system $s \in \mathcal{S}$ achieves (under some metric) on collection C (that is, the mean of the scores that s achieves on the topics making up C), and similarly for $\overline{\mathbf{M}^D}_{s*}$. Then the *root mean squared error* between C and D is:

$$\text{RMSE} = \sqrt{\frac{\sum_{s \in S} \left(\overline{\mathbf{M}^C}_{s*} - \overline{\mathbf{M}^D}_{s*} \right)^2}{S}} \tag{4.9}$$

Essentially, the RMSE value in Equation 4.9 measures the average difference between mean system scores that occurs, under the same metric, from using different topic sets; the greater this average difference, the less stable mean scores are for that metric, and therefore the less comparable they are between collections.

The raw RMSE value, however, is dependent upon the dispersion of mean score values for a metric. To take a simple case, if the scores for one metric are precisely ten times the scores for another, then the RMSE for the former metric will be ten times the RMSE for the latter one, even though the comparability is effectively the same. Some unnormalized metrics (SP, DCG) are not upper-bounded by 1, and standardized metrics are not bounded at all. To compare RMSE scores achieved for different metrics, therefore, a metric's RMSE score must be adjusted by the natural variability of mean scores for that metric, as measured between different systems on the same collection. We measure this variability as the standard deviation of mean scores that the systems $\mathcal{S}$ achieve on each of the collections C and D, and then average those standard deviations. Normalizing by this measure of natural mean score variability for a metric, we derive a *coefficient of inter-collection comparability*, κ (read "kappa"):

$$\kappa = \frac{2 \cdot \text{RMSE}}{\text{sd}(\{\overline{\mathbf{M}^C}_{s*} : s \in S\}) + \text{sd}(\{\overline{\mathbf{M}^D}_{s*} : s \in S\})} \tag{4.10}$$

A κ value of k for a given metric means, roughly speaking, that the average difference in mean scores that the one system achieves on two collections is k times the natural standard deviation of mean scores for that metric. Adjusting by standard deviation in our measure corresponds to the adjustment for the variability of a metric that an experimenter would use, based on (statistical or informal) experience, in interpreting the significance of score deltas for a particular metric; consider, for instance, the empirical calibration of size of score delta and reliability of comparison carried out in Buckley and Voorhees (2000).

Comparability under significance tests

Comparisons of system performance between collections are initially made on mean scores; but the statistical significance of score differences must then be tested. This raises the question of the reliability of two-sample significance tests between collections. For such tests, reliability has two aspects: the rate of false positives, and the rate of false negatives. A false positive occurs when two identical systems are found to be significantly different; a false negative when significance is not found even though the two systems differ by some amount. In calculating both rates, the central problem is establishing what are true negatives and true positives.

The rate of false positives can be investigated by testing a system for significance against itself, when run on two different topic sets. Obviously, a system is not significantly different from itself, so if a significance test finds that it is, it must be a false positive. The experiment is to take a topic set, and randomly partition it in half. A two-sided, two-sample t test is applied for every system in the system set, between the scores the system achieves on each partition of the topic set. The proportion of the tests that achieve significance at the $\alpha = 0.05$ level is recorded as the false positive rate for that partition. The partitioning is repeated multiple times, to produce a distribution of false positive rates. The expected false-positive rate for all metrics, whether raw, normalized, or standardized, is 0.05, the same as α. This follows directly from performing random co-sampling. What is of more interest is the stability of standardization tests; that is, the variability in false positive rates over different co-samples. This variability can be measured by taking a high percentile of the distribution; we take the 97.5th percentile. The variability measures the risk that, by chance, we end up with a pair of incomparable collections. This risk is all the more serious in that, in reality, once a collection is formed, it is reused unchanged in hundreds of experiments; any distortions once created will be perpetuated indefinitely.

To illustrate, we calculate the distribution of false positive rates on our co-sampling dataset for the AP metric; the 97.5th percentile rates for other metrics are given later. Figure 4.9 displays the results of the experiment. The mean rate of false positives is 0.049 for both unstandardized and standardized AP scores respectively, close to the expected $\alpha = 0.05$. But the variability of false positive rates for the unstandardized AP scores is much greater than for the standardized ones. For unstandardized AP, most topic partitionings show no false positives at all; this indicates an insufficiently sensitive test, since a rate of 0.05 is expected. At the other extreme, one of the random partitionings finds all 110 of the 110 systems to be better than themselves—a real boon for the IR researcher with a paper to publish. In contrast, standardized scores give a median false positive rate not too far from the desired mean (0.036, compared to 0.05), and in only 2.5% of cases does the false positive rate exceed 0.15.

Determining the rate of false negatives is more difficult, in that we do not know what the true positives are—that is, which systems truly are better than which. The

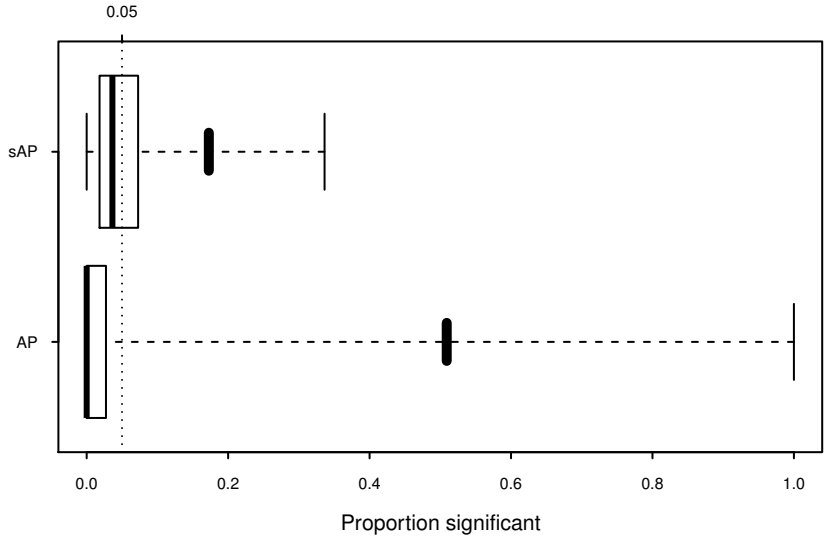


Figure 4.9: False positives on two-tailed two-sample t tests at $\alpha = 0.05$ for TREC 2004 Robust systems, using 50-topic randomly-partitioned subsets of Topics 351–450, repeated $5{,}000$ times. The line within the box is the median; the left and right box edges are the 25th and 75th percentiles, respectively; the dashed whiskers extend to the extreme values; and the thick line on the right whiskers marks the 97.5th percentile. The dotted vertical line is the expected mean of both distributions; the observed means fall approximately on this line.

approach adopted here is to identify system pairs between which a paired significance test finds highly significant differences under all of the raw, normalized, and standardized variants of a given metric. Such system pairs are regarded as being pairs in which one system is almost certainly better than the other. The false negative rate is then determined by sampling true positive pairs and topic partitions. A true positive pair is chosen at random, along with a random topic partitioning, with one half of the topics assigned to the first system, the other to the second. Then, a two-tailed, two-sample t test is performed between the systems, both with standardized and with unstandardized AP scores. If this test fails to find significance, that is taken as a false negative. The process is repeated multiple times to produce an estimate of the false negative rate. The results of the false negative experiment are given below.

4.6.2 Comparability of co-sampled collections

We begin by examining comparability between randomly co-sampled collections. Randomized topic set repartitioning is used to determine κ, the inter-collection comparability of mean scores, for different metrics. Figure 4.10 gives the mean κ of multiple random partitionings of the TREC 7 and TREC 8 AdHoc topics; recall that smaller values of κ indicate greater comparability. The metrics P@10, RBP with $p = 0.95$, SP (unnormalized AP), and DCG are compared, together with their normalized and standardized variants. The results show that every metric with standardization is more stable than any metric in its raw form. And standardization leads to significantly greater stability than normalization, even on co-sampled collections. (As will be seen later, normaliza-

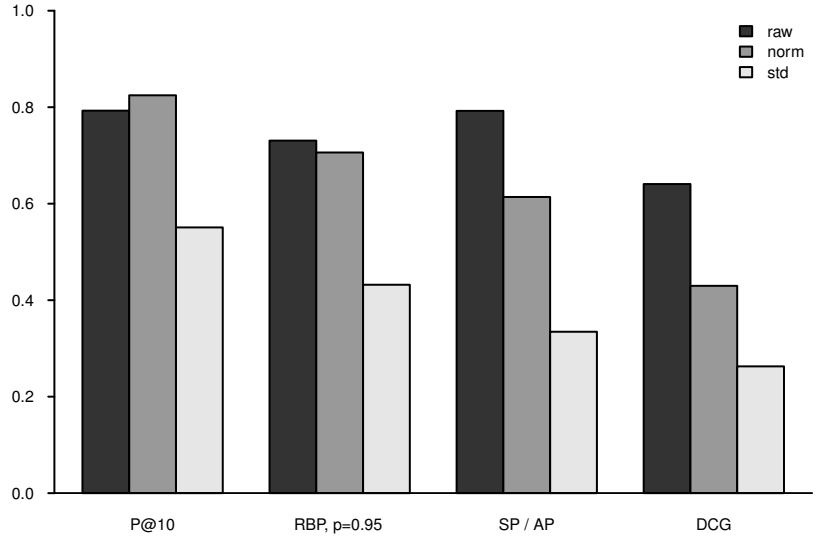


Figure 4.10: Mean inter-collection comparability, κ, for TREC 2004 Robust track systems on $1{,}000$ random partitionings of Topics 351–450 from the TREC 7 and TREC 8 AdHoc tracks, for different metrics, with and without normalization and standardization. Standardization factors are derived from the original TREC 7 and TREC 8 systems. Higher values indicate less comparability.

tion can be far less robust on natural collections.) Standardized system scores are more reliably comparable between collections, and thus more meaningful in themselves.

A method for calculating false negative rates in significance tests was described in Section 4.6.1; we now apply it to the co-sampled data set. True positives are identified as system pairs achieving significance at the 0.001 level in a paired t test on all 100 of the TREC 7 and TREC 8 AdHoc track topics. A total of $5{,}000$ repeated resamplings are made, each of a topic partitioning and "true" significant system pair, with topics partitioned into two subsets of 50 topics each. The target significance level is set at 0.05. Figure 4.11 gives the false negative rates for different metrics and normalization types. False negative rates for unstandardized metrics vary from 0.25 up to more than 0.4; for standardized metrics, they are all below 0.08, and as little as 0.03 for sDCG and sAP. The results are unsurprising, given we have already seen that two-sample tests are much more powerful for standardized than for unstandardized scores. These results demonstrate that real and substantive differences between systems are much less likely to be missed by significance tests in cross-collection comparisons when standardized scores are used than when unstandardized ones are.

That standardized AP has a more stable false positive rate than unstandardized AP was demonstrated in Figure 4.9 above; we now examine this rate for other metrics, in their raw, normalized, and standardized forms. Figure 4.12 gives the 97.5th percentile false positive rates for the TREC 2004 Robust track systems over the TREC 7 and TREC 8 AdHoc track topics; each bar represents the information represented by the solid line in Figure 4.9. The metrics SP and DCG have higher discriminative power than RBP or P@10, so the fact that they have higher false positive rates is not surprising. For all metrics, standardization enormously decreases the 97.5th percentile false positive rates, from around 50% to just over 15%. This is achieved without harming

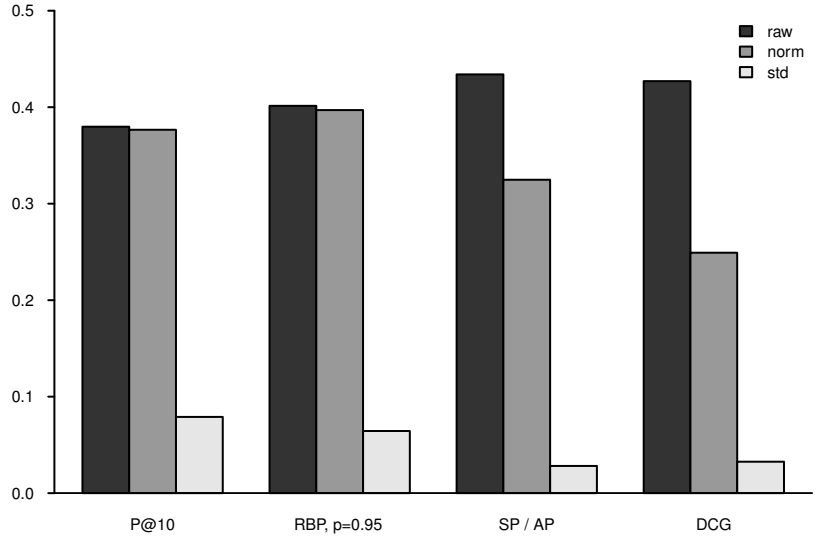


Figure 4.11: False negative rate for various metrics and forms of normalization. A false negative is a failure to find significance, using a two-tailed, two-sample t test at level $\alpha = 0.05$, between two systems that are in fact different from each other. Truly different systems are those which, for a given metric, are found significantly different in a paired t test at level $\alpha = 0.001$, under the raw, normalized, and standardized forms of the metric. Of the $5{,}955$ system pairs, $1{,}714$ were found truly different for P@10, $2{,}408$ for RBP, $2{,}896$ for SP, and $3{,}429$ for DCG. False negative rates are calculated over the 110 TREC 2004 Robust track systems, using $5{,}000$ samplings of system pairs and partitionings of Topics 351–450 from the TREC 7 and TREC 8 AdHoc tracks. Standardization factors are derived from the original TREC 7 and TREC 8 systems.

discriminative power. For instance, for the TREC 8 sub-collection, the proportion of system pairs found significantly different on a two-tailed, paired t test at level $\alpha = 0.05$ is 68.7% for DCG, 69.3% for nDCG, and 68.8% for sDCG. Normalization, in contrast, does little to reduce upper-percentile false positive rates. Thus, unstandardized metrics, in a cross-collection comparison between systems, are more likely than standardized metrics both to miss substantive differences when they exist (as shown in Figure 4.11), and to find them when they don't. This, and the mean score comparability values report in Figure 4.10, show that, where random co-sampling has occurred, use of standardized metrics leads to far more reliable inter-collection comparisons.

4.6.3 Comparability of natural collections

Randomly co-sampled collections are amongst the most favourable cases for inter-collection comparisons. In practice, different natural collections are not identically sampled. The AdHoc and Robust TREC collections do, however, use the same document corpus and were built with similar methodologies. Comparability between these collections would be desirable, even expected; absolute metric scores should be able to convey information about substantially equivalent collections. We now explore the comparability of metrics between natural collections, and the effect of normalization and standardization on this comparability.

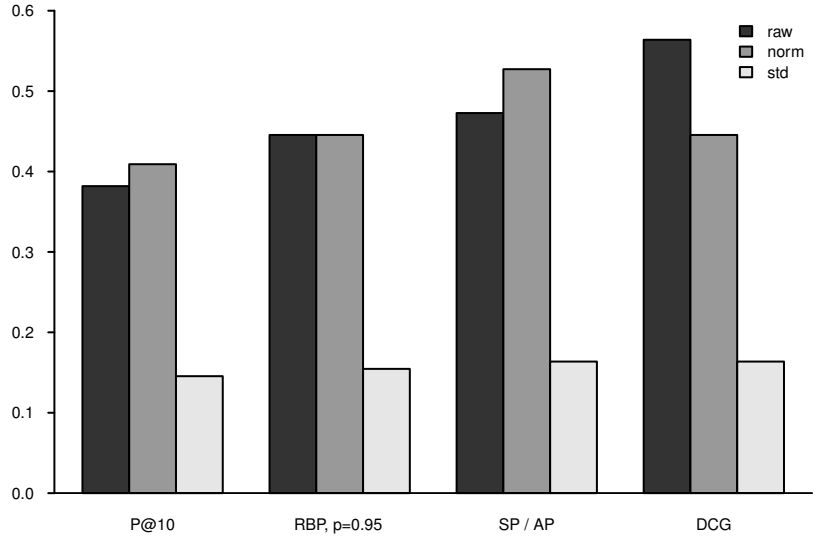


Figure 4.12: The 97.5th highest percentile false positive rates for various metrics, with different forms of normalization. A false positive is a finding that a system is significantly different from itself using a two-tailed, two-sample t test at level $\alpha = 0.05$. False positive rates are calculated for the 110 TREC 2004 Robust track systems, by 5,000 random re-partitionings of Topics 351–450 from the TREC 7 and TREC 8 AdHoc tracks. Standardization factors are derived from the original TREC 7 and TREC 8 systems.

	T8.adh	T03.rob	T04.rob
T7.adh	0.627	1.857	1.285
T8.adh		1.387	0.859
T03.rob			0.583

Table 4.9: Inter-collection comparability κ for unstandardized system AP scores between each pair of collections in the TREC 2004 Robust set for all systems participating in the track. Higher values indicate less comparability.

The first question is whether systems achieve similar mean scores on different collections; the coefficient of inter-collection comparability, κ, is used to measure this. Table 4.9 shows κ for each pair of collections used in the TREC 2004 Robust track, measured on the TREC 2004 Robust runs under the AP metric. The two AdHoc sub-collections are relatively comparable to each other, as are the two Robust subcollections, with each pairing having a κ around 0.6. But comparisons between any AdHoc and any Robust sub-collection are unreliable, with κ values ranging from 0.85 up to 1.85; that is, the average error of mean scores between collections can be almost twice the standard deviation within collections. Significance tests between AdHoc and Robust collections under AP are also highly unreliable, as shown by the high rate of false positives in Table 4.10. Almost all systems seem significantly better than themselves when evaluated using AP against the TREC 2003 collection than when evaluated against the TREC 7 collection.

	T7.adh	T8.adh	T03.rob	T04.rob
T7.adh		0	0	0
T8.adh	2		0	0
T03.rob	103	57		2
T04.rob	61	8	0	

Table 4.10: Number of the 110 TREC 2004 Robust track systems that were found to be significantly better when tested on the sub-collection in the row than on the sub-collection in the column, under unstandardized mean AP. Significance is determined by a two-sample, one-tailed t-test, at level $\alpha = 0.025$.

	T8.adh	T03.rob	T04.rob
T7.adh	0.349	0.396	0.452
T8.adh		0.464	0.462
T03.rob			0.394

Table 4.11: Inter-collection comparability κ for system standardized AP scores, between each pair of collections in the TREC 2004 Robust set, for all systems participating in the track. Standardization factors are derived from the original experiments.

In contrast, sub-collections are much more comparable under standardized AP, even across the AdHoc–Robust separation, as can be observed from Table 4.11. Standardized scores have a lower κ for every collection pair than do the unstandardized AP scores in Table 4.9. The minimum κ for unstandardized scores was 0.583; the maximum κ for standardized scores falls well below this at 0.464, and the minimum is 0.349. Moreover, comparability between AdHoc and Robust sub-collections is only slightly worse than within them, ranging from 0.40 to 0.45 for the former, from 0.35 to 0.40 for the latter. Table 4.12 shows the false positive rates for significance testing between two sub-collections. The number of false positives among the 110 systems is much lower overall than for unstandardized AP. There is, however, a persistently high false positive rate, of over 10% but less than 15%, for finding TREC 2004 systems significantly better than themselves when tested on an earlier sub-collection rather than on their own sub-collection. There are two causes of this. First, the TREC 2004 systems do, on the average, get slightly higher scores on the earlier sub-collections than the original runsets do (see ahead to Table 4.13), perhaps due to training effects. And second, the TREC 2004 systems are self-standardized on their own sub-collection, other-standardized on the earlier ones. Other-standardization leads to higher variability than self-standardization; therefore, a larger proportion of system comparisons end up beyond the significance mark. Nevertheless, comparisons are much more reliable overall for standardized than for unstandardized scores.

While on this topic, it is interesting to consider whether the sub-collections themselves are significantly different from each other, as measured by the κ statistic, with or without standardization. The null hypothesis here is that, for any pair of sub-collections, the topics have been randomly assigned to the two sub-collections; where by "topics" must be understood not just the queries, but the relevance assessments, and normalization and standardization factors that go with them (including, therefore,

	T7.adh	T8.adh	T03.rob	T04.rob
T7.adh		4	5	14
T8.adh	2		5	13
T03.rob	4	13		12
T04.rob	3	8	1	

Table 4.12: Number of the 110 TREC 2004 Robust track systems that were found to be significantly better when tested on the sub-collection in the row than on the sub-collection in the column, under standardized mean AP. Standardization is performed using the standardization factors from the original experiments the sub-collections were formed for. Significance is determined by a two-sample, one-tailed t-test, at level $\alpha = 0.025$.

the effect of the normalizing or standardizing runsets as well). We can test this null hypothesis by once again using topic set re-partitioning. The topic sets of the two sub-collections are combined, then randomly re-partitioned. The proportion of these re-partitions giving a κ greater than that of the natural collections is the p value of the test. So, for instance, under unstandardized AP, the natural TREC 7 and TREC 8 sub-collections have a κ of 0.627 (Table 4.9); the random re-partitioning test finds a (highly non-significant) p value of 0.353 for this outcome. In contrast, the observed κ of 1.857 under unstandardized AP between the natural TREC 7 and TREC 2003 sub-collections is significant at the 0.001 level. Similarly, under standardized AP, TREC 7 and TREC 8 are not significantly different, but TREC 7 and TREC 2004 are at the 0.01 level. The latter significance is despite the fact that the κ value of 0.452 is lower than that of any of the unstandardized comparisons. This is an example of the oft-repeated point that significance of a difference does not equate to the size of that difference. Standardization reduces topic variance between collections, but it also does so within them, meaning that a smaller between-collection κ can be significant on standardized scores, where a larger one is not significant on unstandardized scores.

Figure 4.13 gives the inter-collection comparability, κ, between system mean scores for various metrics, in their raw, normalized, and standardized forms. The value of 1.1 in the middle bar of the SP/AP group, for instance, is the mean of the six values reported in Table 4.9. Note that these means include the two same-track pairs as well as the four different-track (Robust-to-AdHoc) pairs; if only the latter were included, the results would be even less flattering to unstandardized AP and DCG. Standardization moderately improves RBP's observed cross-collection comparability, and marginally worsens that for P@10. The improvements for SP/AP and DCG, however, are dramatic, even from their normalized forms. Comparison to the co-sampled results in Figure 4.10 shows that the standardized scores on the natural collections achieve results similar to those on the co-sampled ones, but DCG and AP, in both unnormalized and normalized forms, perform much worse. For instance, nDCG achieved a mean κ of 0.43 on the co-sampled collections; for the natural collections, the mean κ is 0.88. Errors between mean scores on the natural collections are twice what they are on the co-sampled ones.

The reason why the normalized metrics are even less comparable between the natural collections than for co-sampled collections is due to differences in the constitution of the set of known relevant documents $\mathcal{R}$, and hence in the normalization factor $R = |\mathcal{R}|$. Both Robust and AdHoc judgment pools were formed by pooling to

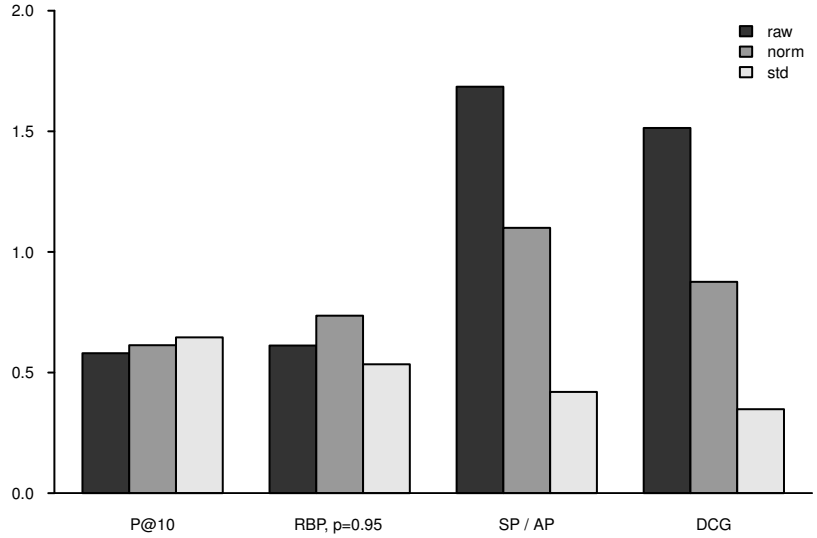


Figure 4.13: Mean inter-collection comparability, κ, for TREC 2004 Robust track systems between each pair of the TREC 7 AdHoc, TREC 8 AdHoc, TREC 2003 Robust, and TREC 2004 Robust collections, for various metrics, without and with standardization. Standardization is performed based on the original experimental systems. Higher values indicate less comparability.

depth 100 (depth 125 for TREC 2003), but the number of participant groups and therefore pooled systems was quite different, with 42 and 41 systems pooled for the two AdHoc collections and only 16 and 14 for the Robust ones. Moreover, the AdHoc tracks included a large number of manual runs, which uniquely identified around 25% of the known relevant documents, whereas the Robust tracks had no manual runs contributing. The effect of these differences in the constitution of the assessment pool can be seen in Table 4.13. The average number of known relevant documents per topic, R, is much greater for the AdHoc than for the Robust collections. The Robust topics are not harder than the AdHoc ones, with the TREC 2004 Robust systems receiving very similar average system P@10 and RBP scores. But the normalized metrics such as AP and nDCG are misled, as it were, by the smaller values of R into thinking these topics are harder, and normalizing their scores higher. Conversely, SP and DCG, being non-convergent metrics that evaluate deep in the runs, give much higher average scores to the sub-collections with more known relevant documents. Standardization, not being reliant upon R, is not affected by these differences in pool coverage. The TREC 2004 runs are, however, slightly stronger as a group than the earlier runsets (if only through the training effect of having the earlier collections available to them in advance), so standardizing by original runs gives them slightly higher scores on the earlier sub-collections than on their own.

These experiments demonstrate that, as anticipated, standardization greatly increases comparability between distinct test collections, so much so that significance tests between collections become feasible. In contrast, normalization based on the number of relevant documents, as performed in AP and nDCG, is sensitive to variability in the way relevance assessments are collected, such as differences in pooling. Standardization is robust to these differences. Where different collections have different reference

	T7.adh	T8.adh	T03.rob	T04.rob
Judged	1606.9	1736.6	958.7	710.0
Relevant	93.5	94.6	33.2	42.1
P@10	0.452	0.450	0.466	0.434
RBP.95	0.331	0.343	0.308	0.308
AP	0.212	0.244	0.327	0.293
nDCG	0.479	0.514	0.617	0.574
SP	17.88	20.29	9.71	11.61
DCG	9.11	9.81	6.22	6.75
sAP	0.088	0.022	0.077	0.000
sDCG	0.137	0.081	0.083	0.000

Table 4.13: Mean number of documents judged and mean number of documents found to be relevant for the different sub-collections of the TREC 2004 Robust collection, and mean P@10, RBP $p = 0.95$, AP, nDCG, SP, DCG, standardized AP, and standardized DCG scores for the TREC 2004 Robust track systems run against each sub-collection. Standardization is performed using original runsets.

sets, standardization is subject to differences between reference sets, an issue that is examined further in Section 4.7; on the experimental data, however, these differences are slight, and far less than those arising from topic variance in unstandardized metrics.

4.7 Outstanding issues in standardization

We have seen that standardization removes topic variance in self-standardized runsets (runsets in which reference and standardized systems are the same) (Section 4.4). We have also seen that, at least for the TREC 2004 Robust test data, standardization of non-reference systems greatly reduces topic variance (Section 4.5). We have observed on the TREC 2004 data that the reduction in topic variance makes cross-collection comparability much stronger for standardized than for unstandardized metrics (Section 4.6). Thus, standardization has been shown to achieve its core goals: increasing the information that absolute mean and per-topic scores convey, and making cross-collection score comparisons, and even significance tests, feasible. There are, though, a number of outstanding issues with standardization, which this section discusses. Much of this material constitutes topics for further research; the intention here is to recognize the questions involved, and propose directions to take in addressing them.

4.7.1 Reference set dependence

Where different reference sets are used to derive standardization factors for different collections, then there is a dependency between standardized scores and the reference set. Most simply, if the reference systems for collection A are weaker as a group than those for collection B, then systems will tend to achieve higher standardized scores on A than on B. As a result, cross-collection comparisons become less reliable. This dependency has been observed in Section 4.6; the TREC 2004 runs appear to be stronger (perhaps due only to a training effect) than those of earlier years. System comparisons

between different collections are still much more dependable for standardized than for unstandardized scores; nevertheless, false positive rates for significance tests between certain collection pairs are higher than the significance level set (Table 4.11).

The apparent solution is to use the one set of reference systems for every collection. Consider, for instance, reversing the setup of our experiments on the TREC 2004 data set: take the TREC 2004 Robust systems as the reference systems, and use them to standardize and compare the scores of the original runsets against the TREC 6, TREC 7, TREC 8, and TREC 2003 collections. This represents in fact a likely use case for standardization, in which comparisons are being made between the results of systems not under the researcher's control, achieved on different collections. Precisely such a method is used in Section 8.2.1 to investigate the trend in system performance over the AdHoc and Robust tracks of TREC: a number of reference systems are run across all the relevant collections, and used to standardize the scores of the original TREC runs, to allow them to be compared over time. A problem is that new reference systems are likely to be fewer and less diverse than are found in a full collaborative experiment such as TREC, leading to less stable standardized scores and more outliers. One possible solution to the lack of diversity is a hybrid scheme, where original collaborative systems are used to control topic variance within a collection, while the common reference systems are used to adjust mean scores between collections. Other approaches are discussed in Section 4.7.3 below.

Common reference systems could also be used where not just the examined systems, but also the experimental collections, are not available. Consider, for instance, the research group of a commercial search company, reporting experiments for a proprietary system on a non-shareable data collection. Some degree of result comparability could be maintained by also running a set of publicly-available systems across the same collection and topics, and publishing standardization factors and standardized scores based on this public reference set. Such a procedure is, in a way, an extension of the experimental principle of using baseline runs to compare new methods against.

Finally, one must not be too sanguine about standardization's capacity to enable comparisons between genuinely dissimilar collections. If different collections represent systematically different search tasks, then even a common set of reference systems will have a limited ability to make results comparable between such collections. In terms of the components of variance analysis, it might be said that the inter-collection system–topic interaction variance is too high: certain systems will perform better on the task that one collection represents than on the task which the other collection does. But in such a situation, inter-collection comparisons may not make sense at all.

4.7.2 Outlier scores

The standardized scores of systems that are part of the reference set have an upper and lower limit. As discussed in Section 4.3, if there are n systems in the reference set, then the maximum or minimum score a system can achieve on a topic is $\pm\sqrt{n-1}$. This prevents any one topic from carrying too much weight in the mean score of a reference system. There is, however, no fixed bound on the standardized score that a non-reference system can achieve. If the reference systems all receive similar raw scores on a topic, the standard deviation for that topic will be small; if a non-reference system receives a score significantly above or below this mean, then its standardized score will be very large (positive or negative).

A large standardized score for a topic achieved due to a small standard deviation amongst reference systems can potentially have a disproportionate influence on a sys-

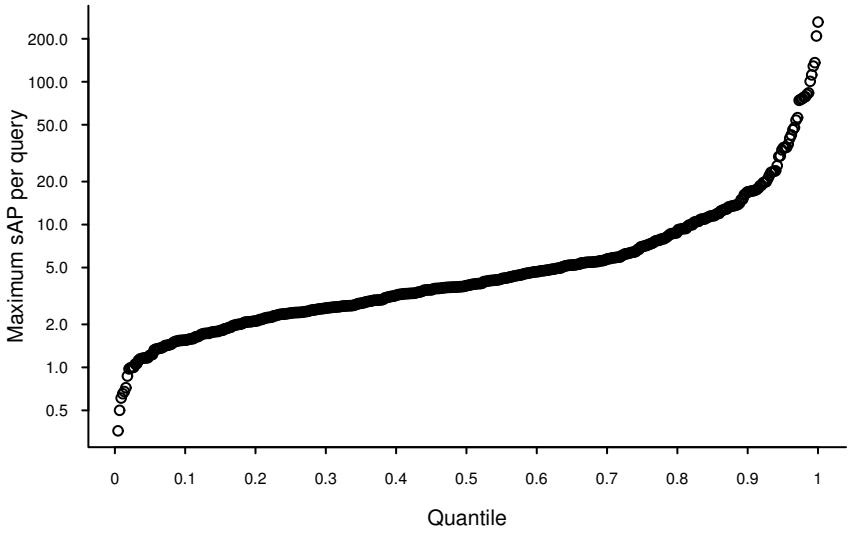


Figure 4.14: Maximum standardized AP scores achieved for each query across the TREC AdHoc and Robust runsets. The reference set was 17 variants of 5 public retrieval systems, run across all collections. The standardized runs were those of the participant systems in the TREC 3 through 8 AdHoc track, and TREC 2003 through 2005 Robust track. Note the log scale on the y axis (all maximum scores are positive).

tem's mean score. An extreme case of this occurs in the TREC 2004 Robust experimental set, where for Topic 309 all the original TREC 6 runs scored 0.0 for precision at ten, whereas four of the TREC 2004 runs score 0.1; the stopgap solution for this situation is to set all standardized scores to 0 where the reference standard deviation is also 0. Beyond this extreme case, there are a number of other standardized scores over the maximum possible reference score, based on the 78 reference systems, of 8.77; for instance, eight TREC 2004 Robust systems achieve a standardized AP score above this limit for Topic 605, with the maximum score being 11.82.

The outlier values observed on the TREC 2004 Robust runs, however, are restrained examples of the phenomenon, because the TREC participant systems constitute large and diverse reference sets. Where references runs must be produced from scratch, as for instance where a common reference set is required across a number of collections (the scenario suggested in Section 4.7.1), then only a more restricted set will generally be possible. Manual runs will be lacking, and the automatic runs will be less tuned and probably more conventional in their approach. In such a situation, the smaller number and reduced diversity of runs will lead to less stable standardization factors, causing the more frequent appearance of outliers.

The problem of narrow reference sets producing outlier standardized scores is encountered in Section 8.2.1. We compare the scores achieved by participant systems at each of the TREC AdHoc and Robust tracks. To make such a comparison, scores must be standardized by some common reference set. The reference set employed consists of five publicly available systems, run in seventeen different configurations. Some of these runs included query expansion, but no great effort at tuning was made; in particular, systems were not tuned to individual collections. This reference set is much less diverse than the original TREC participant systems. The set underperforms many of the

original manual or highly-tuned automatic runs by a wide margin. As a result, some queries receive maximum standardized scores that are quite extreme, as Figure 4.14 reveals. Almost 18% of the queries receive a maximum standardized AP score above 10.0; for one query, the maximum score is 261.9. (Negative outliers are much less extreme; the minimum standardized score is -9.7.) Such extreme outlier values can easily distort system scores and comparisons, and suggest the need for transformations that limit extreme values.

4.7.3 Transformations

The dependency of standardization upon the set of reference systems has been noted in the previous two sections. It has been observed in particular that an insufficiently diverse reference set can lead to extreme standardized scores; and such a reference set is likely to be encountered in practice when one moves beyond using the participants of TREC-style collaborative experiments. In this section, we explore a number of possible transformations and adjustments to standardized scores. These transformations are partly motivated by theoretical considerations; but they also address the dependency and outlier problems described above.

Mapping

Most standard evaluation metrics take scores in the range $[0, 1]$. It would therefore be at least aesthetically desirable for standardized scores to do so, too; a standardized score of 0 means "average", but (in the context of P@10, RBP, AP, and nDCG) is liable to be interpreted as "failed". Mapping to the $[0, 1]$ range also serves to reduce the impact of outlier values; how great the reduction is depends upon the mapping chosen.

Any function mapping from the standardized scores' (potential) range of $[-\infty, \infty]$ to the desired range of $[0, 1]$ is a possible candidate. An obvious class of functions of this sort are the cumulative distribution functions of probability distributions with infinite and continuous domains; and the most obvious of these distributions is the normal distribution. Thus, our first candidate mapping function is the cumulative density function of the standard normal distribution:

$$F_X(m') = \int_{-\infty}^{m'} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \, \mathrm{d}x \tag{4.11}$$

Under normal CDF mapping, a standardized score of 0.5 means "average"; 0.84 and 0.16 represent one standard deviation above and below average; 0.977 and 0.023 represent two standard deviations; and so forth. (The cross-collection comparisons of Section 4.6 were originally performed in Webber et al. (2008b) using normal CDF mapping, which gave similar, but slightly more stable, results to the unmapped standardized scores reported above.)

Such a mapping solves the problem of outlier values having disproportionate influence on mean scores; no matter how high the unmapped standardized score is, the mapped one cannot go beyond 1. There may, however, be a number of standardized systems with high unmapped scores, particular if the experimental set significantly outperforms the reference set as a group. In this case, the telescoping of high values will lead to a loss of discrimination between scores. All unmapped standardized scores beyond 1, for instance, are squeezed into the range $[0.84, 1]$. For almost 18% of the queries covered by the experiments in Section 8.2.1, half or more of the unmapped

standardized system scores are above 1; and for almost 8% of queries, half or more of the scores are above 2. Such queries, potentially distorting under plain standardization, become undiscriminating under mapped standardization.

A partial solution to the telescoping of outlier values under the normal CDF would be to use a mapping function that is less thin-tailed. For instance, a t distribution with 1 degree of freedom would convert a unmapped standardized score of 2 to a mapped score of 0.85, and even a unmapped standardized score of 10 goes to a mapped score of 0.968, leaving some room for discrimination amongst outlier results. The use of a broader mapping function can be seen on an intuitive level as a response to the sampling problem posed by creating reference sets: such sets are not randomly sampled, and even making them representatively diverse is difficult. Nevertheless, a solution of this sort is only ad hoc, and it is difficult either theoretically or empirically to justify the choice of any particular mapping function. Additionally, there is a trade-off between allowing discriminative power to outlier values, and yet preventing them from having a distorting effect on mean scores and comparisons.

A more theoretical objection to performing the mapping from the infinite range of plain standardized scores to the finite $[0, 1]$ range is that it involves undoing the range transformation of a complex function (standardization) with a simple one. Under this view, there should first be a mapping from the finite domain of the original scores to an infinite range; then standardization would be performed on these mapped scores; and finally a reverse mapping would be applied to bring the infinite-ranged standardized scores back into the finite $[0, 1]$ range.[1] Further investigation of this proposal is required.

Smoothing

The reference systems under standardization would ideally be a sample from the full set of possible retrieval systems; and the problems of reference set dependency and standardized score outliers can be regarded as arising from the inevitable incompleteness and partiality that characterizes any real reference set when regarded as a sample. A common approach to ameliorating the incompleteness of samples is that of *smoothing*; it is therefore natural that we consider smoothing as a solution to the sampling inadequacies of standardizing references sets.

The core idea of smoothing is to merge the model derived from some observed sample with a background model derived from the environment or some *a priori* distribution. The observed sample is limited in size, and so the distribution estimates it produces are quantized and spiky; introducing a broader background model helps smooth out these irregularities. The use of smoothing in the retrieval approach of *language modelling* provides a good example (Zhai and Lafferty, 2004). The basic notion behind language modelling is that each document is produced by randomly sampling from a probability distribution over terms, that is, a *language model*. The similarity of a query to a document is then calculated as the probability that the model that produced the document would also produce the query. The most straightforward estimate of a document's language model is simply the histogram of term frequencies in the document. But that allows no probability that the language model would produce terms that do not occur in the document, which is not justified, since we regard the document itself as only a finite sample of its underlying language model. Therefore, the language model derived from the document is *smoothed* by merging it with a language model

[1] This suggestion has been independently communicated by Stephen Robertson and Laurence Park.

derived from, say, term frequencies across the collection as a whole. Among other things, this allows a query to have a non-zero match against a document even if some query keywords do not occur in the document.

We can regard the problem of reference set incompleteness in a similar light. The fundamental idea of standardization is to determine how difficult (and variable) a query is. If the full (conceptual) population of systems were available, difficulty and variability could be determined directly from the scores this population achieves on the query. The full population being hidden, the scores of the reference set instead act as a proxy or (non-random) sample of this population. Even if the sample were random and unbiased, it would be incomplete. But we know that the sample is very much non-random, and is likely to be biased. Therefore, we sceptically (and pragmatically) smooth against some background distribution of scores.

A simple form of smoothing is adopted by us in Section 8.2.1, to avoid both the outliers of plain standardization, and the non-discrimination of CDF mapping. The approach is to add two virtual reference systems to the seventeen real ones. One of these virtual systems is inept, scoring 0 against every topic; the other is perfect, and always scores 1. Such a smoothing technique is plausible: the normalization inherent in AP means that it is always possible, in theory, for a system to achieve a perfect score of 1; and it is certainly possible to create a system that always scores 0. Such smoothing will remove the effect of outliers (or if, as is done in Section 8.2.1, normal CDF mapping is applied, it will reduce non-discrimination); the maximal achievable absolute standardized score will be $\sqrt{n+1}$, where n is the number of real reference systems. Smoothing of this sort, though, reduces the range of standard deviation factors. The minimum standard deviation, occurring when all n real reference systems score 0.5, is $\sqrt{1/2(n+2)}$. For a set of seventeen reference system as in Section 8.2.1, for instance, the minimum standard deviation factor would be 0.16; this is past the 90th percentile of actual standard deviations for that reference set, and just below the 80th percentile of standard deviations of the TREC 2004 Robust runset. Means are also pulled towards 0.5. The result is a dampened form of standardization.

An alternative form of smoothing is to add the score of each standardized system to the reference set, when that system's standardized score is being calculated. This has the effect of reducing the maximum absolute standardized score to $\sqrt{n}$, where n is the number of reference systems (excluding the standardized one). It is also less dampening than smoothing by adding $\{0, 1\}$ scores. The effect, however, is to reduce the relative performance of high-scoring systems. For instance, in the extreme case that the reference systems all achieve the same score, then each standardized score can only take one of the values $\{\sqrt{n}, 0, -\sqrt{n}\}$, depending on whether it is more than, equal to, or less than the common reference set score; a system outscoring the reference set by a long way gets the same score as one outscoring it by a sliver. This is an extreme case; but where reference systems score very close to each other (usually because they all score very badly), performance differences in standardized systems can be dramatically telescoped.

One other form of smoothing that could be employed is against background standardization factors. For instance, the standardization factors for a topic could be a combination of the raw factors for that topic, calculated as usual, and the mean of the standardization factors observed for that reference set across the topic set as a whole. This would ameliorate the situation in which a set of reference systems behaves anomalously against one or a handful of topics in the set. It does little, though, to help the circumstance that the reference systems are weak or insufficiently diverse against the topic set as a whole. Smoothing could also be considered against some prior distri-

bution, either theoretical or based empirically on historical score distributions for that metric.

Given the potential for incompleteness and narrowness in reference sets, smoothing is an attractive idea. Reference sets are most likely to be incomplete when they are gathered and run by a single research group, for instance to create common reference factors across a number of different collections, rather than when they emerge from the joint efforts of a collaborative, TREC-style experiment. Common reference sets offer the least biased foundation for cross-collection comparisons, but at the same time are likely to be the most incomplete. It is in such a situation that smoothing is most valuable. More research, however, is required to determine which of the available smoothing methods is the most reliable and robust.

4.7.4 Standardization and paired comparison

Most of the benefits of standardization outlined in this chapter relate to the informativeness of absolute scores and to score comparisons between collections. The benefits that standardization brings to relative comparisons within the one collection have been less explored. We observed in Section 4.2.2 that not merely means, but also standard deviations, have a high degree of variance between topics. It was suggested that this is an undesirable characteristic, since it results in different topics having different impacts on mean score deltas, and hence on comparisons between systems. In the mean case, this doesn't matter: if deltas agree, it does not matter much how they are distributed. But we have also seen (Table 4.2) that the system–topic interaction effect is quite high, indicating that different systems have quite different performances on different topics. The combination of variability in topic performance with variance in topic standard deviation suggests the possibility of quite unstable pairwise comparisons.

Standardization removes the variance in topic standard deviations; each topic has the same standard deviation (for the reference set). This, perhaps surprisingly, appears to have no great effect on paired significance tests; Table 4.6 found a slight increase in discriminative power for AP on TREC 5, but Voorhees (2009b) fails to confirm this effect on a broader range of metrics and runsets. More work on this question seems appropriate. It certainly is the case, as Figure 4.4 shows, that the equalizing of standard deviations does change the ordering of system ranking. One can produce arguments as to why the standardized ordering should be more reliable; the problem, as with all analysis of evaluation metrics, is to figure out how to objectively establish this reliability.

4.8 Summary

This chapter has presented a novel (for information retrieval) transform for evaluation metrics, namely score standardization. Standardization addresses the high variability in topic difficulty. The variability in topic scores is greater than in system scores; any given system–topic score tells us more about how difficult the topic was than about how effective the system is. Absolute mean system scores, quoted in isolation, are therefore largely meaningless; absolute topic scores, almost entirely so. One must know how other systems performed on the topic or collection to gauge the system's relative effectiveness. This also means that comparisons between collections are weak.

The method proposed to address the issue of topic variability is to calculate the mean and standard deviation of scores under a given metric for each topic, based on

a set of reference systems. These standardization factors are then used to convert raw scores on that topic into standardized z scores. The standardized scores convey information about the system's relative (and, if the reference set is comprehensive enough, absolute) performance on a topic, information that is missing from a raw metric score.

Standardization perfectly equalizes topic score means and standard deviations for the reference set of systems. This is reflected in a components of variance analysis by the removal of the topic variance component, making absolute scores as comparable as relative score deltas. Where standardization is applied to systems other than the reference systems, topic variability is also greatly reduced, provided the reference set is large and diverse enough.

One of the primary practical goals of standardization is to allow comparisons between scores achieved on different collections. We have demonstrated that, on TREC-style runsets, standardization greatly improves score comparability between collections, and even allows tests of statistical significance to be employed. Where each collection has its own reference set, there is a dependence between the results of standardization and this reference set; if the reference set for one collection is weaker than the reference set for another, then systems will tend to achieve higher standardized scores on the former than the latter. Nevertheless, at least on the experimental data set employed, dependence on reference systems in standardized scores is much less of a problem than topic variability in unstandardized, and even in normalized, scores.

The ideal situation for standardization is to have one reference set run across all the collections involved. Forming such a reference set, however, requires a considerable amount of work, especially if it is being retro-fitted onto existing collections. It is likely, therefore, that such a reference set will be smaller, less diverse, and less well-tuned than the system sets produced as part of collaborative experiments such as TREC. This can result in extreme outliers amongst the standardized scores. The distorting effect of outliers can be removed by mapping standardized scores to the $[0, 1]$ range, but if outliers are common enough, this can in turn lead to a loss of discriminative power. An alternative (or complementary) method is to perform smoothing over the standardization factors, for instance by introducing virtual systems or prior distributions over topic scores. More work is required to clarify this issue.

The classic statistical techniques were developed in a research environment where the typical experiment involved two samples, and where extraneous variables prevented the precise replicability of experimental results. The core tool for such an experiment is the significance test, and this has likewise become the main tool of statistical analysis within information retrieval experiments. But the test collection methodology is different from that of the two-sample test in natural or social science. Experiments are precisely replicable, and there is a wealth of information about experimental units (in particular, topics) available in the runsets that participate in the original, collection-forming experiments, not to mention runs made subsequently against those collections. Standardization is one way of taking advantage of this contextual information to improve the accuracy and informativeness of experimental results. Finding other ways to use this contextual information seems a fruitful area for further investigation.

Chapter 5

Statistical Power in Retrieval Evaluation

The effectiveness of information retrieval systems is determined through comparative evaluation on test collections. A finding that one retrieval system achieves a higher mean score than another must be verified using a test for statistical significance. Significance tests determine whether an observed difference in performance could have occurred by chance, in particular in the choice of topics; only if the probability of a chance occurrence is sufficiently low can the result be accepted as significant. If the experiment fails to find significance, though, one cannot simply conclude that no consequential difference exists. Instead, the experimenter wishes to know how large an actual difference in performance could have been missed. Additionally, when designing an IR experiment, the experimenter needs to decide how large a topic set is required to reliably detect a consequential difference in performance.

The reliability with which a significance test can detect a consequential difference is referred to as the *power* of that test. The retrieval experimenter is confronted with the question of power both when considering the use of an existing test collection, and when planning the creation of a new one. If a test collection exists with content of a suitable type for the planned experiments, the experimenter must determine whether that collection contains enough topics to reliably detect the anticipated, consequential improvement in performance. If no test collection of suitable content exists, or those that do are too small, then the experimenter is forced to create a new collection, and must decide how many topics to include in it. To answer either question, the experimenter needs to decide what counts as a consequential difference. Failure to achieve the experimental power required to detect this difference can result in an unproductive experiment, one from which neither a positive nor a negative conclusion can be drawn.

The question of statistical power has been surprisingly neglected in the information retrieval literature. In particular, while some early user studies examine it (Robertson, 1990), its application to batch evaluation using test collections has been little studied. Cormack and Lynam (2007) make an empirical investigation of the post-hoc power of different significance tests, but their method cannot be used at experiment design time. Carterette and Smucker (2007) provide a power analysis of the inferential delta AP measure (Carterette, 2007), but only for the sign test.

We begin this chapter with a description of statistical power, how it relates to statistical significance, and how power analysis is used both to examine the results of a past

experiment and to plan a new one. In Section 5.2, we examine the statistical power of the standard TREC collections, finding that a typical 50-topic TREC collection is insufficiently powerful to reliably detect a realistic improvement over a reasonable baseline. Even if a TREC collection exists in an area, it may well be inadequate to answer research questions; and if no suitable collection exists, the experimenter is forced to create their own. In Section 5.3, we examine the use of power analysis at the time of an experiment's design, to determine the number of topics required for a new collection. This calculation rests primarily on estimating the variability in score deltas, and we analyze three methods for arriving at this estimate: based on past experience; based on a trial experiment; or through an iterative testing process. Experience, if available, is free but unreliable; trial experiments are unbiased but expensive; and the iterative method is efficient but prone to bias. We therefore propose a hybrid methodology.

5.1 Statistical power

The use of hypothesis testing in IR evaluation was described in Section 3.3. There, it was explained that the probability of a null hypothesis is tested against a significance threshold, α, which sets the risk of falsely finding significance when no difference between systems exists. We are concerned now with the corresponding risk, β, of failing to find significance where it exists. This value is directly related to the *power* of a test: the likelihood that a significant difference will be found, if one system is indeed better (by a certain amount) than the other. In this section, we describe the calculation and use of statistical power.

5.1.1 The power of a test

Consider two IR systems, A and B, that are to be evaluated and compared using a test collection, containing a set of n topics $T = \{t_1, \ldots, t_n\}$. Denote the metric (say, AP) score that system A achieves on topic t as $m_{A,t}$. The mean score $\overline{m}_A$ for system A is $\sum_t m_{A,t}/n$, and similarly for system B. The difference between means, $\overline{m}_A - \overline{m}_B$, we denote as $d_{A,B}$ or simply d. It represents the *observed delta* between the systems. For each topic t, the per-topic delta, d_t, is $m_{A,t} - m_{B,t}$. Of course, $d = \sum_t d_t/n$; that is, the delta of the means is the mean of the sum of the per-topic deltas.

Having observed $d_{A,B} > 0$, we conclude that system A has outperformed system B on topic set T, under the metric employed. The significance of this difference $d_{A,B}$ is then tested using a hypothesis test. The hypothesis test assumes that the collection topics T have been randomly sampled from a larger population of topics $\mathcal{T}$. Equivalently, therefore, the observed per-topic deltas $D = \{d_1, \ldots, d_n\}$ between system A and B have also been randomly sampled from the population of score differences between the two systems, $\mathcal{D}$, over the population of topics. The *true delta*, δ, between the systems is the mean of the population of deltas, $\delta = \bar{\mathcal{D}}$. Testing for *significance* involves formulating a *null hypothesis* H_0 that the two systems have in fact identical effectiveness, that is, that $\delta = 0$, and then determining the probability p that the observed difference d or greater could have occurred by chance if this hypothesis were true. If p is below some predetermined *significance level* α (where $\alpha = 0.05$ is a common choice), then H_0 is rejected, and the alternative hypothesis, that the two systems are not equivalent, is accepted.

In hypothesis testing, the value α specifies the risk of falsely finding a significant difference when no difference in fact exists, in what is known as a *Type I* or *false*

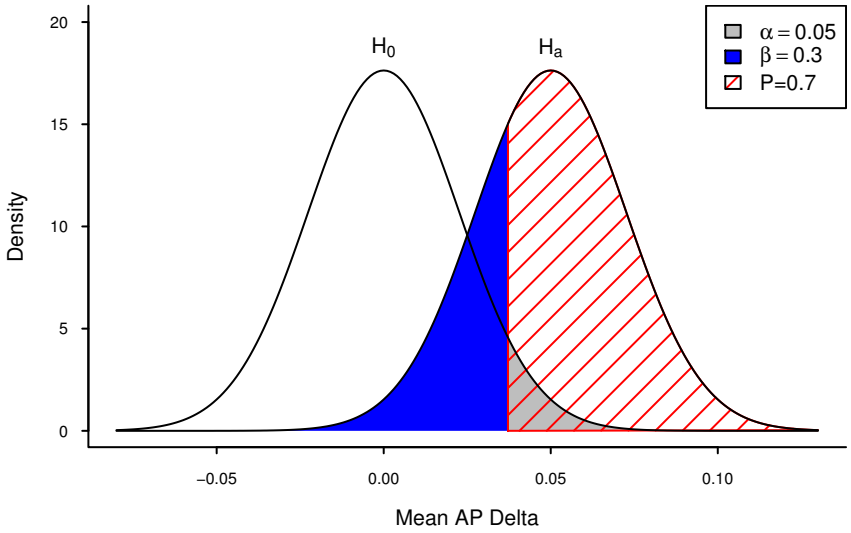


Figure 5.1: False positive rate α, false negative rate β, and power $P = 1 - \beta$, for a true delta δ of 0.05, a standard deviation σ of 0.16, and a sample size of $n = 50$, under a one-tailed, paired t test at significance level $\alpha = 0.05$. Significance will be achieved if the observed mean delta d is 0.0376 or higher and the observed standard deviation s is the same as the population standard deviation σ. The probability β that significance will not be achieved is 0.3, and the power P of the test is therefore 0.7.

Symbol	Description	Effect on power
P	Power of the test	n/a
δ	True or detectable difference	As δ increases, power increases
α	Significance of test	As α increases, power increases
σ	Standard deviation of deltas	As σ increases, power decreases
n	Sample (topic set) size	As n increases, power increases

Table 5.1: Components of statistical power, along with their effect on the power of a test.

positive error. The experimenter can adjust α to reflect how averse they are to this risk. The converse risk, of failing to find significance when a difference between the systems does in fact exist, is termed a *Type II* or *false negative* error, and the probability of it occurring is denoted by β. To calculate a value for β, an alternative hypothesis H_a must be posited, stating a hypothesized difference or true delta δ between the two systems. The obverse of the false negative risk β is the probability $1 - \beta$ of a true positive, given H_a: that is, the probability P that significance will be found if the true difference between the systems is δ. This true positive probability is known as the *power* of a test. Figure 5.1 illustrates the relationship between α, β, and P.

5.1.2 Calculating and predicting power

The power P of a test is determined by several quantities. First is the true score difference δ between the systems under the alternate hypothesis, which becomes the detectable difference under the test. The smaller δ is, the more difficult it is to detect, and, for a given set of topics, the weaker the test. The choice of δ is up to the experimenter; it might be the minimum difference they wish to detect, or the hypothesized result of the experiment they intend. Broadly, we talk of a *consequential* difference. Note that a consequential difference is not necessarily significant, or vice versa, but that the goal of power analysis in experimental design is to ensure that consequential differences are found to be significant. The power of the test also depends on the significance level α, the risk of a false positive: the lower the risk of a false positive, the greater the risk of a false negative. Power further depends on the variability of the per-topic score deltas, as measured by their standard deviation, σ: the greater the variability, the more difficult it is to find significance, and hence the weaker the test. And finally, power depends on the size of the sample, n, which here is the number of topics: the larger the sample, the greater the power. In many experimental situations, including that of retrieval evaluation, only the factor of sample size is directly under the experimenter's control, although other choices (such as the metric used, for instance, or the depth of evaluation) may indirectly affect the standard deviation of the score deltas. These factors are summarized in Table 5.1.

The precise relationship between the different components of the power analysis depends upon the significance test employed. With large samples (for example, 30 or more topics), the t distribution approaches the normal distribution; the power of a large-sample, two-sided, paired t test can be approximated using Φ, the cumulative distribution function (CDF) of the normal distribution, as follows:

$$P \approx \Phi\left(\sqrt{n} \cdot \frac{\delta}{\sigma} - z_{1-(\alpha/2)}\right) \tag{5.1}$$

where $z_{1-(\alpha/2)}$ is the $1-(\alpha/2)$ quantile of the normal CDF (for instance, $z_{1-(\alpha/2)} = 1.96$ for $\alpha = 0.05$); the division by 2 is because this is a two-sided test. We can see from Equation 5.1 that to maintain the same power while halving the detectable delta, or handling twice the standard deviation, requires quadrupling the sample size.

While Equation 5.1 is expressed as calculating the test's power P (that is, the probability of reliably detecting the specified δ), any one of the values in Table 5.1 is determined by specifying the other four. In particular, the experiment designer is often faced with the question of how many sample units (here, topics) are necessary to detect a given score delta. The significance level α and the power P are selected by the experimenter; common values are $\alpha = 0.05$ and $P = 0.8$, implying that the experimenter regards a false positive as being four times as serious as a false negative. The standard deviation σ must also be determined; this is the subject of much of the rest of the chapter. Given these values, the experiment designer is faced with a calculation like that shown in Figure 5.2. If σ is 0.13, for instance, then it takes some 55 topics to reliably detect a score delta of 0.05, but 150 topics to detect a delta of 0.03. If σ is 0.19, however, 150 topics is only sufficient to detect a delta of 0.044. (As we shall see, Figure 5.2 displays σ values representative of AP score deltas.)

The chief problem facing the experiment designer in predicting statistical power using a formula like Equation 5.1 is to estimate σ, the standard deviation of values in the population being sampled. For domains in which the experimental subjects are common and well-studied, this value may be known; the standard deviation of the survival

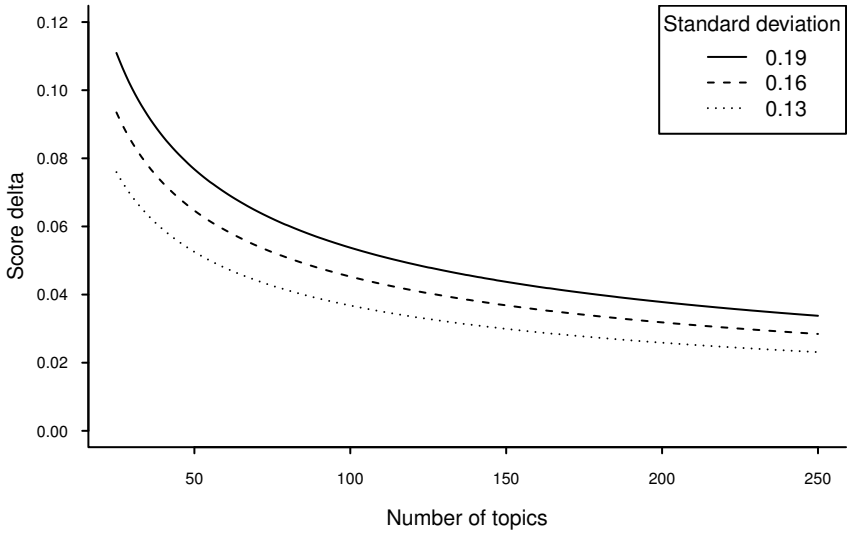


Figure 5.2: Detectable true delta δ, with power 0.8 at significance level $\alpha = 0.05$, as a function of number of topics, for different standard deviations, on a paired, two-sided t test.

time of a certain breed of laboratory mice, for instance. For comparative effectiveness experiments, however, the relevant values are the score deltas between a particular pair of retrieval systems under the chosen metric, and the standard deviation between these two systems will not be known before the experiment is run. Indeed, frequently the comparison will be between a known baseline and a newly developed system, the latter of which may never have been subjected to any properly controlled experimental comparison. Moreover, a competitive baseline would be a reasonable implementation of the current state of the art (although it will be observed in Chapter 8 that competitive baselines are not typical of recent evaluation practice), making the delta in prospect modest, and the question of experimental power acute. In such circumstances, can the standard deviation be reliably predicted from that observed in the past between other system pairs? If not, what method should be used to arrive at an estimation?

5.1.3 Effect size

Instead of quantifying a consequential difference between two systems by raw score delta δ, the experimenter can do so in terms of delta normalized by standard deviation, or *effect size*:

$$\text{ES} = \frac{\delta}{\sigma} \ . \tag{5.2}$$

Normalization by standard deviation makes effect size a unitless metric, or at least one expressed in units of standard deviations, applicable to any experimental population. With significance level α and power P selected, effect size becomes purely a function of the sample size n (refer back to Equation 5.1), no matter what the experimental subjects are. Rules of thumb for generalizing effect size strengths have been proposed. For instance, Cohen (1988) tentatively classifies an effect size of 0.8 as representing a *large* effect, of 0.5 a *medium* effect, and of 0.2 a *small* effect.

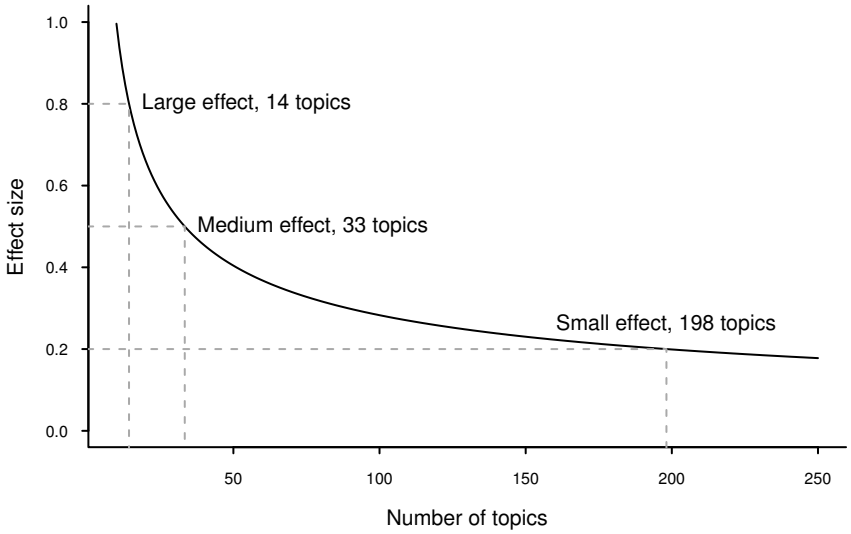


Figure 5.3: Effect size as a function of number of topics. Effect size is defined as δ/σ, the true mean score delta, normalized by the true score delta standard deviation. The significance level α is set to 0.05, and power P is set to 0.8. The number of topics necessary to reliably detect the rule-of-thumb levels (proposed by Cohen (1988)) of a large, a medium, and a small effect are marked.

Given standard values of $\alpha = 0.05$ and $P = 0.8$, Figure 5.3 displays effect size as a function of sample size. From this we can see that a standard, 50-topic TREC test collection is sufficient to reliably detect a medium effect, but 200 topics would be needed to reliably detect a small effect. Cohen's classifications of effect size are, however, only rough guides, as he himself acknowledges, and what counts as a consequential effect still depends on the experimental context and metric. For instance, as technology matures, we might expect the score deltas between baseline and innovative systems to narrow, but whether the standard deviations of these deltas would also narrow in proportion is not obvious. The experimenter can avoid the technical necessity of estimating σ by expressing the hypothesized consequential effect in terms of effect size; but a principled choice of a consequential effect size still requires an estimation (formal or otherwise) of the relationship the between score deltas and the standard deviations of the systems under comparison. In this chapter, we assume that the researcher is quantifying effect in absolute terms, and therefore needs to estimate population standard deviation during the experimental design phase.

5.2 The power of TREC collections

The natural first choice of an IR experimenter is to use an existing test collection, such as those created by TREC and similar efforts. In this section, we examine the power of the TREC collections, as observed on the TREC participant systems. We have already seen that, under a common rule-of-thumb, TREC collections are not large enough to detect small effects. The purpose of the current section is to determine whether these collections are sufficiently powerful to detect achievable improvements in a mature technology.

Test Set	AP delta σ		Detectable δ	
	Median	95%	Median	95%
TREC 3 AdHoc	0.147	0.198	0.059	0.080
TREC 4 AdHoc	0.173	0.220	0.070	0.089
TREC 5 AdHoc	0.170	0.241	0.069	0.097
TREC 6 AdHoc	0.199	0.259	0.080	0.105
TREC 7 AdHoc	0.151	0.207	0.061	0.084
TREC 8 AdHoc	0.159	0.226	0.064	0.091
TREC 9 Web	0.170	0.225	0.069	0.091
TREC 2001 Web	0.141	0.202	0.057	0.081
TREC 2004 TB	0.135	0.185	0.055	0.075
TREC 2005 TB	0.143	0.191	0.058	0.077
Average	0.159	0.215	0.064	0.087

Table 5.2: Median and 95th percentile of standard deviation of per-topic, between-system AP score deltas, for different TREC tracks, across all systems that participated in each track. The two right-hand columns show the minimum true AP delta detectable with power $P = 0.8$ and significance level $\alpha = 0.05$ using 50 topics given these standard deviations.

Once the significance level, power, and number of topics has been set, the detectable delta under a power analysis depends on the standard deviation, σ, of the per-topic score deltas between the pair of systems being compared. The value of σ varies for different system pairs, but for each particular TREC runset, the distribution of σ values can be empirically observed, for the set of topics included in the collection. Table 5.2 gives the median of these σ values under the AP metric for the participant systems from several tracks of TREC. These medians differ noticeably from one TREC to the next, but the overall average is around 0.16. For a standard deviation σ of this size, a collection of 50 topics can reliably ($P = 0.8$) detect a true δ of around 0.064. The largest usable topic set, that of the TREC 2004 Robust track, has 249 queries; this is enough to detect a δ of 0.028 for a σ of 0.16. But this is only the mean σ value for these datasets. The 95th percentile value, which might be considered worst case, averages 0.215, making only a δ of 0.087 reliably detectable with 50 queries, or of 0.038 with 249. Such score deltas are frequent enough across the full runset. For instance, the median delta between TREC 8 systems under AP is 0.082; 57% of observed score deltas on this runset would be reliably detectable, using only 50 topics, with the average-case σ of 0.16, and 48% even with the worst-case σ of 0.215. But these are gross figures, across a runset that includes hand-crafted manual runs at one end and faulty runs at the other; as will be examined shortly, incremental improvements on reasonable baselines are likely to produce much smaller score deltas.

A natural question is whether σ varies with the mean delta. One might suppose, for instance, that system pairs that have smaller mean deltas would tend to have smaller standard deviations; that is, that smaller improvements are (in absolute terms) less variable. This is an important consideration, because it would mean that incremental improvements are easier to find significance for than the overall mean σ estimate suggests. Figure 5.4 graphs the relationship between σ and mean delta for the TREC 2004 Robust runset. The figure shows that, for this data set at least, there is a rela-

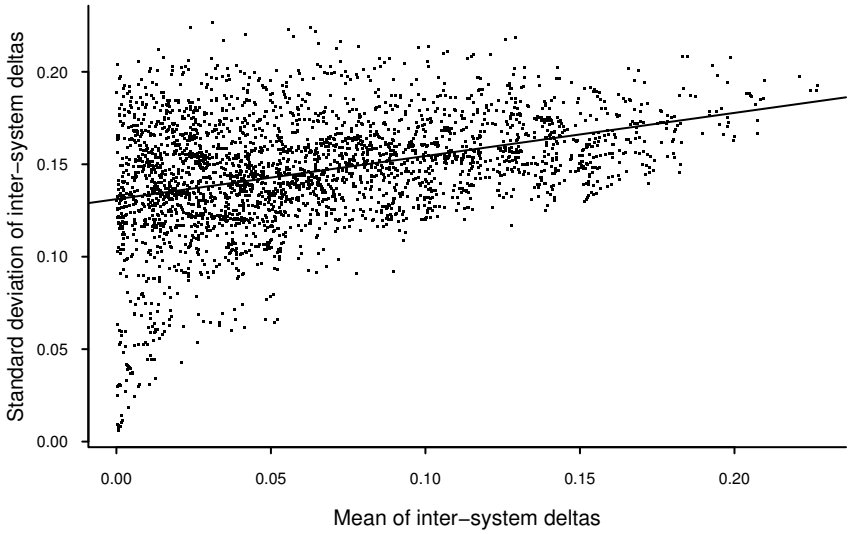


Figure 5.4: Relationship between standard deviation and mean of between-system per-topic AP scores deltas on the TREC 2004 Robust Track systems. Each dot represents the mean and standard deviation of the AP score deltas on Topics 301–450 between a pair of non-description-only TREC 2004 Robust systems. The line of best fit is also drawn; it has intercept 0.131 and slope 0.233.

tionship between the mean delta and σ, but only a slight one. At mean deltas close to 0, the average σ is around 0.13; at a mean delta of 0.1—a relatively large difference between systems—the average standard deviation is around 0.15. Of course, particular incremental improvements in real experiments may have lower standard deviations, for instance if they only affect a small number of topics; but in general, it cannot be assumed that small improvements will lead to much lower variability than large ones.

Rather than the gross comparisons across a full TREC runset, a more realistic setup for a laboratory experiment is one in which the researcher is comparing their (hopefully) improved experimental system against a reasonably strong baseline. One way of approximating this baseline–experimental comparison on the TREC runsets is to treat second-quartile systems as candidate baselines, and for each baseline, any of the systems scoring higher than it as an experimental system; only cases in which experimental systems outscore baselines are considered. It is also necessary to reconsider σ estimates for baseline–experimental pairs; it may be that these pairs are less variable than are all system pairs as a group. Taking the TREC 8 dataset as an example, the median σ for baseline–experimental pairs is 0.138, which is slightly less than the overall mean of 0.159, although the 95th percentile is hardly changed at 0.221. Average score deltas, however, are much lower for baseline–experimental pairs than for all system pairs as a whole, as is to be expected, given that only top-half systems are being considered. The median of the mean score deltas for baseline–experimental pairs is 0.032, compared to 0.082 for all systems. Only 23% of these baseline–experimental pairs are reliably detectable in the mean case, and only 14% in the worst case (95th percentile σ). To reliably detect the median baseline–experimental δ of 0.032, given a median σ of 0.138, requires 145 topics. Table 5.3 gives figures for other TREC test sets. There is a fair amount of variability in these results, but in most cases a topic

Test Set	AP delta		# Topics Required
	Mean	σ	
TREC 3 AdHoc	0.051	0.136	58
TREC 4 AdHoc	0.034	0.165	188
TREC 5 AdHoc	0.045	0.171	118
TREC 6 AdHoc	0.042	0.207	192
TREC 7 AdHoc	0.038	0.149	120
TREC 8 AdHoc	0.032	0.136	145
TREC 9 Web	0.033	0.166	197
TREC 2001 Web	0.020	0.138	375
TREC 2004 TB	0.047	0.125	57
TREC 2005 TB	0.033	0.113	95

Table 5.3: Median of score delta mean and standard deviation for baseline–experimental system pairs across different TRECs under the AP metric, and the number of topics required to achieve power of 0.8 at significance level 0.05 on a two-tailed, paired t test given the median delta and standard deviation. A baseline system is a system in the second quartile by mean AP score; an experimental system is any system that scored better than a baseline system.

set of at least 100 topics, and in some cases closer to 200 topics, is necessary to reliably distinguish a competitive, second quartile baseline system from a representative, better-than-baseline experimental system.

The preceding analysis consists of only an approximation of what might be the delta and deviation characteristics of a realistic research baseline–experimental comparison. Nevertheless, it strongly suggests that the 50-topic TREC collections are simply not big enough to reliably detect the kind of incremental improvements that one should expect to see in a well-established technology such as information retrieval. It is interesting to consider what effect this has had upon the recent development of technology in the research arena; some evidence for this will be examined in Section 8.2.2. At the very least, experimenters should seek to aggregate as many topic sets as possible over the one corpus, as is done with the 249-topic TREC 2004 Robust collection.

5.3 Estimating delta deviation

The previous section analyzed the statistical power of TREC collections, concluding that the 50 topics they typically contain is insufficient to reliably detect incremental improvements in system effectiveness. In this section, we turn to examining how the experiment designer should determine the size of the topic set that is required for reliable experiments. As mentioned before, the choice relies primarily on the estimation of the standard deviation of score deltas between (previously uncompared) retrieval systems. We discuss three ways in which this estimation can be made. The first (Section 5.3.1) is based upon previous performance and knowledge; the second (Section 5.3.2) is through the use of a trial experiment; and the third (Section 5.3.3) is by iteratively increasing the sample size, updating the estimation of standard deviation at each iteration.

The main data used in this section is the set of participant runs from the TREC 2004 Robust Track, over Topics 301–450, excluding the description-only runs. Topics 601–

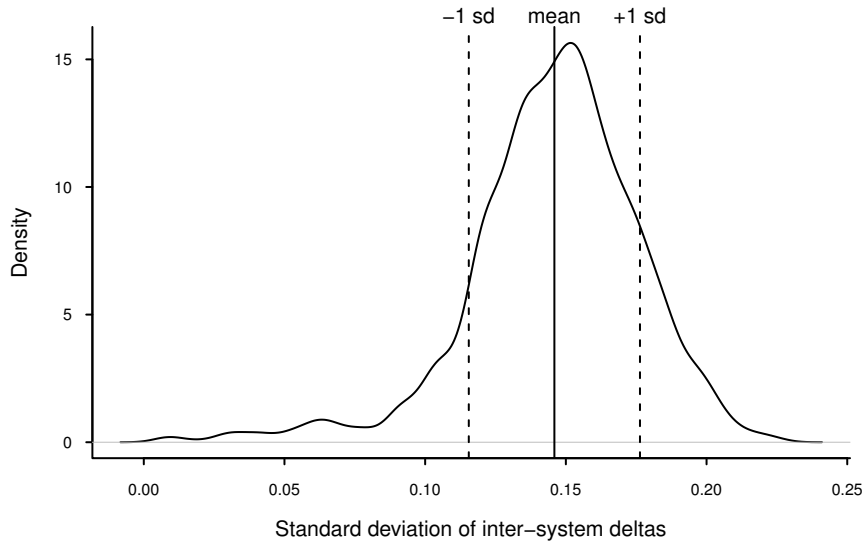


Figure 5.5: Distribution of standard deviations of between-system per-topic AP score deltas, for the TREC 2004 Robust Track runs. Plotted are the 3,003 standard deviation values for the AP score deltas between each of the 3,003 pairs produced by the 78 TREC 2004 Robust Track non-description-only runs on Topics 301–450. The mean and standard deviation are shown. The standard deviation of this distribution (of standard deviations) is 0.030.

700, from the TREC 2003 and TREC 2004 Robust sub-collections, are excluded in order to maximize topic homogeneity; as was observed in Section 4.6.3, these topics have smaller and less diverse pools. The description-only runs are removed from the runset because the description fields of Topics 301–350 (taken from the TREC 6 AdHoc collection) do not contain all topic keywords, leading to anomalously poor performance for description-only runs on these topics.

5.3.1 Based on previous experience

Faced with the task of design-time power analysis, and needing to estimate the likely standard deviation σ of score deltas between a baseline and an experimental system, the experimenter might first turn to past experience as a guide. What exactly constitutes past experience is difficult to quantify, and will vary between different experimental setups. There are, however, two broad questions. First, is there a single standard deviation of score deltas for a metric, shared by all system pairs, than can be applied in each experimental environment—or at least a close approximation to it? And second, if there is no single standard deviation, then how wide are the bounds that past experience might set upon the deviation likely to be observed in a proposed experiment?

First is the question of whether all system pairs share the same (or approximately the same) σ of score deltas—that is of whether there is, for instance, a single σ of AP score deltas. It was observed in Table 5.2 that deviations vary both within and between TREC data sets. Figure 5.5 gives the full distribution of AP delta σ between each of the 3,003 non-description-only system pairs in the TREC 2004 Robust Track, again showing that σ is by no means the same for every pair of systems. This does not in

Algorithm 5.1 Determine p value of observed σ dispersion, and mean resampled dispersion

$D = \{d_{ijk}\}$, an array of score deltas, where d_{ijk} is the delta between the ith and the jth system on the kth topic; n_s is the number of systems, n_t the number of topics, and R the number of times to repeat sampling
$n_p \leftarrow (n_s(n_s-1))/2$ ▷ number of system pairs
$\sigma \leftarrow \mathrm{sd}(\{\mathrm{sd}(d_{ij*}) : i > j\})$ ▷ dispersion of observed distribution
$P \leftarrow \{d_{ijk} - \bar{d}_{ij*} : i > j\}$ ▷ create pool of mean-adjusted deltas
$\Sigma^* \leftarrow \{\}$
for $r \in \{1, \ldots, R\}$ **do**
  $V_r \leftarrow \{\}$
  for $s \in \{1, \ldots, n_p\}$ **do**
    $V_r \leftarrow V_r \cup \mathrm{sd}(\mathrm{sample}(P, n_t))$ ▷ sd of sample of virtual system pair
  end for
  $\Sigma^* \leftarrow \Sigma^* \cup \mathrm{sd}(V_r)$
end for
$p \leftarrow |\{\sigma^* \in \Sigma^* : \sigma^* > \sigma\}| / R$ ▷ p value is proportion resamples $>$ original
return $\overline{\Sigma}^*, p$

itself disprove the hypothesis of a common population σ, because that deviation would apply across the entire (nominal) population of topics, and in each of the above cases what is observed is only the deviation on a sample. Just as different samples from the one population can give different means, so too they can give different standard deviations. But testing the null hypothesis that different system pairs have the same AP delta σ, using the resampling method described in Algorithm 5.1, rejects this null hypothesis with confidence $\alpha = 0.001$. The mean dispersion of the resampled standard deviations, under the hypothesis of common deviations, is 0.015, compared to the observed dispersion of 0.030 in Figure 5.5. We can definitely conclude that different system pairs have significantly different score delta deviations. There is no single σ of AP score deltas for the experimental designer to rely on.

Given that the score delta σ is significantly different between different system pairs, the experiment designer can only rely on past experience to provide at best a distribution over possible σ values for a new pair of systems that are to be compared. The nature of this distribution, and the confidence that the designer can place in it, will vary from situation to situation. Nevertheless, the TREC datasets can be used as a reasonable example of what such an approach might entail. Consider an experiment designer who is testing a new system against a TREC run, on a TREC collection—or, more generally, an experimenter working in an environment that provides a similar (rather high) degree of past information about score deltas. The designer could then take the median σ observed on past system pairs in that data set as their estimate. Table 5.2 indicates that this is useful information, since the median does change from test set to test set. But the differences between test sets are relatively small compared to the variability of σ within a test set. From Table 5.2, the 95th percentile σ is roughly 35% higher than the median. Thus, if the designer were to choose a topic set size based on the median σ, they would run a substantial risk that the achieved power was well below the desired level. On the other hand, based on Equation 5.1, handling the pessimistic case of a 95th percentile σ, roughly 35% higher than the median, requires 80% more topics. That is to say, the high variability of score delta σ means that the experimenter faces a

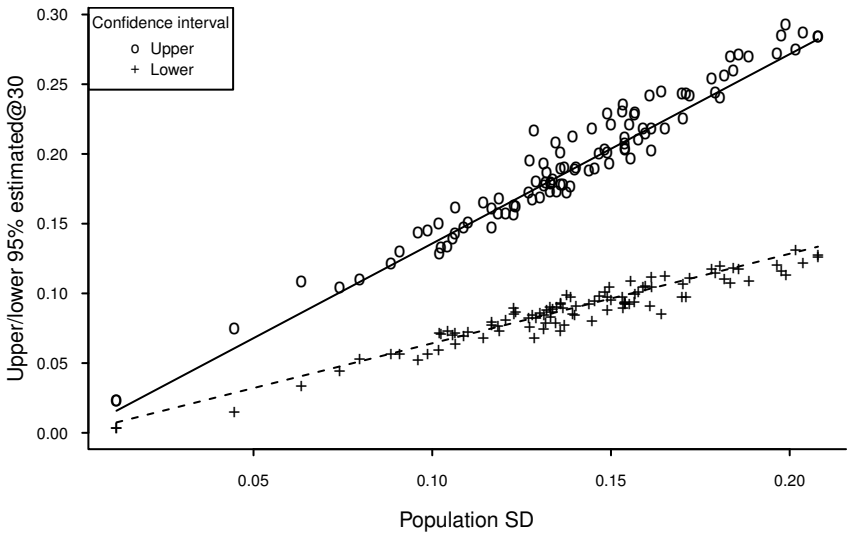


Figure 5.6: Theoretical (line) and empirical (point) 95% confidence interval, estimating score delta σ from a sample of 30 topics, based on the TREC 2004 Robust systems. The metric is AP.

dilemma: aim for the mean and have a high risk of insufficient power; or substantially remove that risk but expend almost twice as much assessment effort as, on average, will eventually turn out to have been necessary.

5.3.2 Based on trial experiments

Past experience being an unreliable guide, the experiment designer might turn next to the use of a trial experiment. The idea behind trial experiments is to use a small sample to estimate features of the subject population, and then use these estimates to design the full experiment. The main estimate of interest is of the population σ, but the designer might also use the trial experiment to gauge the range of the delta, and of course also to try out other elements of the experimental design; these later issues will not be considered further here. We are concerned in particular with trial experiments in which the topics used in the trial are not re-used in the full experiment; re-using topics leads to a situation similar to the iterative approach discussed in Section 5.3.3.

When designing a trial experiment, one must consider, first, how many topics to include in the trial, and second, how to use the estimate of σ that the trial produces. The fewer the trial topics, the cheaper the trial, but the less reliable the estimate; conversely, increasing the number of topics in the trial makes the estimate more accurate, but the trial more expensive. Assuming the trial topics are sampled from the same population as the full experiment, and that other experimental parameters are the same, then the single most likely estimate of the standard deviation of the population will be that observed on the sample in the trial. This estimate will, however, have a standard error to it, which can be derived from the observed variability of the sample and from the size of the trial experiment. Taking the mean as the estimate is risky; taking a higher percentile of the sampling distribution is safer, but results in a more expensive subsequent experiment.

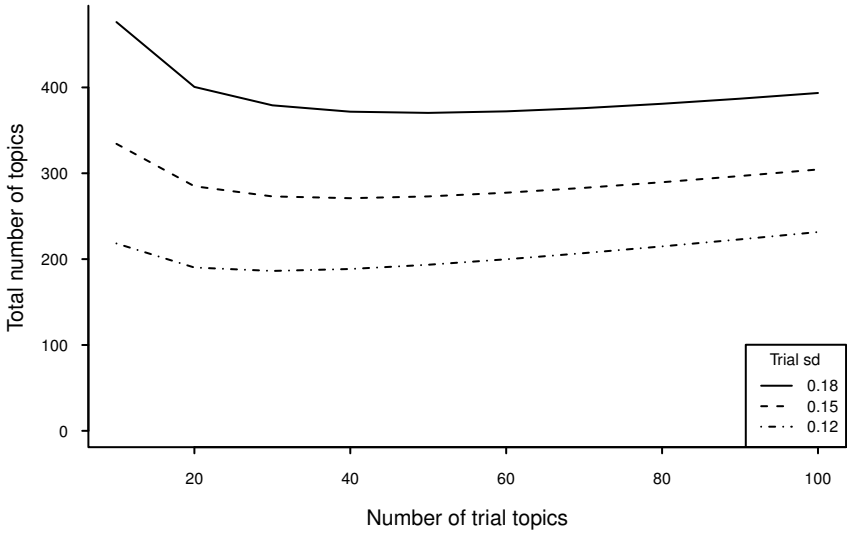


Figure 5.7: Total number of topics assessed, following trial experiments of different sizes, which produce different estimates of standard deviation, where the final experiment requires power of at least 0.8 on $\delta = 0.033$ at $\alpha = 0.05$, with 95% confidence.

An expression for the likely error in an estimate of population standard deviation taken from a sample, such as that used in a trial experiment, can be derived as follows. Assume that the sample, of size n, is drawn from a normal population with mean μ and variance σ^2; let s denote the standard deviation of the sample. It can be shown that the sample standard deviation is, for large enough n, roughly normally distributed, with standard error of approximately (Kirkup and Frenkel, 2006):

$$\sigma_S = \frac{\sigma}{\sqrt{2n}} \,. \tag{5.3}$$

In practice, metric deltas are not normally distributed; moreover, trial experiments are potentially not large enough to justify the normal approximation of the sampling distribution. Figure 5.6 shows the theoretical 95% confidence intervals given by Equation 5.3, and the empirical quantiles on the TREC 2004 Robust Track systems, for samples of 30 topics. The figure indicates that the formula underestimates the variance of the estimator. Nevertheless, Equation 5.3 is useful to inform our discussion.

Consider an experiment designer who has run a trial experiment, derived an estimate of σ, and wishes to choose a topic set size for the full experiment that will cover the 95th percentile of the estimate's distribution—the same degree of conservatism posited for the designer working from previous experience. Under the normal distribution, the 95th percentile of values falls 1.64 standard deviations above the mean. Therefore, based on Equation 5.3, for a trial experiment of (say) 20 topics, the designer would take an estimate around 25% greater than the observed standard deviation of the trial. This is slightly less than the 35% margin of error that the TREC runsets required; but then, the experimenter has already spent 20 topics to arrive at that estimate.

The margin of error from the trial experiment, and therefore the cost of the full experiment, can be decreased by increasing the trial's size. From Equation 5.3, the error can be halved by quadrupling the trial topic set. After a point, however, the

additional cost of the trial experiment outgrows the savings on the full experiment. Figure 5.7 shows the total number of topics assessed in a conservative (95% coverage) experimental setup, for different trial sizes and estimates of standard deviation. The optimal trial size itself depends on the standard deviation of the population—a circular problem—but falls in the range of 20 to 50 topics for the σ observed for average precision deltas. Beyond this many topics, the increased accuracy of the estimate does not justify the additional cost. For each actual standard deviation, the minimum full topic set size needed for confidence in achieving post-hoc power is 60% to 80% greater than that needed in the average case. For instance, for a mean σ estimate of 0.15, the optimal trial size is 40 topics, and gives an 95th percentile σ estimate of 0.178. To cover this percentile requires an experimental set of 231 topics, which, when added to the trial topics, makes a total expense of 271 topics. In the mean, however, a σ of 0.15 only requires 164 topics to achieve the desired power. The conservative approach uses 65% more topics than will, on average, prove necessary. It was observed in Section 5.3.1 that basing estimates on previous experience, at least as represented by a TREC test set, requires an 80% overestimate on the topic set size. Therefore, the trial experiment approach achieves only slightly better efficiency. It does, however, have the advantages that it does not depend on the availability and reliability of previous experience, and that the trial can yield other useful information about the experimental setup.

In the case that the corpus of an existing collection is suitable for an evaluation experiment, but not its topics (if only because the topic set is too small), it is possible to use the existing topics and qrels for the trial experiment, and develop and assess new topics for the full experiment. Re-using the existing topics for the trial leads to an obvious saving of effort; and, as has been observed above, the fifty topics of a typical TREC collection should be enough to give a reasonable estimate of score delta standard deviation. The experimenter does not, however, know how representative trial conditions are of the full experiment, since the trial and full topics are not derived from the same source, nor their assessments. The reliability of the trial experiment is even less assured if the full experiment is to be performed on a different corpus as well. Nevertheless, as a pragmatic step, performing trials on one or more existing collections, even if imperfectly aligned with the final test environment, is likely to provide valuable insight into comparative system behaviour for the full experimental design.

5.3.3 Based on iterative estimation

A conservative choice of topic set size, based either on past experience or on a trial experiment, leads to far more topics being used and assessed than in the mean case will prove to have been necessary, because the experiment designer is faced with a highly variable estimate of standard deviation, and, to be safe, must choose a high percentile. The preceding discussion, however, has implicitly assumed that the choice of topic set size must be made once, and the full experiment (and no more) be run based on that choice. In many fields of research, this assumption is true; it may not be possible, for practical or theoretical reasons, to add new subjects to a sample if it proves to be too small—or to cut an experiment short if the planned sample size turns out to be excessive. But this constraint does not, at least at first sight, seem to apply to comparative retrieval experiments. It would seem admissible to start with a small topic set, and keep increasing it until the desired experimental power is achieved.

A method for the iterative estimation of experimental power is described in Algorithm 5.2. The method is simple. The consequential score delta which needs to be reliably detectable is specified at the start of the experiment; the experiment will con-

tinue until sufficient power to detect this delta is achieved. A topic at a time is added to the topic set (in practice, one would start with at least the minimum number of topics required for the significance test to be meaningful) . The documents returned for that topic by the systems under comparison are assessed for relevance, and the systems are evaluated against the topic. The σ of score deltas is determined, and from that the current power of the experiment. If this power is sufficient to detect the specified delta, then the experiment is complete, and a final evaluation and significance test is performed; if power is insufficient, another topic is added, and power is tested again.

The great advantage of the method presented in Algorithm 5.2 is its efficiency: the desired experimental power is precisely achieved with a minimal number of topics. This is in contrast to the previously discussed design methods, which choose the topic set size once, and therefore need to make a high estimate, to reduce the risk of failing to achieve power. Even then, the once-off estimation methods are not guaranteed to achieve the desired power, whereas the iterative method can be continued until this power is achieved.

It is important to note that the goal of Algorithm 5.2 is to achieve *power*, not *significance*. The method would be seriously flawed if, in addition to power, significance was checked for after each topic had been added, and the iterations halted if significance was achieved before power. Such an approach leads to a bias in favour of finding significance; any prefix of topics that achieves significance will lead to a significant result, even if the full topic set (the set which achieves the desired power) does not. This bias can be observed empirically. Take the TREC 2004 Robust system pairs where a comparative evaluation under AP over Topics 301–450 leads to a p value between 0.05 and 0.10; that is, system pairs that do not quite achieve significance on the full topic set. There are 137 such system pairs. If an iterative topic inclusion method is used, and significance is checked after every topic subset from 50 topics on, then on one random trial almost two-thirds (89) of these system pairs are found significant at level $\alpha = 0.05$ on at least one of the topic prefixes. In other words, the possibility of false positives is greatly increased by such repeated testing, although the methods of sequential analysis (Mukhopadhyay and de Silva, 2009; Siegmund, 1985) could be deployed to compensate for this bias.

Testing significance under the iterative method after each topic is added to the topic set is clearly biased; but testing power, through re-estimating score delta σ, might not appear to be. Repeatedly testing for power is, however, subject to a similar, although more subtle, form of bias. The stopping condition of the iterations is when power is achieved, which means in effect when the current number of topics is sufficient for the existing standard deviation estimate. This means that sampled topic sequences that happen, by chance, to have lower standard deviations, will achieved power earlier, and hence result in smaller topic sets. Conversely, sequences with higher standard

Algorithm 5.2 Iterative sampling

Input: δ, the target detectable true delta
$d \leftarrow \infty, T \leftarrow \{\}$
while $d > \delta$ **do**
    $T \leftarrow T \cup \{sample(\mathcal{T})\}$
    $d \leftarrow$ calcDetect(T)
end while
Perform significance test

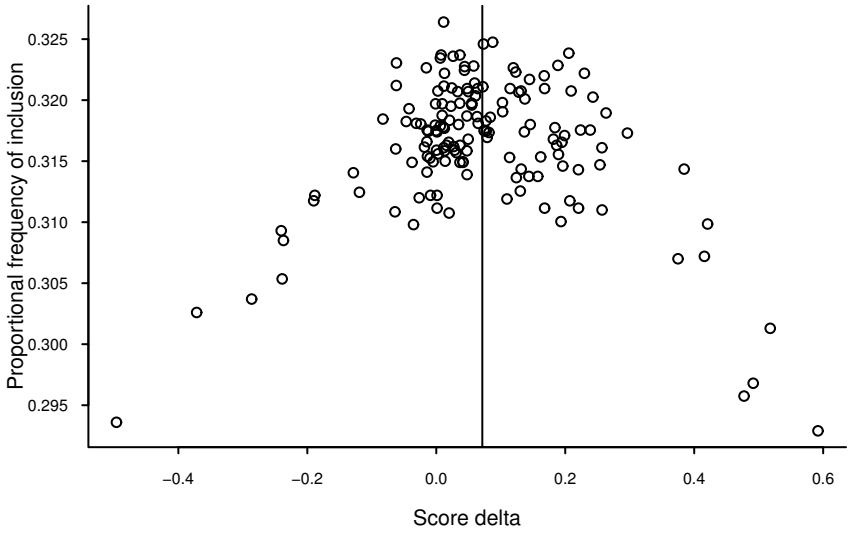


Figure 5.8: Proportional frequency of inclusion against per-topic delta for Topics 301–450 comparing `fub04Tge` and `polyutp1` from the TREC 2004 Robust Track, as averaged over $20{,}000$ random trials. Each trial randomly samples topics without replacement until a test power of 0.8 for a true delta of 0.06 is achieved. The mean delta is marked with a vertical line.

deviations will continue for longer. Lower deviations occur when topics have score deltas more similar to each other, and so are (in general) more typical of the population as a whole. Therefore, although the sampling probability is uniform across topics, the probability that a given topic will be included in a topic set is higher for topics whose score delta is closer to the norm for the population.

The inclusion bias for topics under iterative sampling is demonstrated empirically in Figure 5.8. The iterative power estimation method is repeatedly employed to achieve a specified degree of power in the comparison of two TREC 2004 Robust Track systems, and the proportion of topic sets that each topic is included in is recorded. Different topics have significantly (at level $\alpha = 0.01$ in a χ^2 test on proportions) different chances of being included in an iteratively sample topic set. Those topics whose score deltas are more typical of the population have a higher likelihood of inclusion, whereas atypical topics have a much lower one.

If topics with more typical deltas are more likely to be included in an iteratively-sampled topic set, then that sampled topic set is likely to have a lower standard deviation than the population of topics as a whole. That is, iterative sampling is biased towards underestimating standard deviation. On the other hand, at least where delta distributions are balanced, and the typical delta values cluster around the mean delta (as in Figure 5.8), then the estimation of the mean will not be (noticeably) biased.

The extent of the bias in the estimation of σ depends on the distribution of the population. Figure 5.9 shows the empirical bias observed on actual TREC 2004 Robust Track system pairs. To simulate a baseline–experimental comparison, a second quartile run is randomly selected as the baseline, then a different run from the top three quartiles as the experimental system. Note that this differs from the method of choosing baseline–experimental pairs used in Section 5.2, in which the experimental system

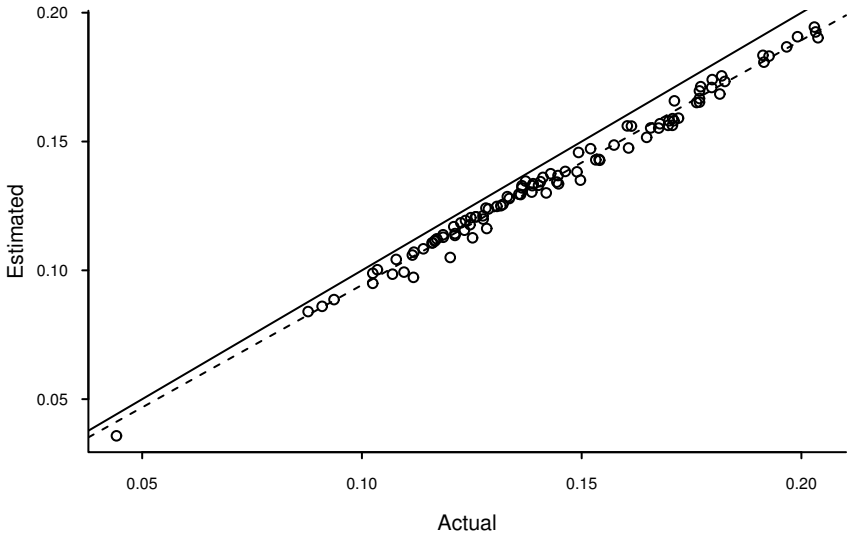


Figure 5.9: Mean σ estimates using the iterative sampling method, compared to actual σ, for AP score deltas between 100 randomly selected baseline and experimental system pairs, drawn from the TREC 2004 Robust Track systems, on Topics 301–450. Baseline systems are sampled from the second quartile of runs by AP, while experimental systems are sample from the top three quartiles. The dotted line is the line of best fit, which has slope 0.965. The solid line marks estimated $=$ actual.

was always superior to baseline: then, we were testing to see what real improvements could be missed; now, we relax the restriction on experimental systems, in order to test rates on two-tailed significance. The iterative power estimation method is employed until estimated power, on the sampled topics, is equal to the mean power for that system pair over 100 topics. This is done repeatedly, and the mean estimate of σ noted. Topics are sampled with replacement to simulate sampling from an infinite population. The mean σ estimates of the iterative method understate the true standard deviations by 3.5% on average. Increasing the sampling step size decreases bias only slightly; adding 40 topics to the topic set per iteration leads to an average underestimate of 2.8%. The estimates of the mean, though, are unbiased.

Underestimating the standard deviation of the population under the iterative topic sampling method leads to experimental topic sets with higher apparent power that pure random sampling would. Fairly estimating mean deltas at the same time means that the experiments are biased towards achieving statistical significance. The bias towards false positives in significance can be empirically demonstrated. Again, baseline–experimental pairs are randomly selected. But this time, the raw score deltas are translated by subtracting the mean score delta for each system pair. This shifts the translated mean score delta to 0, which simulates the null hypothesis of no mean difference. Topics are sampled with replacement, to simulate sampling from an infinite population. Iterative sampling is performed until power is achieved, and the proportion of the topic samples that achieve (false) significance is recorded. Uniform random sampling of topic sets of the same size is also performed, and false positives noted on these sets. This allows us to compare the false positive rates from the uniform random and iterative sampling methods.

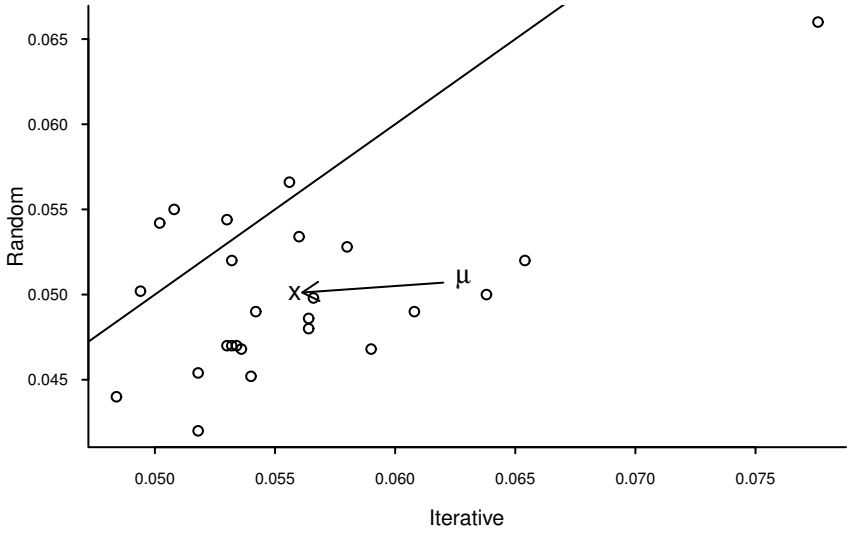


Figure 5.10: Proportion of false significance readings for iteratively sampled topics compared to uniform randomly sampled topics. Each point compares the false significance rate of these methods, on one of 25 randomly sampled baseline–experimental system pairs. The system pairs are sampled from the TREC 2004 AdHoc Track. The per-topic score deltas of each pair are shifted to a mean delta of 0. A total of $5{,}000$ samples are performed per pair. The baseline system is sampled from the second quartile, the experimental from the top three quartiles.

The empirical false positive rates of the random and iterative sampling methods are shown in Figure 5.10, for 25 baseline–experimental system pairs drawn from the TREC 2004 Robust Track runset. The mean false positive rate for true random sampling is 0.0501, almost precisely the expected false positive rate under the $\alpha = 0.05$ significance level. For iterative sampling, however, the mean false positive rate is some 10% higher, at 0.0558. Moreover, the false positive rate for iterative sampling is higher than that for true random sampling in 20 of the 25 system pairs. A two-tailed paired Wilcoxon test finds these differences significant at level 0.001.

The iterative estimation method, therefore, while efficient in its use of topics and easy to implement, leads to a bias in favour of both experimental power and significance. Our experiments suggest, though, that the degree of this bias is slight. On the dataset employed, and with the experimental parameters used, the bias in p values is around 10%; it may be different for other contexts.

5.3.4 Suggested methodology

Faced with a pair of systems to evaluate on new topics, a researcher rich in relevance assessment resources and armed either with strong previous experience or the results of a trial experiment can proceed to make a single, conservative estimate of σ and assess the full (and generally large) set of topics necessary to be confident of achieving the desired power. A poorer but theoretically fastidious and temperamentally stoic researcher might take an average estimate and risk variability turning out to exceed that estimate and rendering the experiment inconclusive. Or the researcher might abandon

absolute measures and express consequential effect in terms of effect size, with its attendant limitations and vagueness. Any of these approaches will enable the statistical significance of the experiment's results to be tested and reported with the minimum of caveats.

However, for a researcher with scarce assessment resources who wishes to quantify consequential effect in absolute terms and is (understandably) unwilling to risk an inconclusive experiment, we suggest a hybrid approach. This method constitutes a pragmatic form of sequential analysis (Mukhopadhyay and de Silva, 2009; Siegmund, 1985), though one aimed at the intermediate target of statistical power, rather than the ultimate target of statistical significance. The predicted or consequential δ must be stated at the outset. An initial best (non-conservative) estimate of σ should be made, either through experience and a judgment of the likely similarity of the two systems, or using a trial experiment (whose topics, under the hybrid method, can be reused). The indicated number of topics should then be assessed, and the systems evaluated. If desired power has not been achieved, then σ should be re-estimated as the observed sample standard deviation, and the indicated number of additional topics assessed and evaluated. (Observed standard deviation is likely to be an overestimate of population σ, since the only reason we are observing it is that it is higher, possibly by chance, than our initial estimate; however, a slightly conservative estimate here is desirable to reduce the number of iterations and hence the potential for bias.) This process is repeated until power is achieved. Then, and only then, significance can be tested for.

The proposed methodology is assessment-thrifty and guaranteed to obtain the desired power. The downside is that the reported significance is likely to be slightly exaggerated. Naturally, the researcher needs to report this fact, and also that the exact degree of bias is uncertain. The researcher further needs to state the experimental methodology employed, including the δ used to calculate power, the initial topic set size, and the number of iterations. This must be reported even if power is achieved by the initial topic set, without the need for further iterations. The only reason in such a case that there were no further iterations is because power was achieved; the subsequent significance test is not independent of this methodological choice, and will be (mildly) biased.

5.4 Evaluation depth

We have so far dealt with power analysis as a tool for deciding the topic set size required for an experiment. It also has a useful role to play in analyzing other features of the experimental setup, such as the choice of metric and metric parameters. The choice of metric affects both the consequential deltas that the experimenter might expect, and also the variability that the per-topic deltas might display. What is desirable is that the ratio between these (that is, the effect size) is as large as possible; this means that less effort is required to achieve significance. Other researchers have quantified the effort to achieve significance in terms of discriminative power, by calculating the proportion of system pairs that are actually found significant in group experiments such as TREC (Sakai, 2006); the significance rate, however, depends on the effect size, and this latter value and its components, mean and standard deviation, are more amenable to analysis.

One of the most important metric parameters is the depth to which evaluation is performed (or, for sampling-based methods, the number of samples made for each topic). Greater depth of evaluation leads presumably to more stability, and hence a stronger effect; but it also requires more relevance assessments, and relevance assessments are

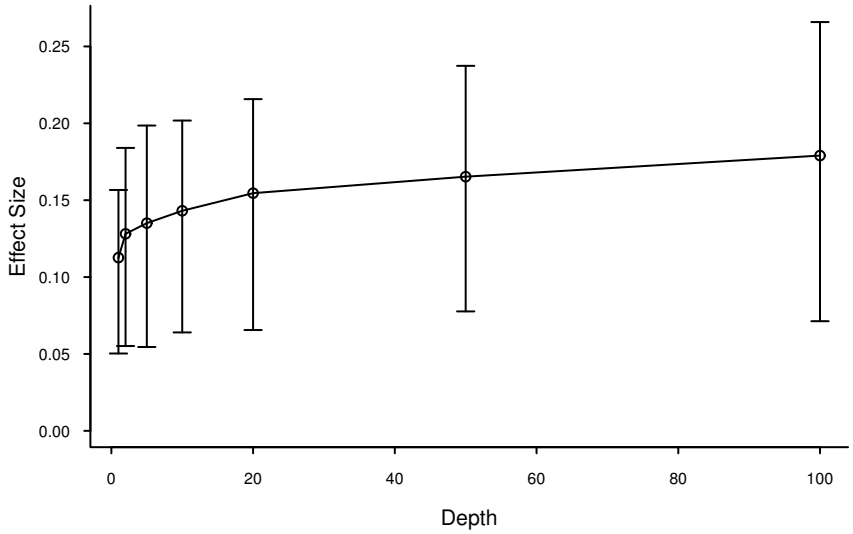


Figure 5.11: Observed effect size of AP at different evaluation depths. The mean effect size along with the inter-quartile range is shown. A total of 1,000 baseline–experimental system pairs are randomly selected from the TREC 2004 Robust Track systems, and observed effect size calculated at each evaluation depth, with full pooling to that depth (based on the official qrels). Baseline systems are sampled from the second quartile, experimental systems from the top three quartiles. For each system pair, AP scores are calculated to the specified depth, estimating R only from the known relevant documents found by the two systems to that depth.

the most expensive element in test collection formation. It may be more efficient overall to evaluate less deeply, but have more topics, particularly for collections that are purpose-built for a particular experiment. Power analysis permits us to quantify this trade-off. In particular, the measure of *observed effect size* will be used; this is, for a pair of retrieval systems, the mean delta divided by the delta standard deviation across a set of topics.

The first question to answer is how observed effect size changes with depth of evaluation. Figure 5.11 shows the range of observed effect sizes for different depths under the AP metric, across a sample of TREC 2004 baseline–experimental system pairs. As evaluation depth is increased, score deltas do become more consistent, leading to a rise in observed effect size, and therefore in experimental power and likelihood of finding significance. The effect, though, is only slight.

The number of documents that must be assessed for relevance in a paired experiment is almost linear in the depth of the evaluation. Averaging across 100 randomly selected baseline–experimental system pairs from the Robust Track experimental data set, there are 151 documents to assess for depth 100 evaluation of the two runs, 15.7 for depth 10, 8.1 for depth 5, and 3.37 for depth 2. Thus, for two runs, there is roughly the same document assessment effort in evaluating 50 topics to depth 100 as 900 topics to depth 5. This assumes that there is no start-up cost for each topic, such for the assessor to read and interpret it. Such an assumption is unrealistic, but is sufficient for the current discussion; more complex models can readily be developed when the per-topic start-up cost is known (see Carterette and Smucker (2007) for an example).

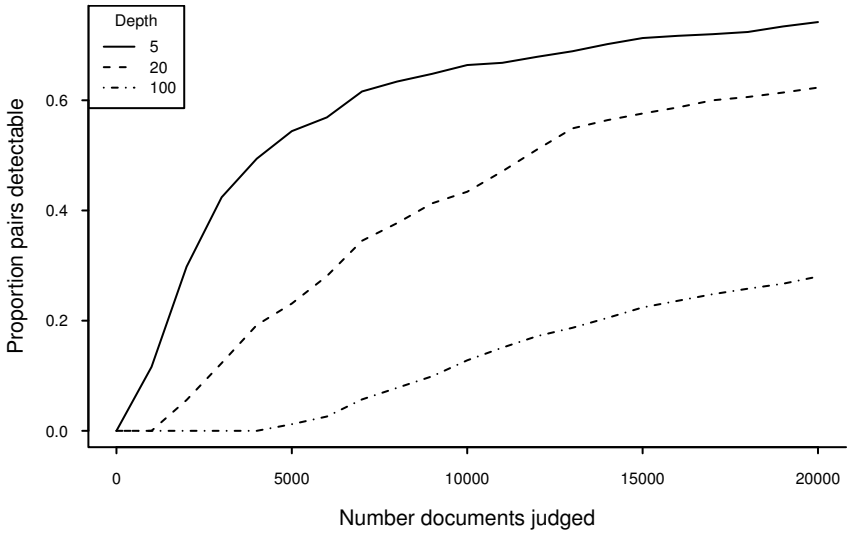


Figure 5.12: Proportion of empirical experimental effect sizes detectable for different number of documents judged with different assessment depths under AP. The distribution of effect sizes is taken from that observed on $1{,}000$ baseline–experimental system pairs randomly sampled from the TREC 2004 Robust Track systems. Power is 0.8, α is 0.05. Baseline systems are sampled from the second quartile, experimental systems from the top three quartiles.

The almost linear increase in assessment effort with evaluation depth, combined with the only slight improvement in observed effect size (Figure 5.11), means that it is far more efficient to spend effort on more topics than on deeper evaluation, as Figure 5.12 shows. Some $5{,}000$ documents must be judged with depth 100 evaluation for any of the observed effect sizes to be reliably detected as significant, whereas after this many judgments 23% of observed effect sizes are detectable with depth 20 assessment, and 56% with depth 5 assessment. This many judgments represent 33 topics at depth 100, 161 topics at depth 20, and 617 topics at depth 5.

One reason to consider performing deeper assessments than the above analysis suggests is to improve the reusability of the test topics. If the topics are later reused to test new systems, then the deeper the initial assessment, the less likely it is that new systems will return unassessed documents. This is the primary motivation for the deep assessment performed on the TREC collections. In a private lab, however, such depth of assessment may be prohibitively expensive. In such an environment, the method of score adjustment for correcting pooling bias, proposed in Chapter 6, offers a more efficient and flexible solution.

5.5 Summary

We have investigated the use of statistical power analysis in IR experimental design and interpretation. One of the main problems in design phase power analysis is predicting the variability of between-system score deltas. We have demonstrated that there is no single population of score deltas for any given metric, but rather a different population for each pair of systems. Estimating delta variability from past experience or from trial

experiments is inexact, and establishing reasonable confidence is expensive. On the other hand, iterative re-estimation of test power leads to bias in favour of finding significance, albeit a mild one. A hybrid approach is possible, but the experimenter must be explicit about their methodology. The issue can be avoided if the experimenter is able to specify predicted or consequential effect not as an absolute delta, but normalized by standard deviation; that is, as an effect size (ES). Which option the researcher should choose depends on their particular circumstances, but we propose the hybrid approach as an efficient (if methodologically complex) default.

One of the great benefits of power analysis is that it forces the experimenter to quantify the meaning of the experiment they are planning or have carried out. Contrary to common assumption, failure to find significance does not mean that consequential differences do not exist; one must examine the power of the test (or related measures, such as the confidence interval on the result) to draw such conclusions. And before performing an experiment, the researcher should consider what size of effect they expect, and whether the proposed test will detect it, even if they do not proceed to a more formal estimation of delta standard deviation. Inconclusive experiments are the bane of the scrupulous researcher, and trying one test collection (or metric) after another until some meaningful outcome is achieved is not, to say the least, methodologically sound.

For the purposes of planning experiments, having a rough estimate of a metric's typical range of delta standard deviations, and of how much a good new system might be expected to improve over a baseline, is valuable. In these terms, the 50-topic TREC collections are distinctly unpromising from a power-analysis point of view: to reliably distinguish a baseline (second-quartile) from an experimental (superior to baseline) system, a set of 100 to 200 topics is generally required under the AP metric. At the least, the experimenter should aggregate as many such collections together as possible to boost test power, as has been done with the Robust test collection. And if the researcher chooses or is forced to develop their own topics, then power analysis strongly suggests that shallow assessment of many queries is more reliable than deep assessment of a few.

Chapter 6

Score Adjustment for Pooling Bias

Ideally, every document in a test collection would be assessed for relevance to each query. In practice, exhaustive assessment is not feasible, due to the size of the document corpus. Instead, a subset of documents is selected for assessment. The standard method of selecting documents, used by collection formation efforts such as TREC, is to pool the top-ranked results from the participating systems, and assess only the documents in the pool. The assumption is that such a pool should cover the majority of relevant documents, so that unpooled documents can be assumed irrelevant. Increases in corpus size have, however, made this assumption suspect, leading to concerns that relevance assessments of pooled collections are seriously incomplete, and biased against unpooled systems (Buckley et al., 2007). Moreover, even pooling requires a considerable assessment effort per topic, beyond the means of most research groups deploying purpose-built collections. Such groups could make more efficient use of their resources by performing shallow assessment of a larger number of topics; but then the issue of pooling bias, if assessments are re-used to evaluate new systems, becomes still more severe, because of the even sparser coverage of relevant documents.

In this chapter, we propose a novel solution to the problem of assessment incompleteness and pooling bias. The basic approach is to empirically estimate the degree of bias in a particular collection and experimental setup, and then apply a score adjustment factor to the scores of unpooled systems to compensate for this bias. This adjustment can be calculated directly from the existing fully pooled systems, without performing any additional evaluation. We term this method *bias inference from systems*. Such inference, however, assumes that the unpooled system is similar to the pooled ones. More reliable is to fully assess all systems, existing and new, on an additional, small set of topics; to directly observe pooling bias against the new system on those topics; and to adjust the scores of the new system on the existing topics, for which it is unpooled, to compensate for this bias. We call this approach *bias inference from topics*. The latter approach is particularly suited to the dynamic collection of an ongoing development project at a research group or private lab. Here, retrieval methods are being continually developed and refined, and new topics added; meanwhile, a large number of legacy topics, assessed for some but not all systems under development, remain available. In such an environment, the assessments necessary for inference from topics are likely to already exist, and the method can be applied without further effort.

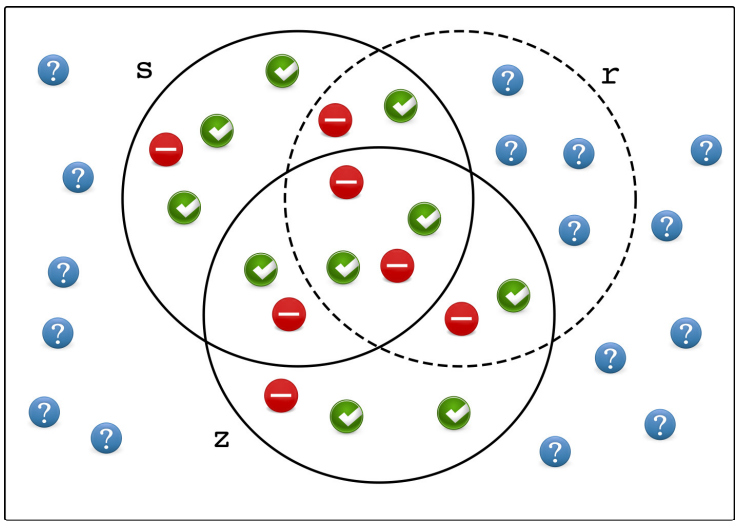


Figure 6.1: Pooled and unpooled systems. Systems s and z are included in the pool, and all documents returned by those systems are assessed for relevance. System r, however, is unpooled, and documents uniquely returned by it are unassessed.

We begin in Section 6.1 with an examination of qrel incompleteness, existing methods of dealing with this incompleteness, and the pooling bias that these methods display. In Section 6.2, we describe bias inference and score adjustment from systems, demonstrating that, while it works where systems are similar to each other, a high degree of bias remains if the new, unpooled system is markedly different from the existing, fully-pooled ones. Therefore, we propose instead, in Section 6.3, to infer bias and adjust scores based on a small set of common topics, fully assessed for all systems. We demonstrate that such a method is effective for even a small set of common topics, and is robust to heterogeneous systems.

6.1 Pooling bias

There has been considerable recent research interest in the subject of evaluation with incomplete relevance assessment; a survey was provided in Section 3.4. Exhaustive assessment being impractical, the traditional method of selecting documents for assessment is to run a set of representative retrieval systems against the collection, and pool their top-ranked results. Unpooled, and therefore unassessed, documents are assumed to be irrelevant. Such a method is biased against unpooled systems: all relevant documents located by pooled systems to assessment depth will be recognized, whereas unpooled systems can return unassessed, but in fact relevant, documents; assuming these to be irrelevant understates the effectiveness of the unpooled system. This is commonly referred to as *pooling bias*. The pooling process, and the condition of unpooled systems, is illustrated in Figure 6.1.

Studies on the early TREC collections estimated the degree of pooling bias using experiments in which a pooled system was removed from the pool, and its unpooled score compared to its pooled one (Zobel, 1998; Voorhees and Harman, 1999). Such studies concluded pooling bias is minimal, though Sanderson and Zobel (2005) later

found greater bias if all of a team's systems were held out, and even more so if a pool of automatic systems was used to evaluate manual ones. But the increase in corpus sizes since then makes it likely that the proportion of relevant documents found by pooling is decreasing, and hence the incompleteness of qrel sets is increasing, potentially worsening pooling bias. In particular, there is concern that, with large collections, pools are filled with keyword-rich, easy-to-find documents, making collections systematically biased against novel retrieval methods that attempt to go beyond keyword matching (Buckley et al., 2007). Even if the incompleteness of pooling were not an increasing concern, it would be attractive to have an evaluation method that was robust to incompleteness, so that less assessment effort could be expended on each topic and more topics could be judged; we have already seen in Chapter 5 that a large number of shallowly assessed topics achieves greater statistical power than a smaller number of deeply assessed ones.

Rather than assuming unassessed documents to be irrelevant, a proposed alternative for dealing with qrel incompleteness is to ignore unassessed documents altogether during evaluation. The Bpref metric is calculated only over assessed documents, with unassessed documents not considered (Buckley and Voorhees, 2004). The approach was extended to AP in the form of induced AP by Yilmaz and Aslam (2006), and then generalized by Sakai (2007b) into the concept of applying any standard evaluation metric to *condensed lists*; that is, lists from which unassessed documents have been removed, with the remaining, assessed documents shuffled up to form a continuous ranking. Unfortunately, condensed lists also suffer from bias, in this case in favour of unpooled systems (Sakai, 2008). This is because, if a document is not returned by pooling depth by any of the pooled systems, that is evidence in favour of its not being relevant; removing it from the ranking of an unpooled system, and allowing another document that was returned by pooled systems to take its place, replaces a document less likely to be relevant with one more likely to be relevant.

Besides having opposite effects on unpooled systems, the respective biases of assumed irrelevance and of condensed lists differ in other ways. First, the bias of assumed irrelevance is strictly one-sided, while that of condensed lists is not necessarily so. That is, assuming an unassessed document to be irrelevant can only be to the detriment of an unpooled system's score; ignoring it, while generally beneficial to the unpooled system, is not always so, since it could be that the unpooled document is relevant and the pooled document that replaces it when the list is shuffled up is not. A second difference is that, per unpooled document, the bias of assumed irrelevance decreases with an increase in the size of the pool, whereas that of condensed lists increases. The reason for this is the same for both methods: the larger the pool, the more likely it is that an unpooled document is irrelevant. We empirically observe the effects of these different bias characteristics in Section 6.1.2.

6.1.1 Materials

The data sets for the experiments reported in this chapter are the TREC 2004 Robust Track collection and runset, as well as those for the TREC 8 AdHoc Track. The Robust dataset is useful for the number of topics it includes. Participating systems were not pooled for the re-used topics, however. To avoid confusion between documents unpooled in our experiments, and documents unassessed in the original dataset, we treat all unassessed documents as if they had been pooled in the original dataset from the systems returning them, but assessed as irrelevant. The AdHoc dataset is useful because, unlike the Robust set, it includes manual runs. The 13 TREC 8 manual runs find

24% of the relevant documents, while the remaining 116 automatic runs find only 17% between them (the remainder are returned by both categories of runs). Additionally, the best 11 manual runs are also the best 11 systems altogether under many metrics. Manual runs therefore have quite different characteristics from automatic ones. We observe later that calculating the bias adjustment for heterogeneous systems is more demanding than for homogeneous ones; this will be investigated by attempting to calculate an unpooled manual system's bias adjustment based on a pool of automatic systems.

The choice of metric for the following investigation is affected by several considerations. First, we assume an environment in which only shallow assessment is performed, as one of the motivations is to make such assessment reliable, thereby enabling larger topic sets. Second, in our experiments, the qrel set will be changed frequently, by the exclusion and inclusion of different runs. Varying the qrel set makes normalized metrics unstable, as the estimated number of relevant documents will vary, too; not just the newly-unpooled system's score will be change, but also the scores of the still-pooled systems. Third, a top-weighted metric is desirable, for the greater sensitivity it provides. Thus, we want a shallow, unnormalized and therefore precision-based, top-weighted metric. The metric selected is what will be termed rank-biased precision, truncated at ten (tRBP@10), with the parameter p set to 0.8. This metric is a variant of RBP, using the base score, evaluated to depth 10 only, and with the rank weights scaled to sum to 1. The rank weights are thus:

$$\langle 0.224, 0.179, 0.143, 0.115, 0.092, 0.073, 0.059, 0.047, 0.038, 0.030 \rangle \ .$$

The metric can be understood as a top-weighted version of precision at ten. Using the base score of standard RBP would give the same results, aside from the scaling factor. Experiments were also performed with the standard precision at ten metric, and, allowing for its reduced sensitivity, achieved similar results.

The score adjustment methods described in this chapter use the mean system scores of unpooled systems. The goal of these methods is to adjust a system's unpooled score to be the same as the system's score would have been, had it been pooled. We evaluate the effectiveness of the adjustment measures by experimentally unpooling a pooled test system, calculating its adjusted unpooled score, and observing how close the adjusted score is to the pooled score, and for comparison how close the unadjusted score is, too. This is done multiple times, each time sampling a different unpooled system and set of pooled systems. To summarize the overall effectiveness of the method, we calculate the mean absolute error (MAE) over these randomly sampled experimental sets, as follows. Let the number of random samplings be n. On the ith sample, let t_i be the pooled score of the test system selected for that sample, and s_i be the system's unpooled (adjusted or unadjusted score); then the MAE of the method is:

$$MAE = \frac{1}{n} \sum_{i}^{n} |t_i - s_i| \ . \tag{6.1}$$

Statistical bias is mean non-absolute error (Equation 6.1 without the absolute operator, $|\ |$). Where error is uniformly one-sided (positive or negative), MAE is equal to statistical bias. Many of the adjusted methods described below have zero statistical bias; that is, the expected score after adjustment, taken over many different systems, is the true score on full pooling. Nevertheless, for a specific system, the adjusted methods may make errors one way or the other (that is, though statistically unbiased, they have non-zero variance); using MAE as our measure of accuracy accounts for this. When

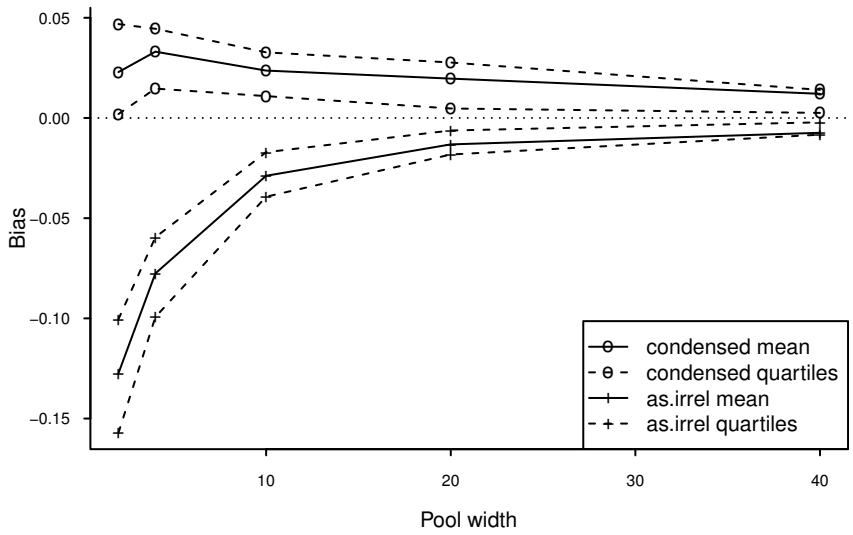


Figure 6.2: Empirical tRBP@10 ($p = 0.8$) bias for unpooled systems, using different pooled widths, on the TREC 2004 Robust Track data set. Pooling is to depth 10. Graphed is the mean, and quartiles, of the difference in mean system tRBP score, between the true score for the unpooled system, and the score using either condensed lists or assumed irrelevance. Each data point represents 100 system set samplings.

in this chapter we speak of pooling bias, we are referring to the error in mean score suffered by a particular system (an element in the sum in Equation 6.1), not the mean error (that is, the statistical bias) observed across all systems.

The size of a pool is determined by two factors: the number of systems included in the pool, which we will refer to as the pool's *width*; and the depth to which those systems are pooled. Throughout this chapter, pool depth will be 10, corresponding to the evaluation depth of the metric. Having evaluation and pooling depth being the same means that pooled systems are fully assessed: evaluating beyond pooling depth would create a second-order pooling bias even for pooled systems, as systems may benefit beyond the pool depth by similarity to other pooled systems.

6.1.2 Bias of exclusion from the pool

In this section, we experimentally observe the empirical bias that affects a system's score when that system is excluded from the pool. A set of $n \in \{2, 4, 10, 20, 40\}$ systems is randomly sampled from the TREC 2004 Robust data set as the pool, and an additional system is sampled as the evaluated system. The evaluated system's score is calculated under full assessment, as if it were pooled; then it is removed from the pool, and its score is calculated again. Two unpooled scores are calculated for the evaluated system, one with unassessed documents assumed irrelevant, and the other with such documents excluded (that is, using condensed lists). The unpooled minus the pooled score is the empirical bias, under the chosen scoring scheme, suffered by the system on exclusion from the pool. This process is repeated for a large number of randomly sampled pooled and unpooled systems, and the mean and quartiles of the biases for each unpooled assessment method are calculated for each pool width.

The pooling bias suffered (or enjoyed) by unpooled systems in our data set, as calculated using the above experimental method, is shown in Figure 6.2. As anticipated, the use of condensed lists is biased in favour of the unpooled system, while assuming unassessed documents to be irrelevant is biased against it. For assumed irrelevance, the bias steadily decreases as the pool width increases, roughly halving when the pool width is doubled. There are two causes for this: as the number of unassessed documents falls, the weight of the unassessed documents in the final score also drops; and as the pool width increases, the probability that an unassessed document is in fact irrelevant (as assumed) also increases. The change in bias for condensed lists is less straightforward, however. Again, wider pools mean fewer unassessed documents, and a higher likelihood that the remaining unassessed documents are irrelevant; but since a greater likelihood of irrelevance means a greater bias per unassessed document under condensed lists, these two effects counteract each other. Thus, condensed lists are less biased than assumed irrelevance for narrow pools, whereas assumed irrelevance is less biased for wide pools. It can also be observed that, with wider pools, the distribution of bias becomes more skewed (mean closer to third quartile). The reason for this skew is that the data set contains families of similar runs; as pool width increases, so does the probability that the unpooled system will have its documents in fact pooled by another, almost identical system from the same family. A greater proportion of systems therefore demonstrate almost no pooling bias; but the mean bias is still pulled higher by the remaining, non-familial unpooled systems.

How serious is the bias observed in Figure 6.2? Consider the typical evaluation situation of comparing a new experimental system to an existing baseline; assume (as is likely) that the baseline is pooled, but the new system is not. From the TREC 2004 Robust systems, we shall take the second quartile systems, by mean RBP, as representative baselines; and, for each such baseline, any higher-scoring system as a representative experimental improvement (the same setup as used in Section 5.2). Then the median difference between the mean RBP ($p = 0.8$) score of a baseline and an experimental system is 0.030. With narrow pools (of, say, ten pooled systems or fewer), the median delta is small enough to be almost entirely swamped by the bias shown in Figure 6.2. And even with wider pools, while the median delta is greater than the bias, the bias will often be sufficient to prevent a true delta from achieving statistical significance.

It can be observed from Figure 6.2 that the degree of bias under each scheme changes with pool width, and varies considerably even within the one pool width depending on the particular set of systems sampled. It would also vary for different pool depths, for different metrics, for different collections, and for different system populations. Thus, it is not possible to come up with a single general adjustment factor, or even parameterized set of factors, that can be satisfactorily applied to correct pooling bias. Bias must be empirically estimated on each experimental data set.

6.2 Bias inference from systems

In Section 6.1.2 we observed pooling bias through a leave-one-out experiment. We now propose two methods of estimating and correcting pooling bias, in an evaluation where one or more unpooled systems are compared against a set of pooled ones. The first method uses a simulated leave-one-out experiment on the systems under evaluation. The leave-one-out experiment cannot be performed on the unpooled system itself, because the relevance assessments to calculate its pooled score do not exist (if they did, there would be no need for score adjustment). But the experiment can be performed on

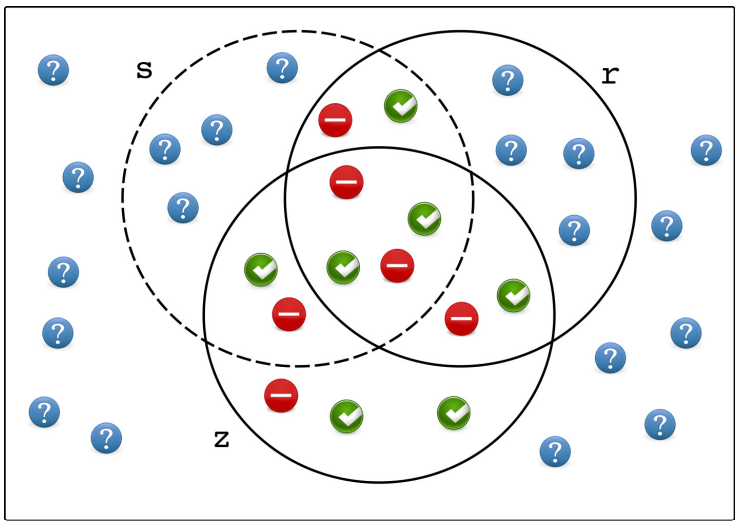


Figure 6.3: Experimental unpooling. System s is withdrawn from the pool, and its uniquely returned documents are marked as unassessed. System r, previously unpooled, is added to the pool, to maintain a pool width of 2. The unpooled score of System s is then calculated. We do not have assessments for documents uniquely returned by System r, but since this does not affect the score of System s (unless we are using a recall-normalized metric), these assessments are not required. The effect of adding System r to the pool is to retain the assessments for documents returned by Systems r and s, but not by System z (or any other systems).

the pooled systems in the evaluation set, and the pooling bias observed on these pooled systems used to estimate and correct the bias suffered by the actually unpooled system. We call this method of score adjustment *bias inference from systems*.

Bias inference from systems is described in Algorithm 6.1. Each of the fully-pooled systems $s \in S$ is held out of the pool in turn, which is to say that the documents uniquely returned by s up to pooling depth are marked as unassessed. A key refinement is that the unpooled system, r, is added to the pool when s is removed. This has the effect of retaining the relevance assessments of documents returned by only r and s. This refinement is necessary to ensure that the pooling bias of a pool of width $n = |S|$ is being calculated, not that of a pool of width $n - 1$. Because these documents were pooled by s, they have relevance assessments. Documents uniquely returned by r alone are unassessed, but since they cannot affect the score of s (or indeed of any of the pooled systems), that does not matter. (Note that if a normalized metric were being used, the relevance status of documents uniquely returned by r would become germane.) Simulated unpooling is illustrated in Figure 6.3. The simulated pool having been constructed, the observed bias β_s between the mean pooled and unpooled score of s is noted. Repeating this experiment for each of the n pooled systems gives n different observed pooling biases β_s, and the mean of these β_s values give an estimate of the mean pooling bias, which is used as the unpooled adjustment factor a. The adjustment factor a is added to the score of the unpooled system, r, to derive its adjusted score. A sample working is given in Figure 6.4.

We assess the effectiveness of system-based bias adjustment on systems randomly sampled from the TREC 2004 Robust Track data set, for the base unpooled scoring

Algorithm 6.1 Adjust scores based on inference from systems

$T \leftarrow$ set of topics
$S \leftarrow$ set of (pooled) systems
$Q \leftarrow$ set of qrels on T derived from pool of S
$r \leftarrow$ (unpooled) system
for $s \in S$ **do**
  $Q' \leftarrow Q \backslash \{$documents uniquely pooled from $s\}$
  $Q' \leftarrow Q' \cup \{$documents returned by $r\}$
  $t_s \leftarrow$ mean (pooled) score of s evaluated against Q
  $u_s \leftarrow$ mean (unpooled) score of s evaluated against Q'
  $\beta_s \leftarrow t_s - u_s$ ▷ unpooled for against s
end for
$a \leftarrow \sum_{s \in S} \beta_s / |S|$ ▷ adjustment factor
$u_r \leftarrow$ mean (unpooled) score of r evaluated against Q
return $u_r + a$

Pool Width	Robust TREC 2004		AdHoc TREC 8 Manual	
	Raw	Adjusted	Raw	Adjusted
2	0.127	0.041	0.451	0.302
4	0.078	0.028	0.384	0.294
10	0.029	0.015	0.283	0.237
20	0.013	0.008	0.231	0.203
40	0.007	0.006	0.177	0.159

Table 6.1: Bias inference from systems. Mean absolute error (MAE) of leave-one-out score adjustment and unadjusted scores for tRBP@10 ($p = 0.8$) under presumed irrelevance, for different numbers of pooled systems. The left columns are for all systems from the TREC 2004 Robust Track. The right columns show estimation of unpooled manual system scores from pooled automatic systems on the TREC-8 AdHoc Track data set. Bias (mean non-absolute error) is 0 for column 3 (adjusted scores on all TREC 2004 Robust systems), and negative MAE for columns 2, 4, and 5 (unadjusted scores, and adjusted scores on TREC 8 Manual systems).

method of assuming unassessed documents to be irrelevant. The error of adjusted and raw scores is compared. A total of $n \in \{2, 4, 10, 20, 40\}$ systems are selected to form the pooled set, and one further system is selected as the unpooled, test system. A qrel set is formed from the documents returned by the pooled systems up to depth 10. The true pooled score of the unpooled system is calculated, and compared against the raw and the adjusted unpooled score. This is repeated 100 times for each pool width. The results, as measured by MAE, are shown in the second and third columns of Table 6.1. The adjusted scores have a much smaller mean absolute error from the true pooled scores than do the unadjusted scores, an effect that is strongest with small pool widths. Moreover, because of the random sampling, the adjusted scores are an unbiased estimate of the true pooled score (as likely to over- as to under-estimate), so that the mean (non-absolute) error is 0; in contrast, the unadjusted scores are all underestimates, making the mean error identical to the (negative) mean absolute error.

Topics	Pooled Systems						New System
	s_1		s_2		s_3		r
t_1	0.26	*0.23*	0.31	*0.28*	0.25	*0.17*	*0.28*
t_2	0.11	*0.09*	0.25	*0.19*	0.35	*0.22*	*0.18*
t_3	0.08	*0.07*	0.09	*0.08*	0.14	*0.11*	*0.15*
t_4	0.12	*0.08*	0.12	*0.10*	0.48	*0.42*	*0.21*
t_5	0.12	*0.07*	0.35	*0.31*	0.42	*0.38*	*0.31*
t_6	0.15	*0.12*	0.27	*0.24*	0.41	*0.35*	*0.37*
t_s	0.14		0.23		0.34		
u_s		*0.11*		*0.20*		*0.28*	*0.25*
β_s	0.03		0.03		0.06		
a							+ 0.04
$\hat{t}_r$							0.29

Figure 6.4: Illustrative calculation of bias inference from systems. Pooled scores are shown in black; unpooled in red italics. Systems s_1, s_2, and s_3, are pooled on the topic set $\{t_1, \ldots, t_6\}$, while System r is unpooled. Each of the pooled systems is withdrawn from the pool, with r added, and its unpooled score is derived. The difference (β_s) between the mean pooled score (t_s) and unpooled score (u_s) of each pooled system is calculated. The mean of these differences becomes the adjustment factor a. The adjustment factor is added to the unpooled score for System r to derive our estimate $\hat{t}_r$ for that system's pooled score.

The high accuracy observed for the adjusted scores on the TREC 2004 Robust Track data sets is, however, misleading. In estimating the pooling bias of the unpooled system based on that observed on the pooled systems, the assumption is being made that the unpooled system is similar to the pooled ones. The random sampling method employed in the assessment reported in Table 6.1 artificially satisfies this assumption: because pooled and unpooled systems are randomly sampled from the same population, they are by (statistical) definition similar. In real evaluations, however, we cannot automatically assume that the new, unpooled system is similar to the existing, pooled systems. After all, the research and development effort that has gone into the new system is aimed at making it different from, and better than, the existing ones. And if the new system is better than the existing ones, then its uniquely returned documents are more likely to be relevant than those of the existing systems.

The more demanding and realistic test, therefore, is how accurate the adjustment method is when the unpooled system is not (randomly) similar to the pooled systems, but instead embodies some systematic differences. We simulate this using the TREC 8 AdHoc dataset, attempting to adjust the score of each of the 11 high-scoring manual runs based on pools of automatic systems. The results are shown in columns four and five of Table 6.1. The pooling bias on raw scores is enormous for narrow pools, almost half of the maximum achievable score for a pool width of 2, and is still substantial even for a relatively broad (but shallow) pool of 40 systems. The adjusted score does reduce the bias somewhat, but proportionally much less so than for the randomly-cosampled TREC 2004 systems. The relative effectiveness of the adjustment diminishes, too, with pool width: at pool width 40, the leave-one-out simulation on the pooled automatic

systems is finding few unassessed documents and slight biases, and so producing an adjustment factor entirely inadequate for the large bias still suffered by manual runs.

The high-scoring manual runs of the TREC 8 AdHoc data set form a particularly demanding challenge for score adjustment. Nevertheless, the experiments reported in the rightmost columns of Table 6.1 underline the weakness of bias estimation from systems. The method assumes that the unpooled system is similar to the pooled ones, and derives the estimation of bias and the adjustment factor based on this assumption. In practice, though, the assumption of (randomly) similar systems is unlikely to be justified by the circumstances of the evaluation. And if the new, unpooled system is systematically different from the existing, pooled ones, then score adjustment based on bias estimation from systems will suffer lower accuracy.

6.3 Bias inference from topics

Bias estimation from systems, described in Section 6.2, relies on the (often unrealistic) assumption that the new, unpooled system has been sampled from the same population as the pooled systems. If this assumption is invalid, then the adjustment is unreliable. The second method of score adjustment we propose performs inference along a different dimension. Rather than inferring pooling bias from one set of systems to another system, the second method, *bias inference from topics*, infers bias from one set of topics to another set of topics. The method does require full assessment of the new system on a topic subset, which is not achievable in all experimental circumstances. But the method rests on a sounder inferential basis, since the assumption that topics have been randomly sampled from the same population is more plausible (and, in many circumstances, directly enforceable) than the assumption that systems have been.

The evaluation scenario to which bias inference from topics is addressed is as follows. There is a set of existing systems, S, which have been fully assessed on a set of existing topics T. A new system, r, is to be evaluated. This system was not pooled for the relevance assessments on T, and so is not fully assessed for them. We wish to estimate the pooling bias that r suffers on the topics T. To do this requires the existence or creation of a set of *common topics* C which the new system r and the existing systems S are all pooled and fully assessed on. The common topics C could be created by fully assessing a new set of topics, or performing supplementary assessments on a subset of the existing topics T for documents uniquely returned by r (if such supplementary assessments are judged to be methodologically sound). It may also happen, in a dynamic evaluation and development environment, with new systems and new topics being regularly added, that the set of common topics C will already have been produced as part of the ongoing evaluation process.

Given the existence of a set of common topics C that all systems, new and existing, are fully assessed upon, it becomes possible to directly perform a leave-one-out experiment over C for the new system, r. This leave-one-out experiment calculates the true pooling bias against r on the topics C, and this bias can be used as an adjustment factor for the unpooled scores r achieves on existing topics T. Moreover, if it is the case (or can reasonably be assumed) that T and C have been randomly sampled from the same population of topics, then the adjusted mean score will be an unbiased estimate of the true pooled mean score for r. This adjustment method is described in Algorithm 6.2, and a sample working is given in Figure 6.5.

We begin by analyzing bias estimation from topics as a sample-based ratio (or rather, delta) estimator. Then an experimental assessment is performed, validating the

Algorithm 6.2 Adjust scores based on inference from topics

$T \leftarrow$ set of topics
$S \leftarrow$ set of (pooled) systems
$Q_T \leftarrow$ qrels on T derived from pool of S
$r \leftarrow$ (unpooled) system
$C \leftarrow$ set of common topics
$Q_C \leftarrow$ pool $S \cup r$ on C and assess for relevance
for $c \in C$ **do**
  $Q'_C \leftarrow Q_C \backslash \{$documents uniquely pooled from $r\}$
  $t_{r,c} \leftarrow$ (pooled) score of r on c evaluated against Q_C
  $u_{r,c} \leftarrow$ (unpooled) score of r on c evaluated against Q'_C
  $\beta_{r,c} \leftarrow t_{r,c} - u_{r,c}$ ▷ unpooled bias against r on c
end for
$a \leftarrow \sum_{c \in C} \beta_{r,c} / |C|$ ▷ adjustment factor
$u_r \leftarrow$ mean (unpooled) score of r evaluated against Q_T
return $u_r + a$

formal analysis and demonstrating that topic-based adjustment is more accurate than system-based adjustment, and is robust to the case that the system whose score is adjusted is quite different in nature from the existing ones.

6.3.1 Analysis

The proposed method of topic-based adjustment is a form of *ratio estimator* (Thompson, 2002, Chapter 7) (though, as will be explained, we are working with deltas rather than ratios; see also Lessler and Kalsbeek (1992, Chapter 10) for a description of bias-correction through resampling). Ratio estimators are of use in a situation where high-bias observations exist for a large sample, while low-bias or unbiased observations exist for a small subset of that sample. For the subset that both low- and high-bias observations exist on, the mean ratio (or difference) between the high- and low-bias observations is calculated. This ratio is then applied to adjust the high-bias observations, reducing the bias of the final estimator. Note that what matters is the relative bias of the estimators, not their relative variability; a low-variability estimator cannot be used to adjust a high-variability estimator if both are unbiased.

The application of ratio estimation as described above to the estimation of pool bias using common topics is fairly straightforward. The value we wish to estimate is the true pooled mean score of System r. The full sample is the superset of topics, $T \cup C$. The high-bias observations are the unpooled scores of r on $T \cup C$, and the low-bias observations are the pooled scores of r on C. These latter observations are in fact exact, free of both bias and variance. Importantly, the unpooled scores have a definite bias (tendency to error either above or below the true score). Under assumed irrelevance, this inaccuracy is one-sided, never overestimating scores. Condensed list inaccuracy is not one-sided, but nevertheless is strongly biased high. Therefore, in both cases, ratio estimation can reduce error. Rather than estimating the ratio between pooled and unpooled scores, however, we estimate the difference; this is because unpooled scores can be 0, leading to an undefined ratio.

More formally, let r be the system whose score we wish to adjust on the topics T, for which r is unpooled. Let the desired value, namely the mean of the true pooled

Topics		Pooled Systems			New System			
		s_1	s_2	s_3	r		$\beta_{r,c}$	a
C	t_1	0.26	0.31	0.25	*0.28*	0.36	0.08	0.09
	t_2	0.11	0.25	0.35	*0.18*	0.28	0.10	
	t_3	0.08	0.09	0.14	*0.15*			
	t_4	0.12	0.12	0.48	*0.21*			
	t_5	0.12	0.35	0.42	*0.31*			
	t_6	0.15	0.27	0.41	*0.37*			
	t_s	0.14	0.23	0.34				
	u_r				$\Rightarrow$			*+ 0.25*
	$\hat{t}_r$							0.34

Figure 6.5: Illustrative calculation of bias inference from topics. Systems s_1, s_2, and s_3 are pooled on topics $\{t_1, \ldots, t_6\}$, whereas System r is initially unpooled on all topics. Topics t_1 and t_2 are chosen as the common topics C, and System r is pooled for these topics by assessing any documents uniquely returned by the system. (Alternatively, an entirely new set of topics could be chosen as the common topics.) The difference $\beta_{r,c}$ between the pooled and unpooled scores of System r on every common topic c is calculated. The mean of these differences becomes the adjustment factor a, which is added to System r's unpooled score u_r to derive an estimate $\hat{t}_r$ for its pooled score. Note that inference from topics gives a much higher estimate of pooling bias in this instance than estimation from systems (Figure 6.4). Systems s_1 through s_3 are more similar to each other than they are to r, and hence cover each others' documents more thoroughly. Therefore, inference from systems understates the bias that System r suffers from not being in the pool.

scores for r, be μ_t. Let the total number of topics $|C \cup T|$ be N, and let $n = |C|$ be the number of common topics. Denote the unpooled score achieved by System r on topic i as u_i, and the true pooled score (known only on the common topics) as t_i. Then the adjustment factor a is derived from the n common topics as:

$$a = \frac{1}{n} \sum_{i=1}^{n} (t_i - u_i) \,. \tag{6.2}$$

The estimation of the true mean score μ_t, using the adjusted estimator $\hat{\mu}_a$, for all N topics is:

$$\hat{\mu}_a = \frac{1}{N} \sum_{i=1}^{N} (u_i + a) \,. \tag{6.3}$$

For n of the N topics, we know the true score t_i, not just the unpooled score u_i; however, the mean true score of these n topics is by derivation $\sum_i^n (u_i + a)$, and so does not need to be separately accounted for in Equation 6.3.

The adjustment factor a is itself an estimate of the true adjustment factor A which should be applied to the mean unpooled scores μ_u to achieve the true mean score μ_t. The value of the true adjustment is:

$$A = \frac{1}{N} \sum_{i=1}^{N} (t_i - u_i) \,. \tag{6.4}$$

Since a is calculated from n topics randomly sampled from the same population as the N full topics (see Equation 6.2), it follows that a is an unbiased estimator of A. Therefore, $\hat{\mu}_a = \mu_u + a$ is an unbiased estimator of μ_t. The approximate variance of this estimator is (Thompson, 2002, Chapter 7, Equation 4):

$$\text{var}(\hat{\mu}_a) \approx \frac{N-n}{N} \cdot \frac{\sigma_a^2}{n} \,. \tag{6.5}$$

The left-hand fraction of Equation 6.5 is a small-population adjustment, accounting for the fact that n of the N values are precisely known in each sample (if all N values were known, there would be no variance to the estimator). The numerator of the right-hand fraction is:

$$\sigma_a^2 = \frac{1}{N-1} \sum_{i=1}^{N} (t_i - (u_i + A))^2 \tag{6.6}$$

namely, the mean squared error of the per-topic adjusted scores, compared to the true scores, across all N topics, assuming the true adjustment factor A were used (A is correct for the mean score, but not for every topic score). To be clear, the value of Equation 6.5 cannot be calculated in actual evaluations (though it can be estimated), because not all N true scores are known; the equation is given here as an analytical tool for the experimental data sets.

For the n common topics C, we have the true pooled scores, without the need for an adjustment. An alternative estimator for the overall true mean score μ_t would therefore be simply to take the mean true pooled scores $\bar{t}$ observed across these n topics. Denote this *sampled estimator* as $\hat{\mu}_n$. Since C and T are randomly co-sampled, $\hat{\mu}_n$ is an unbiased estimator of μ_t. The variance of the estimator is (Thompson, 2002, Chapter 2, Equation 5):

$$\text{var}(\hat{\mu}_n) = \frac{N-n}{N} \cdot \frac{\sigma_t^2}{n} \,, \tag{6.7}$$

where σ_t^2 is the variance of the true scores across all N topics. Comparing Equations 6.5 and 6.7, we can see that the sampled estimator $\hat{\mu}_n$ is more accurate (has lower variance) than the adjusted estimator $\hat{\mu}_a$ when σ_t^2, the variance of the true scores, is less than σ_a^2, the mean squared error of the adjusted scores. That is, if systems tend to get similar scores for different queries, then a small number n of fully-pooled queries are sufficient in themselves to give a reliable estimate of the true mean score of r. But it has been seen in Chapter 4 that (unstandardized) scores are highly variable between topics. Therefore, provided the adjustment values given by bias estimation from topics are reasonably accurate, then it can be anticipated that adjustment on the unpooled topics will give more reliable results than using the fully-pooled topics alone. This is confirmed empirically in Section 6.3.2.

Another estimator for the true mean pooled score is, simply, the unadjusted score. We know from Figure 6.2 that this *unadjusted estimator* $\hat{\mu}_u$ is biased, unlike the preceding estimators; nevertheless, it is possible for a biased but low-variance estimator to give lower mean errors than a high-variance unbiased one. The mean error on the unadjusted scores is A, of which a is an estimator. The error on the mean adjusted score is $A - a$; that is, it is dependent on how accurate a is as an estimate of A. Therefore, the adjusted mean score will be more accurate than the unadjusted mean score if and only if $0 < (a/A) < 2$; that is, if the following two conditions are met:

1. the estimated adjustment a is the same sign as the true adjustment A (adjustment is not making the error worse); and

Algorithm 6.3 Sample systems, topics to assess adjustment accuracy

```
T ← 249 TREC Robust 2004 topics
X ← 110 TREC Robust 2004 systems
I ← 100                                   ▷ number of system sampling repeats
J ← 200                                   ▷ number of topic sampling repeats
for w ∈ {2, 4, 10, 20, 40} do             ▷ pool widths
    for i ∈ 1 → I do
        S ← sample(X, w)
        r ← sample(X\S, 1)
        Q ← pool S ∪ r on T
        Q′ ← pool S on T
        t_r ← mean (true) score of r evaluated against Q
        u_r ← mean (unpooled) score of r evaluated against Q′
        for n ∈ {10, 20, 40, 100} do
            for j ∈ 1 → J do
                C ← sample(T, n)          ▷ common topics
                a ← estimate adjust. on C as in Algorithm 6.2
                e_r ← t_r − (u_r + a)     ▷ adjustment error
                E_{w,n} ← E_{w,n} + |e_r|
            end for
        end for
    end for
end for
E ← E/(I ∗ J)                             ▷ Take mean error over I ∗ J repeats
return E
```

2. the estimated adjustment a is no more than twice the true adjustment A (adjustment is not overshooting).

Where unadjusted scores always misestimate the true scores in the same direction, as occurs when unassessed documents are assumed irrelevant, Condition 1 is always met. And since the expected value of a is A, Condition 2 will rarely fail to be met. Thus, adjusted scores will almost always be more accurate than unadjusted scores where the base scoring method is to assume unassessed documents are irrelevant. For condensed lists, however, the unadjusted scores, which are usually overestimates, may be underestimates, such that Condition 1 is not guaranteed; the calculated adjustment could end up having the wrong sign from the actual error. Additionally, with the center of the distribution of errors under condensed lists being closer to 0, there is a greater likelihood of Condition 2 being violated. Therefore, the effectiveness of score adjustment for condensed lists requires empirical assessment.

6.3.2 Experiments

In this section, we empirically assess the improvement in accuracy that score adjustment, based on bias inference from topics, provides over using the unadjusted, unpooled scores. We also compare the accuracy of the adjusted scores with that of the mean score estimate provided by the fully-pooled scores on the common topics alone. The method is again to random sample $m \in \{2, 4, 10, 20, 40\}$ systems from the TREC 2004 Robust dataset as pooled systems, and one system as the unpooled

Pool Width	Estimator	Common Topics				
		0	10	20	40	100
	True		0.078	0.054	0.036	0.019
2	Unadjusted	0.127	0.122	0.117	0.107	0.076
	Adjusted		0.041	0.028	0.019	0.010
4	Unadjusted	0.078	0.074	0.071	0.065	0.046
	Adjusted		0.031	0.021	0.014	0.008
10	Unadjusted	0.029	0.028	0.027	0.024	0.017
	Adjusted		0.016	0.012	0.008	0.004
20	Unadjusted	0.013	0.013	0.012	0.011	0.008
	Adjusted		0.010	0.007	0.005	0.003
40	Unadjusted	0.007	0.007	0.007	0.006	0.004
	Adjusted		0.006	0.005	0.003	0.002

Table 6.2: Mean absolute errors for different estimators of true mean pooled score, over randomly-sampled systems from the TREC 2004 Robust Track, for different pool widths and numbers of common (fully-assessed) scores. The metric is tRBP@10, $p = 0.8$. The base method is to assume that unassessed documents are irrelevant. The "true" method estimates the true mean score across all 249 topics from the mean scores of the fully-assessed common topics alone. The "unadjusted" method takes the mean of the unadjusted scores for the unpooled topics, and of the fully-assessed scores for the common topics. The "adjusted" method performs score adjustment based on bias estimation on the common topics, then takes the mean of the adjusted score on the unpooled topics, and of the fully-assessed scores for the common topics.

system, whose true (pooled) score we wish to estimate. Three estimation methods are compared: using the unadjusted score, with unassessed documents assumed irrelevant; fully assessing $n \in \{10, 20, 40, 100\}$ topics for the unpooled system, and estimating its overall mean score based solely on these n topics; and choosing $n \in \{10, 20, 40, 100\}$ topics for full assessment as common topics, and performing score adjustment on the unpooled system using bias inference based on these common topics. The experimental method is given in more detail for the adjusted scores in Algorithm 6.3. For both the unadjusted and the adjusted estimation methods, the scores of the fully-assessed topics are included in calculating the overall mean.

The resulting mean absolute error scores are given in Table 6.2. The mean of the fully assessed scores alone (the sampled estimator, $\hat{\mu}_n$) is superior to the combination of fully-assessed and unadjusted scores for narrow pools (width of 2 or 4 systems). But in all cases, the adjusted score method outperforms the alternatives. It most widely outperforms unadjusted scores for narrow pools but many common topics, and the sampled estimator for wide pools but few common topics. In any case, as few as 10 common topics are sufficient to reduce unadjusted score error to as little as a third of the original. Taking the mean absolute error, and averaging over so many systems, obscures the precise calculations, but the order of variation with number of common topics, laid out in Equation 6.5 for the adjusted estimator $\hat{\mu}_a$, and Equation 6.7 for

Pool Width	Estimator	Common Topics				
		0	10	20	40	100
	True		0.078	0.054	0.036	0.019
2	Unadjusted	0.034	0.033	0.032	0.029	0.021
	Adjusted		0.041	0.028	0.019	0.010
4	Unadjusted	0.035	0.034	0.032	0.030	0.021
	Adjusted		0.031	0.022	0.015	0.008
10	Unadjusted	0.024	0.023	0.022	0.020	0.014
	Adjusted		0.018	0.013	0.009	0.005
20	Unadjusted	0.020	0.019	0.018	0.016	0.012
	Adjusted		0.013	0.009	0.007	0.003
40	Unadjusted	0.012	0.012	0.011	0.010	0.007
	Adjusted		0.009	0.007	0.005	0.003

Table 6.3: Mean absolute score errors for different estimators of true mean pooled scores, with the base method of handling unassessed documents being to remove them and create condensed lists. Other details are as for Table 6.2.

the true-scores-only estimator $\hat{\mu}_n$, is roughly borne out by Table 6.2. In particular, the errors on $\hat{\mu}_a$ and $\hat{\mu}_n$ decline, with the increase in the number of common topics, at roughly the same rate. Compared to bias estimation from systems, shown in Table 6.1, estimation from topics leads to similar or marginally higher error with 10 common topics, with the error decreasing as the number of common topics increases. But this is on randomly-sampled system sets that are artificially favourable to the assumptions of the from-system inference method. As will be observed later, inference from topics performs far better than from systems when the latter's assumptions are violated.

Table 6.3 reports the same experiment as Table 6.2, but using condensed lists as the base scoring method for unassessed documents. The analysis in Section 6.3.1 indicated that score adjustment from common topics would almost always outperform unadjusted scores where unassessed documents are assumed irrelevant, since that unadjusted method has one-sided bias; this prediction is validated by Table 6.2. The same analysis suggested that score adjustment would not be so unequivocally superior for condensed lists, though, because the bias there, while predominantly positive, is not universally so. The latter prediction is borne out by the results in Table 6.3. For very narrow pools, of only 2 or 4 systems, the fact that a document is unpooled is only weak evidence against its relevance, so excluding it from the condensed lists is only mildly biased in favour of the unpooled system; in these circumstances, if the number of common topics is also small (say only 10, or perhaps 20), then score adjustment is not able to improve much on the unadjusted score, and can make things worse. Nevertheless, as either the pool width or the number of common topics increases, score adjustment achieves distinctly superior accuracy to unadjusted scores on condensed lists.

So far, we have tested the case where the unpooled system is randomly sampled from the same population as the pooled one. We saw previously in Section 6.2 that this is a situation in which bias estimation from systems, via a simulated leave-one-out

Pool Width	Estimator	Common Topics			
		0	10	15	20
	True		0.067	0.052	0.041
2	Unadjusted	0.451	0.361	0.316	0.271
	Adjusted		0.060	0.046	0.037
4	Unadjusted	0.384	0.308	0.269	0.231
	Adjusted		0.059	0.045	0.037
10	Unadjusted	0.283	0.226	0.198	0.170
	Adjusted		0.054	0.041	0.033
20	Unadjusted	0.231	0.185	0.162	0.139
	Adjusted		0.050	0.038	0.031
40	Unadjusted	0.177	0.141	0.124	0.106
	Adjusted		0.045	0.035	0.028

Table 6.4: Mean absolute score errors for different estimators of true mean pooled scores, estimating the scores of the 11 best TREC 8 AdHoc track manual runs, with pools drawn from the automatic runs. The base method of handling unassessed documents is to assume them irrelevant. Other details are as for Table 6.2.

experiment, also performs well. Yet to be tested is the claim that bias estimation from common topics is also robust when the unpooled system is systematically different or distinct from the pooled ones. For inference from systems, this requirement was tested (and found wanting) by attempting to adjust the scores of a system drawn from the best 11 manual runs in the TREC 8 AdHoc dataset. We now repeat that experiment using common topic inference. The results are shown in Table 6.4. Because the topic set is much smaller than for the Robust collection (50 topics, compared to 249), the range of common topic set sizes is more constrained. As before, the error on unadjusted scores is very wide. But whereas inference from systems only managed to reduce this error by at most a third (see Table 6.1), inference from common topics slashes it to as little as a sixth on 10 common topics, and even less for larger common topic set sizes. Error is still about 50% higher than for the randomly-sampled Robust set, whereas the error of the true topic estimator is slightly lower; nevertheless, the adjusted estimator has 10% lower error than relying on the true topics alone for a 2-system pool, and this error decreases as pool width increases. Even in the extreme case of attempting to estimate the true score of a top-performing manual system based on a pool drawn from two automatic runs, score adjustment based on common topics is able to achieve higher accuracy than relying on the common topics alone.

Finally, Table 6.5 shows the results for condensed lists on the TREC 8 dataset, again estimating high-scoring manual runs from automatic pools. On this dataset, the condensed list method with unadjusted scores shows much lower error for unpooled systems than does assuming documents irrelevant, especially for wider pools. The reason is that the negative evidence of an unassessed document's being irrelevant, from it not being returned in the pool, is balanced by the positive evidence of it being returned by a top-ranking manual run; with wider pools, these two largely balance out,

Pool Width	Estimator	Common Topics			
		0	10	15	20
	True		0.067	0.052	0.041
2	Unadjusted	0.154	0.123	0.108	0.093
	Adjusted		0.054	0.041	0.033
4	Unadjusted	0.091	0.073	0.064	0.055
	Adjusted		0.046	0.035	0.028
10	Unadjusted	0.033	0.027	0.023	0.020
	Adjusted		0.036	0.027	0.022
20	Unadjusted	0.018	0.014	0.013	0.011
	Adjusted		0.030	0.023	0.019
40	Unadjusted	0.013	0.010	0.009	0.008
	Adjusted		0.026	0.020	0.016

Table 6.5: Mean absolute score errors for different estimators of true mean pooled scores, estimating the scores of the 11 best TREC 8 AdHoc track manual runs, with pools drawn from the automatic runs. The base method of handling unassessed documents is to exclude them, forming condensed lists. Other details are as for Table 6.2.

and excluding unassessed documents from the ranking is a relatively neutral act. And again, because the bias is not strictly one-sided, score adjustment does not always lead to reduced error. For wide pools and smaller common topic sets, therefore, the adjusted scores are on average slightly less accurate than the unadjusted ones, though in both cases the error is slight. On narrower pools, though, and especially with larger common topic sets, score adjustment again leads to significant error reduction.

The most notable feature of score adjustment based on common topics across the above experiments is its consistency. It gives similar, low error rates whether the base method is condensed lists or assuming unassessed documents to be irrelevant, and whether the unpooled system is randomly similar to the pooled ones or is distinctive from, and markedly superior to, them. In contrast, score adjustment from systems is not robust to systematic differences between the unpooled and pooled systems, while the error from unadjusted scores varies considerably depending on system composition and base scoring method. Adjusted scores are consistently more accurate than unadjusted where the base method is to assume unassessed documents to be irrelevant. Where condensed lists are used, some situations make unadjusted scores are more accurate; here, that was the case where the unpooled systems was distinct and superior to the pool, and the pool width was wide. But this is a judgment that can only be made with oracular knowledge of the evaluated systems: the working evaluator does not know the true relationship between the pooled and the unpooled systems, and so cannot judge how reliable unadjusted scores are likely to be. The error for unadjusted scores can be quite wide; for adjusted scores, it is invariably narrow, at least on average, provided only that the random-sampling hypothesis applies to the common and pooled topics.

6.4 Summary

Bias in scoring retrieval systems due to incomplete relevance assessments is a significant issue, due both to increases in corpus size and a desire to reduce per-topic assessment effort. The traditional method has been to assume that unassessed documents are irrelevant, but this is increasingly biased against unpooled systems with increasing incompleteness of the assessment set. An alternative proposal is to exclude unassessed documents from the ranking, producing condensed lists; this method, however, is biased in favour of unpooled systems, since their unassessed documents are more likely to be irrelevant than their assessed ones. Moreover, the degree of bias in each of the above scoring methods varies depending on metric, pool width, pool depth, collection, and system set. It is therefore not possible to derive a single estimate of bias, or even a parameterized set of estimates, independent of the particular evaluation context.

In this chapter, we have proposed, analyzed, and empirically assessed two methods of bias estimation, and hence of score adjustment to correct pooling bias. The first method, bias estimation from systems, involves performing a leave-one-out simulation on the fully-pooled systems, to observe the pooling bias they suffer. The mean of these observed biases can then be employed as an adjustment factor for the score of the unpooled system. This method requires no additional assessment. However, it relies on the assumption that the unpooled system is randomly similar to the pooled ones. If the unpooled system is systematically different, then this method of bias estimation is unreliable. And, in practice, research and development will generally have been focused on trying to ensure that the new, unpooled system is indeed different from, and hopefully better than, the existing, pooled ones.

Rather than estimating bias based on systems, we have proposed that pooling bias should be estimated based on a set of common topics. These topics are fully assessed on all systems, existing and new. The degree of pooling bias is then estimated directly on the common topics, and the resulting estimate used to adjust scores of the new system on the existing topics, for which it is unpooled. Bias estimation from common topics is robust to systematic differences between the new and existing systems; the only assumption required is the (generally realistic) one that the common and existing topics have been randomly co-sampled. The error of unpooled scores adjusted in this way is consistently low, and generally much lower than that of unadjusted scores. Estimation from common topics does require that such a set of topics either exist or be created; but in a dynamic evaluation environment, it is likely that such sets will emerge as part of the ongoing evaluation process.

Chapter 7

A Similarity Measure for Indefinite Rankings

So far in this thesis, we have been concerned with evaluating the effectiveness of retrieval systems. We have assumed that relevance assessments, if only incomplete ones, are either available or will be made; we have dealt with document rankings only once they have been converted to relevance vectors; and we have compared rankings solely by their effectiveness scores. There are, however, many circumstances in which we wish to compare document rankings directly, without relevance assessments. In some cases, the similarity between rankings can act as a proxy, where relevance assessments have not yet been made, for a much more expensive effectiveness evaluation. If two systems produce very similar rankings, they can hardly differ much in effectiveness; if one system is meant to approximate another, perhaps more efficiently, then a great difference in rankings is a cause for concern. In other cases, document rankings are compared with no concern for effectiveness. We may be interested in which search engines give the most similar results to each other; or in how much, and in what way, the results of one search engine change over time.

To compare document rankings in a systematic, objective, and repeatable way requires an appropriate measure of rank similarity. This may seem a well-studied problem, amenable to the use of existing rank correlation coefficients such as Kendall's τ or Spearman's ρ. But there are some features of document rankings that make existing measures inapplicable. Most immediately, the standard rank correlation measures assume that the rankings are conjoint, whereas document rankings are substantially non-conjoint. Additionally, we care about differences at the top of document rankings more than those further down, on the first page of search results more than on the tenth; but most existing correlations are not top-weighted. Furthermore, the rank at which we cut off the comparison is often arbitrary. There may be millions of search results, of which we choose to compare only the first hundred; but there is nothing special about the choice of cutoff depth, and we would prefer it not to become a dependency in the measure used.

In this chapter, we assert that the features of document rankings—non-conjointness, top-weightedness, and indefiniteness—together determine a particular class, that of *indefinite rankings*. We argue that none of the existing rank similarity measures are appropriate measures for indefinite rankings. And we propose a new similarity measure on indefinite rankings, called *rank-biased overlap* (RBO). Nor are the features

of indefiniteness confined to document rankings. On the contrary, indefinite rankings are widely encountered in research and in daily life, making RBO a measure of wide application.

The chapter is laid out as follows. In Section 7.1, we describe the features of indefinite rankings, and prescribe criteria for a suitable similarity measure on such rankings. In Section 7.2, we discuss existing rank similarity measures, in particular those that are able to handle non-conjoint rankings; we conclude, however, that none of these are adequate measures for indefinite rankings. Our new similarity measure, rank-biased overlap, is presented and analyzed in Section 7.3. And in Section 7.4, we apply RBO to real and simulated comparisons, and compare its behaviour with that of existing measures on non-conjoint rankings.

7.1 Indefinite rankings

Document rankings are produced by search engines on the web, and by retrieval systems in the laboratory. We discuss the latter here, but our remarks apply to the former, too. A major branch of retrieval research is that of efficiency; how to produce the same, or at least nearly the same, results with less processing effort. In efficiency studies, there is generally an objective ranking, produced by full evaluation, and one or more observed rankings, produced by efficient short-cuts. One form of short-cut is query pruning, which sets a limit on the amount of working space, in the form of document accumulators, that is allocated to query processing. The only documents that are fully evaluated are those which, on an initial evaluation, seem most likely to achieve a high similarity score for the query (Lester, Moffat, Webber, and Zobel, 2005). Figure 7.1 gives part of the output of a query pruning experiment, in which the ranking produced by a full evaluation (on the left) is being compared with those produced by two pruned evaluations (on the right), one more strictly pruned than the other. The researcher wishes to know how much impact the different pruning levels are having on the fidelity of the ranking. Retrieval effectiveness may be the final arbiter, but assessing effectiveness is expensive. The researcher wants initially to compare rankings over perhaps thousands of queries, both to quantify the degree of change, and to determine which queries are most strongly affected. This is one of the scenarios in which a similarity measure on document rankings is required.

There are several characteristics of document rankings, such as those shown in Figure 7.1, that should be observed before selecting a measure of rank similarity. One is that the documents at the top of each ranking are more important than those further down. Figure 7.1 only displays the top ten results from each system, and it is intuitive that disagreements amongst these leading results are more important than those later in the ranking. Even within these ten results, a manual inspection naturally starts by comparing the top results first. So too, the first page of results returned by a public search engine is more likely to be viewed by the user than the second, and the second than the third. Also, if more subtly, there is usually a steeper gradient in potential relevance between the few, highly similar documents and the many, moderately similar ones, as the similarity scores in Figure 7.1 suggest. The bias in attention and importance towards the top of the rankings calls for the similarity measure used to compare them to itself be *top-weighted*; that is, to exact harsher penalties for differences at the top of the ranking than for differences further down.

A second characteristic of document rankings is that they are generally *incomplete*; they do not provide a full ranking across all the elements in their domains, that is, across

rnk	docid	sim
1	FBIS4-13392	6.44
2	FT931-12892	6.13
3	FT921-11935	5.66
4	FT933-7566	5.62
5	FT924-12615	5.49
6	FBIS4-59400	5.46
7	FT943-14288	5.31
8	FT941-373	5.30
9	FT923-12606	5.29
10	FBIS4-11824	5.29
...	...	...

(a) Full Evaluation

docid	sim
FBIS4-13392	6.44
FT931-12892	6.13
FT923-12606	5.29
FBIS4-11824	5.29
FBIS4-38863	5.24
FBIS4-46500	5.22
FBIS4-39925	5.19
FBIS4-46560	5.15
FBIS4-61085	5.00
FBIS3-55156	4.99
...	...

(b) 1000 Accumulators

docid	sim
FBIS4-13392	6.44
FT931-12892	6.13
FT921-11935	5.66
FT933-7566	5.62
FT943-14288	5.31
FT923-12606	5.29
FBIS4-11824	5.29
FBIS4-38863	5.24
FT942-2178	5.23
FBIS4-46500	5.22
...	...

(c) 400 Accumulators

Figure 7.1: Runs returned by an experimental retrieval system to a test topic, under (a) full evaluation of index information; and (b, c) two different abbreviated evaluations. Each row is a document that the system has returned for the particular query. The first column gives the document identifier, by which the document is represented internally. The second column gives the similarity score calculated between each document and the query. The leftmost column gives the document's rank in the result; the rank is determined by the similarity score.

all the documents in the corpus (for web search engines, all pages on the web). Thus, when compared, such rankings are *non-conjoint*; some elements turn up in one ranking but not the other. Indeed, in the case of web search results, it is not clear that even the domains from which the rankings are drawn are conjoint, since different engines may have different policies on what constitutes an indexable document. For retrieval systems under experimental control, the corpus behind each ranking is in general the same; nevertheless, it is hardly ever the case that the document ranking produced by an experimental system is exhaustive. Any similarity measure on such rankings must be able to handle non-conjointness.

The incompleteness of document rankings arises because, even if conceptually the entire corpus could be ranked, only the prefix of each ranking is returned. But the length of this prefix is essentially arbitrary. The experimenter might set it to 100, or to $1{,}000$, or to some other value, without essentially changing the nature of the list or of the comparison. Moreover, the prefix length may not be entirely under the researcher's control. A public engine, or even an experimental system, may return shorter rankings for some queries than for others, and some of these may fall short of the prefix length chosen by the researcher. Thus, even within the one experiment, rankings of disparate lengths may be under comparison. We refer to the arbitrariness and variability of the ranking prefix length as the ranking's *indefiniteness*. Any similarity measure upon such rankings should be flexible to their indefiniteness; it should not embed the length of the prefix in the measure itself, nor give incomparable results for different prefix lengths. As will be seen later, an implication of comparability across cutoff depths is that the score at a given cutoff depth must place bounds upon the score achieved at greater depths. Otherwise, even the one ranking will not be comparable at different cutoffs.

The three qualities described above, of top-weightedness, incompleteness, and indefiniteness, are related. It is because the top of the ranking is the most important part

that we only need to see a prefix of it, thus rendering it incomplete; and similarly, it is because most of the weight attaches to the top of the ranking that the depth of the prefix can vary without fundamentally changing the nature of the metric. A ranking that has these three qualities of top-weightedness, incompleteness, and indefiniteness, is referred to here as an *indefinite ranking*, and a measure of similarity between two such rankings as an *indefinite rank similarity measure*. We set out in this chapter to show that existing rank similarity measures are not adequate indefinite rank similarity measures, and then to propose a new measure, rank-biased overlap, that is.

7.2 Non-conjoint rank similarity measures

Section 3.3.6 introduced several rank similarity metrics, unweighted and weighted, on conjoint rankings, such as the unweighted Kendall's τ measure (Kendall, 1948), and the weighted variant, τ_{AP}, proposed by Yilmaz et al. (2008a). Such measures cannot directly be used to compare non-conjoint rankings. In this section, we survey rank similarity measures that handle non-conjointness. First, in Section 7.2.1, we describe the unweighted measures on non-conjoint rankings that have been proposed in the literature; then, in Section 7.2.2 we examine the (much smaller) class of weighted non-conjoint measures.

7.2.1 Unweighted non-conjoint measures

A common approach to deriving a similarity measure on non-conjoint rankings is to modify a conjoint similarity measure to handle list non-conjointness. One such modification is simply to ignore non-conjoint elements. This approach is unsatisfactory, however, since the presence of non-conjoint elements provides information that ignoring them throws away. Consider the two rankings $\langle$`ab???`$\rangle$ and $\langle$`a???b`$\rangle$, where `?` denotes a non-conjoint element. Removing these non-conjoint elements leaves both rankings as $\langle$`ab`$\rangle$. A rank similarity measure applied to these condensed rankings would regard them as identical; but the presence of the non-conjoint elements makes it clear that they are not.

A more satisfactory method of handling non-conjointness starts by viewing the lists as the truncated prefixes of otherwise conjoint rankings; what are called *top-k* lists, where k is the depth of the prefix. Elements that appear in only one of the two top-k lists are assumed to appear somewhere beneath depth k in the other list. Placing unranked items below rank k is the approach taken by Fagin et al. (2003). They adapt both Kendall's τ and Spearman's footrule in this way to handle top-k lists. For τ_k, the top-k version of τ, if element i appears in ranking S but not ranking T, it is assumed to be ranked beneath every item that does appear in ranking T. The only ambiguity occurs if elements i and j both appear in ranking S, but neither appear in ranking T. In this case, Fagin et al. provide for a parameterizable penalty of between 0 (assumed concordant) and 1 (assumed discordant). They propose that the default value for this penalty should be 0, as this fixes the score for conjoint but reversed rankings as close as possible to half way between the scores for identical rankings, which is 1, and disjoint rankings, which is -1. A top-k version of Spearman's footrule, f_k, is similarly defined.

The desideratum stated by Fagin et al. that conjoint but reversed top-k rankings should score roughly half way between identical and disjoint is not a compelling one. How close a relatedness reverse conjointness indicates depends on how large k is in relation to the full list size n. Moreover, conjoint but reversed to depth k is more a

peculiarity than a meaningful characteristic for top-k lists, since by definition it cannot continue to be true if the evaluation is then extended to depth $k + 1$. Partly at fault is a desire to produce a measure that is similar in form to correlation measures on conjoint lists; similarly, having a negative score for a top-k measure is hardly meaningful. More fundamentally, though, the special treatment of reversed top-k rankings does not properly reflect the fact that these are indefinite rankings, and that the choice of k as the cutoff point is essentially an arbitrary one.

In addition to Kendall's τ and Spearman's footrule, Fagin et al. describe a top-k variant of Spearman's ρ. The treatment of non-conjoint elements is similar to that for the other methods. Fagin et al. define the notion of equivalence classes over rank similarity measures; τ_k and the f_k are in the same equivalence class, but ρ_k does not fall into this class.

Goodman and Kruskal's γ is a correlation coefficient related to Kendall's τ, in which tied items are effectively ignored (Goodman and Kruskal, 1954). Fagin et al. also extend γ to the top-k case by regarding the pair ij both appearing in list S but neither appearing in list T as tied, and therefore ignoring it.

Bar-Ilan (2005) and Bar-Ilan et al. (2006) adapt Spearman's ρ and Spearman's footrule respectively to the top-k case by excluding non-conjoint elements (rather than treating them as occurring beyond depth k) and calculating the coefficients on the condensed lists. Bar-Ilan et al. point out the loss of information that condensing lists in this way entails.

7.2.2 Weighted non-conjoint measures

The measures τ_k and f_k are not top-weighted, but similar assumptions about the location of non-conjoint elements could be applied to top-weighted conjoint rank measures to derive weighted top-k measures. Weightedness makes the assumption of unlisted elements being ranked beyond rank k more complex in its implications, though. For instance, in τ_{AP}, when randomly selecting an item i and a higher-ranked item j, the question arises of whether the items beyond depth k are to be regarded as above or below each other. In particular, τ_{AP} does not (as currently defined) handle tied items, so the non-conjoint elements cannot simply be placed at rank $k + 1$. Instead, Yilmaz et al. (2008a) propose that any such elements be excluded; but this loses information about implied misorderings, as described above.

Most of the measures discussed so far, both the non-conjoint ones introduced in Section 7.2.1 and the conjoint ones described in Section 3.3.6, have been founded on the notion of correlation. When dealing with non-conjoint lists, it is also possible, and arguably more natural, to start instead from set intersection. A simple similarity measure on top-k lists would be the size of the intersection, or *overlap*, between the two rankings, calculated as the proportion of the ranking length; that is, $|S \cap T|/k$. Of course, such a measure, while directly handling non-conjointness, takes no notice of ranking, and therefore is not top-weighted.

The idea of overlap can be extended by considering, not simply the overlap at depth k, but the cumulative overlap at increasing depths. Under this principle, the approach is to calculate the overlap for each $d \in \{1 \dots k\}$, and then average those overlaps to derive the similarity measure. This measure is described by Fagin et al. (2003) and called the intersection metric, and was simultaneously described by Wu and Crestani (2003) and named average accuracy. We refer to it as average overlap (AO). Because of its cumulative nature, AO is top-weighted: rank 1 is included in every subset, rank 2 in every subset but the first, and rank r in subsets r through k but not 1 through $r - 1$.

d	$S_{:d}$	$T_{:d}$	$A_{S,T,d}$	$\mathrm{AO}(S,T,d)$
1	⟨a⟩	⟨z⟩	0.000	0.000
2	⟨ab⟩	⟨zc⟩	0.000	0.000
3	⟨abc⟩	⟨zca⟩	0.667	0.222
4	⟨abcd⟩	⟨zcav⟩	0.500	0.292
5	⟨abcde⟩	⟨zcavw⟩	0.400	0.313
6	⟨abcdef⟩	⟨zcavwx⟩	0.333	0.317
7	⟨abcdefg⟩	⟨zcavwxy⟩	0.286	0.312
n	⟨abcdefg...⟩	⟨zcavwxy...⟩	?	?

Figure 7.2: Illustrative calculation of the average overlap (AO) of two lists to increasing depths, along with their proportional overlap or agreement A at each depth. Average overlap continues to increase even as agreement decreases, and the value at depth k does not bound the value at arbitrary depth $n > k$. The notation used is described in more detail in Section 7.3.1.

Thus, AO is the first of the measures we have examined that both handles non-conjoint lists and is top-weighted, and indeed is one of the very few such measures described in the literature. Figure 7.2 gives a sample calculation.

Although average overlap is closer to a satisfactory indefinite rank similarity measure than any of the previous alternatives, it fails our criteria for an indefinite measure because the depth of evaluation is implicitly embedded in the measure. The score at prefix depth k sets no bounds on the score that would occur if the prefix were lengthened indefinitely. The reason for this is the measure's non-convergence. The weight of the infinite tail always dominates that of the finite prefix, no matter how long the prefix is. A proof is given in Appendix A.2; intuitively, we see that each overlap to depth k has weight $1/k$ under AO@k, but weight $1/\infty$ under AO@∞. Thus, prefix evaluation sets no bounds on the full score: after comparing k elements, the AO@∞ score could still be anywhere in the range $[0, 1]$, not matter how large k is.

Average overlap has another peculiarity: it is not monotonic with agreement. Finding greater agreement with deeper evaluation does not necessarily lead to a higher score, nor finding decreased agreement to a lower one. For instance, in Figure 7.2, the elements newly revealed at depths 4 through 6 are all disjoint, yet the AO score increases. This counter-intuitive behaviour occurs because, in calculating AO, the contribution of each overlap at depth d is only considered up to k, whereas in fact it continues to contribute up to n as n goes to infinity; increasing the evaluation depth k thus captures more of this residual contribution. To avoid this non-monotonicity, the contribution of each overlap would have to be calculated as depth goes to infinity; but because average overlap is non-convergent, the contribution to infinity is undefined. Both because of its non-convergence and because of its non-monotonicity, average overlap scores are essentially incomparable between different prefix depths.

Bar-Ilan et al. (2006) describe and employ a measure M which is the normalized sum of the difference in the reciprocal of the ranks of each item in the two lists, with items not ranked in one list assumed to occur at depth $k + 1$ in that ranking. Like AO, this measure is top-weighted and handles non-conjointness, but is dependent on the cutoff depth k.

Buckley (2004) proposes the AnchorMAP measure, which is based upon the retrieval effectiveness evaluation metric, (mean) average precision (MAP), described in

Section 3.2.2. One of the rankings under comparison is chosen as the objective ranking, and its first s documents are treated as relevant; Buckley suggests $s = 30$ as a reasonable value. The MAP score of the other ranking is then calculated to depth k, based on these artificial relevance judgments. AnchorMAP is asymmetric. It is top-weighted, but weights are not fixed for ranks. The metric is non-monotonic both in s and k.

Recently, Sun et al. (2010) have proposed a new rank similarity measure, the expected weighted Hoeffding distance, d_w. The d_w measure is based on an edit (or "earth mover's") distance between permutations, where the cost of different moves can be weighted, allowing for arbitrary assignment of weights to different sections of the ranking (such as the top). Like RBO (see Section 7.3.5), d_w is a true metric. Non-conjointness is handled probabilistically, by calculating the expected similarity over all possible permutations of the domain, given the observed top k results in the ranking; a computationally tractable form of this calculation is provided. The monotonicity of d_w in increasing depth is unclear: the expected permutations are calculated over a universe of known items (though not necessarily the union of observed prefixes), so the appearance with increasing prefix depth of previously unknown items would seem to destroy monotonicity. Sun et al. deploy d_w for data visualization and clustering. They suggest that a rank similarity measure should be evaluated in part on its suitability as a clustering metric; this is an interesting suggestion and merits further attention.

Another recent proposal for a rank similarity measure is that of Kumar and Vassilvitskii (2010). They generalize Kendall's τ and Spearman's footrule with both position weights (allowing higher ranks to carry more emphasis) and element weights (incorporating, say, relevance information about individual documents). Like the original measures they are based on, however, the generalized measures do not handle non-conjoint rankings.

A referee of Webber, Moffat, and Zobel (2010) suggested an alternative mechanism, based on a rank-weighted evaluation metric such as discounted cumulative gain (DCG) (Järvelin and Kekäläinen, 2002). In a rank-weighted metric, as described in Section 3.2.3, each rank i is assigned a fixed weight w_i, and the document at that rank makes a contribution $w_i \cdot r_i$ to the ranking's effectiveness score, where r_i is the document's assessed degree of relevance. A similarity measure between two rankings S and T can then be derived by assigning fractional relevancies to documents based on their rank weight in S, and then using these relevancies to calculate the effectiveness metric on T. Such a measure would be symmetric, and seems likely to possess some of the properties sought in an indefinite rank similarity measure, provided that rank weights are chosen so as to create a convergent measure; note that DCG's inverse-logarithmic weights do not (Section 3.2.3). The need for properties such as convergence, and the need to ensure sensible behaviour in limiting cases, means that developing an approach of this kind is not straightforward, and is an area for future investigation. How such an approach would ultimately compare with RBO as defined below is not clear.

7.3 Rank-biased overlap

We have shown that the non-conjoint rank similarity measures described in the literature do not meet the criteria we have identified for similarity measures on indefinite rankings, while the conjoint similarity measures discussed in Section 3.3.6 cannot be used at all, because of the non-conjointness of indefinite rankings. We now propose a new measure which does meet these criteria: rank-biased overlap (RBO).

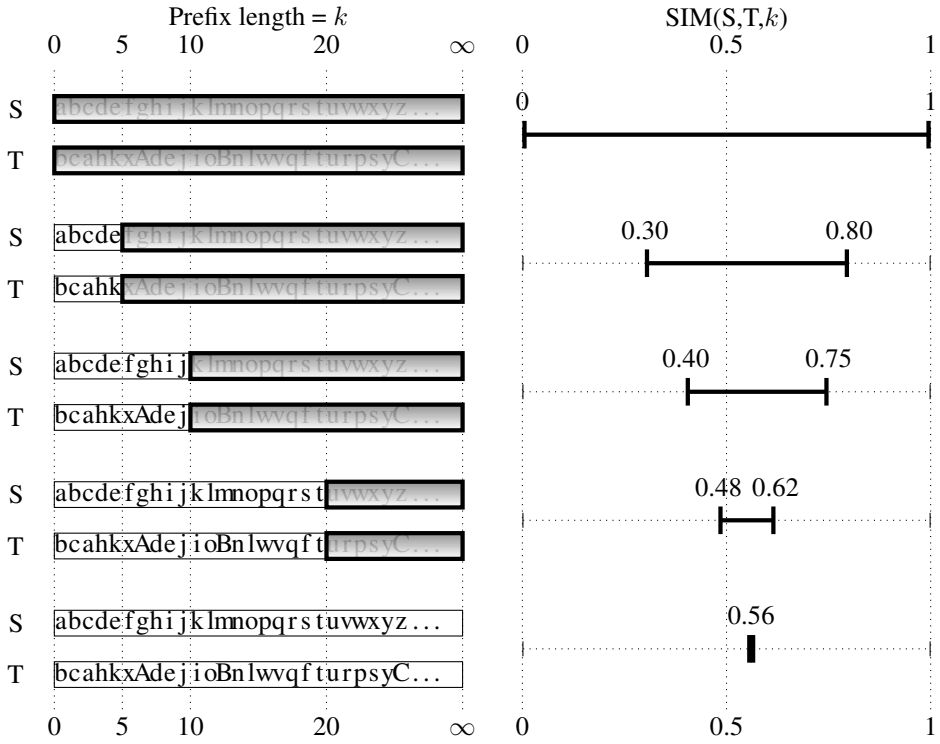


Figure 7.3: Convergence of scores with more information. Before examining either of the rankings, their similarity score could range anywhere from 0 to 1. As the length of the examined prefix k increases, the range of the possible full similarity score decreases monotonically. These ranges bound the similarity score achievable on infinite evaluation.

Our core idea is to define a measure on the similarity of the full rankings, and design the measure such that a partial, prefix evaluation bounds the value that a full evaluation would produce. The deeper the prefix that is examined, the narrower the bounds on the full score become. The idea is illustrated in Figure 7.3. With no elements examined, the similarity score between the rankings could take any value in the measure's legal range; say, anywhere between 0 and 1. After seeing the first 5 elements in each ranking, the range of possible scores of a full evaluation is narrowed to between, say, 0.3 and 0.8. And after extending the prefixes to depth 20, the range is narrowed to 0.48 and 0.62, as illustrated in the fourth segment of Figure 7.3. The key is to choose a sequence of decreasing weights over the depths of the comparison, such that the sum of the weights is convergent; that is, so that the weight of the unseen, conceptually infinite tail of the lists is limited, and does not dominate the weight of the seen, finite prefix. Such a weighting scheme, besides being attractive mathematically, is justified representationally by the assumptions underlying indefinite rankings; that is, that the interest of the consumer of the ranking is sufficiently top-weighted for a truncated ranking to be satisfactory.

Rank-biased overlap is an overlap-based measure, superficially similar to average overlap. The crucial difference, though, is that it biases the proportional overlap at each depth by a convergent set of weights, in the manner described in the preceding

paragraph. Because of the convergence of RBO's weights, the infinite tail does not dominate the finite head. Therefore, similarity assessment using RBO consists of using prefix evaluation to set upper and lower bounds (Section 7.3.2) on the score that full evaluation (that is, comparison to infinite depth) could achieve (Section 7.3.1)—precisely as illustrated in Figure 7.3. In Section 7.3.3 we derive the weight of each rank under RBO, and therefore the weight of the prefix. The precise full RBO score is, of course, not knowable without evaluation to infinite depth; however, in situations where a single value is needed, a reasonable point estimate can be extrapolated (Section 7.3.4). Because RBO is a similarity, not a distance, measure, it is not a metric; however, $1 - \text{RBO}$ is a metric, as we prove in Section 7.3.5. Finally, Section 7.3.6 considers the handling of ties and of rankings of different lengths.

7.3.1 RBO on infinite lists

We begin by laying out some notation. Let S and T be two infinite rankings, and let S_i be the element at rank i in list S. Denote the set of the elements from position c to position d in list S, that is $\{S_i : c \le i \le d\}$, as $S_{c:d}$. Let $S_{:d}$ be equivalent to $S_{1:d}$, and $S_{c:}$ be equivalent to $S_{c:\infty}$. At each depth d, the *intersection* of lists S and T to depth d is:

$$I_{S,T,d} = S_{:d} \cap T_{:d} \; . \tag{7.1}$$

The size of this intersection is the *overlap* of lists S and T to depth d,

$$X_{S,T,d} = |I_{S,T,d}| \; , \tag{7.2}$$

and the proportion of S and T that are overlapped at depth d is their *agreement*,

$$A_{S,T,d} = \frac{X_{S,T,d}}{d} \; . \tag{7.3}$$

For brevity, we refer to I_d, X_d, and A_d when it is unambiguous which lists are being compared. Using this notation, average overlap can be defined as:

$$\text{AO}(S, T, k) = \frac{1}{k} \sum_{d=1}^{k} A_d \tag{7.4}$$

where k is the evaluation depth. An example calculation has already been shown in Figure 7.2.

Consider the family of overlap-based rank similarity measures of the form:

$$\text{SIM}(S, T, w) = \sum_{d=1}^{\infty} w_d \cdot A_d \tag{7.5}$$

where w is a vector of weights, and w_d is the weight at position d. Then $0 \le \text{SIM} \le \sum_d w_d$, and if w is convergent, each A_d has a fixed contribution $w_d / \sum_d w_d$ (if w is not convergent, then the denominator of this expression goes to infinity). One such convergent series is the geometric progression, where the dth term has the value p^{d-1}, for $0 < p < 1$, and the infinite sum is:

$$\sum_{d=1}^{\infty} p^{d-1} = \frac{1}{1-p} \tag{7.6}$$

Setting w_d to $(1-p)\cdot p^{d-1}$, so that $\sum_d w_d = 1$, derives rank-biased overlap:

$$\mathrm{RBO}(S,T,p) = \frac{1-p}{p}\sum_{d=1}^{\infty} p^d \cdot A_d \ . \tag{7.7}$$

Rank-biased overlap falls in the range $[0,1]$, where 0 means disjoint, and 1 means identical. The parameter p determines how steep the decline in weights is: the smaller p, the more top-weighted the metric is. In the limit, when $p = 0$, only the top-ranked item is considered, and the RBO score is either zero or one. On the other hand, as p approaches arbitrarily close to 1, the weights become arbitrarily flat, and the evaluation becomes arbitrarily deep.

Rank-biased overlap has an attractive interpretation as a probabilistic user model. Consider a user comparing the two lists. Assume they always look at the first item in each list. At each depth down the two lists, they have probability p of continuing to the next rank, and conversely probability $1-p$ of deciding to stop. Thus, the parameter p models the user's *persistence*. This concept of persistence was introduced for the retrieval effectiveness metric *rank-biased precision* (Moffat and Zobel, 2008). Once the user has run out of patience at depth d, they then calculate the agreement between the two lists at that depth, and take this as their measure of similarity between the lists. Let D be the random variable giving the depth that the user stops at, and $P(D=d)$ be the probability that the user stops at any given depth d. The expected value of this random experiment is then:

$$\mathbb{E}[A_D] = \sum_{d=1}^{\infty} P(D=d)\cdot A_d \ . \tag{7.8}$$

Since $P(D=d) = (1-p)\cdot p^{d-1}$, it follows that $\mathbb{E}[A_D] = \mathrm{RBO}(S,T,p)$. Indeed, this probabilistic model can be extended further by observing that A_d itself gives the probability that an element randomly selected from one prefix will appear in the other. Such probabilistic models help to interpret the meaning of the similarity scores achieved.

7.3.2 Bounding RBO from prefix evaluation

Rank-biased overlap is defined on infinite lists. Because it is convergent, the evaluation of a prefix sets a minimum and a maximum on the full score, with the range between them being the residual uncertainty attendant upon prefix, rather than full, evaluation. In this section, formulae for the minimum score, $\mathrm{RBO}_{\mathrm{MIN}}$, and the residual, $\mathrm{RBO}_{\mathrm{RES}}$, are derived.

Simply calculating Equation 7.7 to prefix depth k (let us call this RBO@k) sets a lower bound on the full evaluation, but not a tight one. Indeed, if RBO@k > 0, it is certain that RBO $>$ RBO@k. This is because the overlap in the prefix also contributes to all overlaps at greater depths; the same problem was observed with average overlap in Figure 7.2. More formally, for all $d > k$, $I_d \supseteq I_k$, meaning $X_d \geq X_k$ and A_d is at least X_k/d. Thus, even if all items beyond the prefix turned out on full evaluation to be disjoint, the sum of the agreements at depths beyond k would be:

$$\frac{1-p}{p}\sum_{d=k+1}^{\infty}\frac{X_k}{d}\cdot p^d \ . \tag{7.9}$$

d	$S_{:d}$	$T_{:d}$	$\min(A_d)$	$\max(A_d)$	weight
1	⟨a⟩	⟨c⟩	$0/1$	$0/1$	p^0
2	⟨ab⟩	⟨cb⟩	$1/2$	$1/2$	p^1
3	⟨abd⟩	⟨cbe⟩	$1/3$	$1/3$	p^2
4	⟨abd?[c]⟩	⟨cbe?[a]⟩	$1/4$	$3/4$	p^3
5	⟨abd??[ce]⟩	⟨cbe??[ad]⟩	$1/5$	$5/5$	p^4
6	⟨abd???[cef]⟩	⟨cbe???[adf]⟩	$1/6$	$6/6$	p^5
⋮	⋮	⋮	⋮	⋮	⋮
d	⟨abd...⟩	⟨cbe...⟩	$1/d$	d/d	p^{d-1}

Figure 7.4: Minimum and maximum agreements between two indefinite lists at different depths, with evaluation finishing at depth 3. Unseen items for ranks 4 through d are marked as ?. Example hypothetical maximally agreeing elements for these ranks are shown in square brackets.

To set a true minimum on full evaluation, Equation 7.9 is added to the RBO@k score. The infinite sum can be resolved to finite form by the useful equality:

$$\sum_{i=1}^{\infty} \frac{p^i}{i} = \ln \frac{1}{1-p}\ ,\ 0 < p < 1 \tag{7.10}$$

which is derived by integrating both sides of Equation 7.6. After some rearrangement, we arrive at:

$$\mathrm{RBO}_{\mathrm{MIN}}(S,T,p,k) = \frac{1-p}{p}\left(\sum_{d=1}^{k}(X_d - X_k)\cdot\frac{p^d}{d} - X_k\ln(1-p)\right) \tag{7.11}$$

where k is the length of the prefix. The $\mathrm{RBO}_{\mathrm{MIN}}(S,T,p,k)$ value gives a tight lower bound on the full $\mathrm{RBO}(S,T,p)$ score. It follows from this that $\mathrm{RBO}_{\mathrm{MIN}}(S,T,p,k)$ is monotonically non-decreasing on deeper evaluation; that is,

$$\forall j > 0,\ \mathrm{RBO}_{\mathrm{MIN}}(S,T,p,j+1) \geq \mathrm{RBO}_{\mathrm{MIN}}(S,T,p,j)\,. \tag{7.12}$$

Prefix evaluation can also be used to derive a tight maximum on the full RBO score; the residual uncertainty of the evaluation is then the distance between the minimum and maximum scores. The maximum score occurs when every element past prefix depth k in each list matches an element in the other list, beginning with those elements in the prefix that were previously unmatched. Figure 7.4 illustrates this with an example. The prefix length is $k = 3$, and the overlap X_k at this depth is 1. At each successive depth, two more elements are added, one to each ranking. Therefore, the maximum overlap increases by two until agreement is complete, which occurs at depth $f = 2k - X_k$. Beyond that depth, agreement is fixed at 1. The residual RBO value is therefore:

$$\mathrm{RBO}_{\mathrm{RES}}(S,T,p,k) = \frac{1-p}{p}\left(\sum_{d=k+1}^{f}\frac{2(d-k)}{d}p^d + \sum_{d=f+1}^{\infty}\left(1-\frac{X_k}{d}\right)p^d\right)\,. \tag{7.13}$$

Some rearranging, and again using Equation 7.10 to reduce the infinite sum, gives:

$$\mathrm{RBO}_{\mathrm{RES}}(S,T,p,k) = p^f + \frac{1-p}{p}\left\{2\sum_{d=k+1}^{f}\frac{(d-k)p^d}{d} - X_k\left[\ln\frac{1}{1-p} - \sum_{d=1}^{f}\frac{p^d}{d}\right]\right\}. \tag{7.14}$$

One might prefer the residual uncertainty of prefix evaluation to be dependent only on the prefix length, not on prefix content. This is not the case with RBO, as prefix agreement determines how long it takes before the difference between the maximum and minimum agreements at subsequent depths d reaches the stationary value of $1 - X_k/d$, as well as this stationary value itself. It is possible, though, to set a range on the values that $\mathrm{RBO}_{\mathrm{RES}}$ can take for a given prefix length, irrespective of prefix contents. The residual will be smallest when $X_k = k$, that is, when the prefix is conjoint. In this case, Equation 7.13 becomes:

$$\begin{aligned}\mathrm{RBO}^{\min}_{\mathrm{RES}}(*,*,p,k) &= \frac{1-p}{p}\sum_{d=k+1}^{\infty}\left(1-\frac{k}{d}\right)p^d &(7.15)\\ &= p^k - k\cdot\frac{1-p}{p}\cdot\left(\ln\frac{1}{1-p} - \sum_{d=1}^{k}\frac{p^d}{d}\right). &(7.16)\end{aligned}$$

The residual will be largest when $X_k = 0$, that is, when the prefix is disjoint. Then, we have:

$$\begin{aligned}\mathrm{RBO}^{\max}_{\mathrm{RES}}(*,*,p,k) &= \frac{1-p}{p}\left(\sum_{d=k+1}^{2k}\frac{2(d-k)}{d}p^d + \sum_{d=2k+1}^{\infty}p^d\right) &(7.17)\\ &= 2p^k - p^{2k} - 2k\cdot\frac{1-p}{p}\cdot\sum_{d=k+1}^{2k}\frac{p^d}{d}. &(7.18)\end{aligned}$$

It also follows that $\mathrm{RBO}^{\min}_{\mathrm{RES}}$ occurs when $\mathrm{RBO}_{\mathrm{MAX}} = 1$, and $\mathrm{RBO}^{\max}_{\mathrm{RES}}$ occurs when $\mathrm{RBO}_{\mathrm{MIN}} = 0$. These formulae are useful in experimental planning. For instance, if two search engines are to be compared on multiple queries, then a first-page or ten-result evaluation with $p = 0.9$ will give a maximum residual of 0.254, for a range of 0.000 to 0.254, and a minimum residual of 0.144, for a range of 0.856 to 1.000. These residuals can be decreased either by examining more results or by using a lower value of p.

Prefix evaluation, then, can be used to set tight bounds upon the full RBO score, meeting our main criteria for a similarity measure on indefinite rankings. The upper and lower limits are monotonically non-increasing and non-decreasing respectively as evaluation continues further down the two lists, in the manner illustrated in Figure 7.3. Also, $\mathrm{RBO}_{\mathrm{RES}}$ is monotonically decreasing with evaluation depth: the greater the information about the two lists, the smaller the degree of uncertainty about their full similarity. These monotonic properties are what qualifies RBO to be a satisfactory similarity measure on indefinite rankings, and ensure that the RBO measure provides consistent values for whatever evaluation depth k happens to be chosen, and maintains consistency as this evaluation depth increases. Moreover, the score at any depth of partial evaluation gives strict limits on the score that would be achieved by full evaluation. In contrast, top-k measures are measures only on the lists to depth k, and provide no bounds on the value of full evaluation. Even with partial evaluation, RBO is a measure on the full lists.

7.3.3 Rank weights under RBO

The agreement at each depth d under RBO is assigned a weight. This weight, however, is not the same as the weight that the elements at rank d themselves take, as these elements contribute to multiple agreements. In this section, we derive a formula for the weight of each rank under RBO. From this, the weight of a prefix can be calculated, which in turn helps guide the choice of the p parameter in the RBO evaluation

The pair of elements at depth d makes no contribution to partial agreements prior to d, takes up $1/d$ th of A_d, $1/(d+1)$ th of A_{d+1}, and so forth. Their precise contribution to the overall score depends on which depth, if any, they are matched at. Consider the difference in the final score between, on the one hand, both elements at depth d being matched at or prior to depth d (maximum agreement), and, on the other, neither element being matched at infinite depth (minimum agreement). We will refer to this difference as the *weight* of rank d, denoted as $W_{\mathrm{RBO}}(d)$. Accounting for the weighting of the agreements $w_d = (1-p)\cdot p^{d-1}$ (Equation 7.7), the weight of rank d under RBO is therefore:

$$W_{\mathrm{RBO}}(d) = \frac{1-p}{p}\sum_{i=d}^{\infty}\frac{p^i}{i} \tag{7.19}$$

The weight of the prefix of length d, $W_{\mathrm{RBO}}(1:d)$, is then the sum of the weights of the ranks to that depth:

$$W_{\mathrm{RBO}}(1:d) = \sum_{j=1}^{d} W_{\mathrm{RBO}}(d) = \frac{1-p}{p}\sum_{j=1}^{d}\sum_{i=j}^{\infty}\frac{p^i}{i} \tag{7.20}$$

which after some rearrangement, and using Equation 7.10 to resolve the infinite sum, gives:

$$W_{\mathrm{RBO}}(1:d) = 1 - p^{d-1} + \frac{1-p}{p}\cdot d\cdot\left(\ln\frac{1}{1-p} - \sum_{i=1}^{d-1}\frac{p^i}{i}\right)\ . \tag{7.21}$$

The weight of the tail, $W_{\mathrm{RBO}}(d+1:\infty)$, is $1-W_{\mathrm{RBO}}(1:d)$. And since $W_{\mathrm{RBO}}(1:d)$ is invariant on the length of the list, it follows that the weight of the infinite tail does not dominate that of the finite head.

Equation 7.21 helps inform the choice of the parameter p, which determines the degree of top-weightedness of the RBO metric. For instance, $p = 0.9$ means that the first 10 ranks have 86% of the weight of the evaluation; to give the top 50 ranks the same weight involves taking $p = 0.98$ as the setting. Thus, the experimenter can tune the metric to achieve a given weight for a certain length of prefix.

7.3.4 Extrapolation

Definitions of $\mathrm{RBO}_{\mathrm{MIN}}$ and $\mathrm{RBO}_{\mathrm{RES}}$ were formulated in Section 7.3.2. The RBO score can then be quoted either as base+residual or as a min–max range. For many practical and statistical applications, though, it is desirable or necessary to have a single score or point estimate, rather than a range of values.

The simplest method is to take the base RBO value as the single score for the partial evaluation. The base score gives the known similarity between the two lists, the most that can be said with certainty given the information available. The base score,

however, is dependent on the evaluation depth, k. The highest base score that can be achieved for depth k evaluation using persistence p is:

$$1 - p^k - \frac{k(1-p)}{p}\left(\sum_{d=1}^{k}\frac{p^d}{d} + \ln(1-p)\right) \tag{7.22}$$

which, for large p and small k, is well short of 1. There are practical situations in which a list is conceptually indefinite but where only the first few items are available. For instance, if two search engines each only supply 7 results to a query, and the p parameter employed is 0.9, then even if both results lists are identical (to the supplied depth), the base RBO score will only be 0.767. In such situations, base RBO can easily become a measure of result list length, not difference.

An alternative formulation for a single RBO score is to extrapolate from the visible lists, assuming that the degree of agreement seen up to depth k is continued indefinitely. Denote as RBO_{EXT} the result of such an extrapolation. To derive a direct formula for RBO_{EXT}, we start from Equation 7.9, which gives the adjustment to the RBO value, calculated on the k seen items, to make it a true minimum value. The assumption for the lower bound is that the remaining items are all non-conjoint, so that the agreement at ranks $r > k$ is X_k/r. Instead, extrapolation assumes that the degree of agreement seen at k is expected to continue to higher ranks, that is, that for $r > k, A_r = X_k/k$. The inferred agreement values may not in reality be precisely possible, because they would require fractional overlap. Consider as an analogy, though, that the expectation of a random variable does not have to be a possible value of that variable; for instance, the expected value of rolling a fair six-sided die is 3.5. Constant agreement considerably simplifies our formulae, resulting in:

$$\text{RBO}_{\text{EXT}}(S,T,p,k) = \frac{X_k}{k}\cdot p^k + \frac{1-p}{p}\sum_{d=1}^{k}\frac{X_d}{d}\cdot p^d\,. \tag{7.23}$$

Note that this is not equivalent to simply extrapolating a score between the numeric values of RBO_{MIN} and RBO_{MAX}. Since those scores are weighted to higher ranks, such an extrapolation would also be weighted to the agreement observed in higher ranks. Instead, RBO_{EXT} extrapolates out from A_k, that is, the agreement observed at evaluation depth k.

Extrapolated RBO is not monotonic; it could either increase or decrease as the prefix lengthens. However, RBO_{EXT} will always increase with increasing agreement and decrease with decreasing agreement. That is, if $A_{d+1} > A_d$ then $\text{RBO}_{\text{EXT}}(d+1) > \text{RBO}_{\text{EXT}}(d)$, and conversely if $A_{d+1} < A_d$ then $\text{RBO}_{\text{EXT}}(d+1) < \text{RBO}_{\text{EXT}}(d)$, for all $d > 0$. It was noted in Section 7.2.2 that this property is not observed by average overlap. And of course, RBO_{EXT} is bounded, by RBO_{MIN} and RBO_{MAX}.

Where a point score is needed, there is the choice of RBO_{BASE} or RBO_{EXT}. In many cases, evaluation will be performed deeply enough, and p will be small enough (say, $p \le 0.9$ and depth of 50), that the residual disappears at normal reporting fidelity, leaving RBO_{EXT} and RBO_{BASE} as indistinguishable and almost-exact estimates of the true RBO score. Where the residual is noticeable, RBO_{EXT} should in general be the preferred point estimate, in part because it is less sensitive than RBO_{BASE} to the actual evaluation depth, which may vary between different ranking pairs in the one experiment. For noticeable residuals, the full reporting format is $\text{RBO}_{\text{EXT}}[\text{RBO}_{\text{MIN}} - \text{RBO}_{\text{MAX}}]$.

7.3.5 Metricity

Since RBO measures similarity, not distance, it is not a metric. However, RBO can be trivially turned into a distance measure, rank-biased distance (RBD), by RBD $=$ $1 -$ RBO. We now prove that RBD is a metric.

Proposition 7.1 *RBD is a metric.*

Proof. Since RBD is clearly symmetric, it is sufficient to show that the triangle inequality holds, that is,

$$\forall R, S, T,\ \mathrm{RBD}(R, T, p) \leq \mathrm{RBD}(R, S, p) + \mathrm{RBD}(S, T, p) \cdot \tag{7.24}$$

Now

$$\begin{aligned} \mathrm{RBD}(S, T, p) &= 1 - \mathrm{RBO}(S, T, p) \\ &= 1 - \frac{1-p}{p} \sum_{d=1}^{\infty} \frac{|S_{:d} \cap T_{:d}|}{d} \cdot p^d \\ &= \frac{1-p}{p} \sum_{d=1}^{\infty} \frac{|S_{:d} \triangle T_{:d}|}{2d} \cdot p^d \end{aligned} \tag{7.25}$$

where $\triangle$ is symmetric difference, that is, the elements that are in one set or the other but not both. The last simplification is derived from the fact that:

$$\begin{aligned} 2d = |S_{:d}| + |T_{:d}| &= |S_{:d} \triangle T_{:d}| + 2 \cdot |S_{:d} \cap T_{:d}| \\ \Rightarrow 1 - \frac{|S_{:d} \cap T_{:d}|}{d} &= \frac{|S_{:d} \triangle T_{:d}|}{2d} \end{aligned} \tag{7.26}$$

So $\mathrm{RBD}(S, T)$ is the weighted sum of these $|S_{:d} \triangle T_{:d}|$, where the weighting is invariant on the contents of the list. Therefore, we need only demonstrate that

$$\forall d, |R_{:d} \triangle T_{:d}| \leq |R_{:d} \triangle S_{:d}| + |S_{:d} \triangle T_{:d}| \tag{7.27}$$

The remainder of the proof follows Fagin et al. (2003). Consider an element $x \in R \triangle T$. Assume, without loss of generality, that $x \in R$; therefore, $x \notin T$. There are two cases: $x \in S$, in which case $x \in S \triangle T$ but $x \notin R \triangle S$; or $x \notin S$, in which case $x \in R \triangle S$ but $x \notin S \triangle T$. Either way, if an element occurs on (contributes to) the left side of Equation 7.27, it must occur on (contribute to) the right side. Equation 7.27 then holds, and therefore so does Equation 7.24. □

Similar proofs hold for the metricity of $1 -$ RBO$_{\mathrm{MIN}}$ and $1 -$ RBO$_{\mathrm{EXT}}$.

7.3.6 Ties and uneven rankings

Ties may be handled by assuming that, if t items are tied for ranks d to $d + (t - 1)$, they all occur at rank d. To support this, we modify the definition of agreement given in Equation 7.3:

$$A_{S,T,d} = \frac{2 \cdot X_{S,T,d}}{|S_{:d}| + |T_{:d}|} \cdot \tag{7.28}$$

Equations 7.3 and 7.28 are equivalent in the absence of ties extending over rank d, but in the presence of such ties, the former formulation can lead to agreements greater than 1.

It occasionally happens that indefinite rankings are compared with different evaluation depths on each ranking. One cause of such irregularity is that the providers of the rankings are returning lists shorter than the evaluation depth chosen for the assessment and different from each other. We will call such lists *uneven rankings*. For instance, for an obscure query, one public search engine might return five results, another might return seven. These can still be treated as indefinite rankings; there are many more web pages beyond these depths, but they have not met the engine's threshold of estimated relevance. For the following discussion, let L be the longer of the two lists, with length l, and S be the shorter, with length s.

The formula for $\mathrm{RBO}_{\mathrm{MIN}}$ given in Equation 7.11 handles uneven rankings without modification, since it is implicitly assumed that $\forall d \in \{s+1, \ldots, l\}, S_d \notin L$; that is, we assume maximal disjointness and are done with it. Conversely, $\mathrm{RBO}_{\mathrm{MAX}}$ is found by assuming that every item in the extension of S matches one item in L, increasing the overlap by one. Therefore, $\forall d \in \{s+1, \ldots, l\}, X_d^{\max} - X_d^{\min} = d - s$, regardless of the contents of the preceding lists. The definition of $\mathrm{RBO}_{\mathrm{RES}}$ on uneven lists then becomes:

$$\mathrm{RBO}_{\mathrm{RES}}(L, S, l, s) = \\ \frac{1-p}{p}\left(\sum_{d=s+1}^{l}\frac{d-s}{d}p^d + \sum_{d=l+1}^{f}\frac{2d-l-s}{d}p^d + \sum_{d=f+1}^{\infty}\left(1-\frac{X_l}{d}\right)p^d\right) \quad (7.29)$$

where $f = l + s - X_l$ is the rank at which maximum agreement becomes 1. Removing the infinite sum using Equation 7.10 once again, and simplifying, results in:

$$\mathrm{RBO}_{\mathrm{RES}}(L, S, l, s) = p^s + p^l - p^f \\ - \frac{1-p}{p}\left(s\sum_{d=s+1}^{f}\frac{p^d}{d} + l\sum_{d=l+1}^{f}\frac{p^d}{d} + X_l\left[\ln\frac{1}{1-p} - \sum_{d=1}^{f}\frac{p^d}{d}\right]\right) \quad (7.30)$$

Modifying $\mathrm{RBO}_{\mathrm{EXT}}$ to handle uneven rankings is less straightforward. The extrapolation for even rankings is achieved by assuming the agreement in the unseen part of the lists is the same as in the prefixes. However, agreement between L and S is not known to depth l. And while agreement to depth s is known, truncation at this depth loses information on the degree of overlap between $L_{(s+1):l}$ and S. Therefore, extrapolation for uneven rankings must separately extrapolate agreement for $S_{(s+1):l}$.

Consider the method of extrapolation for even lists. The agreement A_k at common evaluation depth k is assumed to continue unchanged at later evaluation depths. In other words, $\forall d > k$, $A_d = A_k$, and specifically $A_{k+1} = A_k$. Referring to the definition of agreement in Equation 7.3, this means that

$$|S_{:k+1} \cap T_{:k+1}| \stackrel{\mathrm{def}}{=} X_{k+1} = X_k + A_k \; . \quad (7.31)$$

If $0 < A_k < 1$, which is generally the case, then working backwards through the formula implicitly requires $X_{d>k}$ to take on fractional values. This suggests the concept of degree of set membership. An element occurring in the seen prefix will have a membership degree of 1 or 0, depending on whether it is matched in the other list at the current evaluation depth. An unseen element, however, is assigned under extrapolation a (usually fractional) membership degree; one could think of it as a "probability of membership". The elements S_{k+1} and T_{k+1} in Equation 7.31, for even lists, each have membership A_k. In the case of uneven lists, the conjointness of $L_{(s+1):l}$ is known to

be either 0 or 1. Nevertheless, the membership of the unseen elements $S_{(s+1):l}$ can still be set to A_s. This will provide an assumed A_l, which can be extrapolated for elements beyond depth l, unseen in both lists. The formula then is:

$$\begin{aligned} \mathrm{RBO}_{\mathrm{EXT}}(L,S,l,s) &= \left(\frac{X_l - X_s}{l} + \frac{X_s}{s}\right) p^l \\ &\quad + \frac{1-p}{p}\left(\sum_{d=1}^{l} \frac{X_d}{d} p^d + \sum_{d=s+1}^{l} \frac{X_s(d-s)}{sd} p^d\right) \end{aligned} \quad (7.32)$$

Note that X_l here means the overlap on the seen lists at depth l, even though $|S| < l$; the maximum value of X_l is therefore s.

Calculating $\mathrm{RBO}_{\mathrm{EXT}}$ on uneven lists in this way maintains two important criteria met by extrapolation on even lists. First, $\mathrm{RBO}_{\mathrm{MIN}} \le \mathrm{RBO}_{\mathrm{EXT}} \le \mathrm{RBO}_{\mathrm{MAX}}$. And second, $\mathrm{RBO}_{\mathrm{EXT}}$ is non-increasing with deeper evaluation if S_{s+1} or L_{l+1} is found to be disjoint, and non-decreasing if the element is found to be conjoint.

7.4 Experimental demonstrations

Section 7.3 has defined the RBO metric, and described how it meets the criteria for an indefinite rank similarity measure, which the measures discussed in Section 7.2 failed to do. We now illustrate the use of RBO, first in comparing document rankings produced by public search engines, and secondly as an experimental tool in the research laboratory of the IR system developer. These domains involve non-conjoint rankings, so rank similarity measures such as τ that require conjointness (described in Section 3.3.6) cannot be applied. The only viable alternatives to RBO are other non-conjoint rank similarity measures. We provide comparisons with two of these: Kendall's distance (KD) and average overlap (AO).

7.4.1 Comparing search engines

We begin by using RBO to compare the results returned by public search engines. Twenty search engine users, drawn from the author's colleagues and acquaintances, were asked to provide search queries taken either from their recent search history or as examples of queries they might currently be searching for. Each user returned between three and eight queries, making a total of 113 queries, collected from mid-August to early September 2008. The queries were then submitted once a day to a number of public search engines, beginning on October 20th, 2008, and running up until February 26th, 2009. Eleven different search engines were searched, as listed in Table 7.1. Three of these are the Australian portals of international search engines. In every case, queries were submitted directly to the web site via URL manipulation and the HTML results list was scraped; the search APIs of these engines were not used. Except where noted, the top 100 results were retrieved from each search engine. Most search engines only provide at most two results from the one host, with the second result folded directly under the first, making the ranking between results from the one site non-determinable; therefore, in the experiments reported here, only the first result from any given host was retained. Result URLs were captured as returned by the search engines; no further normalization was performed.

Public search engines commonly return ten search results per page, including on the first results page. Therefore a reasonable choice of the p parameter is one that sets the

Name	URL	Notes
Google	`www.google.com`	Global Google. Google maps, news, blog results excluded; Google books results retained.
Yahoo	`search.yahoo.com`	Global Yahoo!.
Live	`search.live.com`	Global Live.
Ask	`www.ask.com`	Ask.
Dogpile	`www.dogpile.com`	Dogpile. Maximum 80 results.
Sensis	`www.sensis.com.au`	Sensis. Maximum 10 results. Not restricted to Australian-only results.
Alexa	`www.alexa.com`	Alexa.
A9	`a9.com`	A9. Maximum 20 results. Ceased offering general web search in January 2009. Prior to that, results based on Alexa.
Google (AU)	`www.google.com.au`	Google Australia search portal. Not restricted to Australian-only results.
Yahoo (AU)	`au.search.yahoo.com`	Yahoo! Australia search portal. Not restricted to Australian-only results.
Live (AU)	`www.live.com`	Specified `\?mkt=en-au`. Not restricted to Australian-only results.

Table 7.1: Public search engines used in experimental demonstrations.

expected number of results compared by the p-persistent user to 10. This is achieved by setting p to 0.9. As described in Section 7.3.3, this is equivalent to giving the first ten results 86% of the weight in the similarity comparison. It is also convenient to concentrate on the top ten results because, for interface reasons, it was not practical to retrieve more than the first ten results from some search engines. This again illustrates the importance of a rank similarity measure being monotonic in depth: we will be comparing rankings with a variety of depths, some going to depth 100, others to depth 10, and yet others somewhere in between, and we want the similarity scores produced to be comparable across all cases.

Table 7.2 gives the mean RBO_{EXT}, $p = 0.9$, between the different global search engines across all 113 queries on December 5th, 2008. The key to interpreting the numerical value of these scores is to remember that RBO is a measurement of expected overlap, or equivalently of a weighted mean of overlap at different depths. Thus, the RBO score of 0.25 between Google and Live can very roughly be understood as saying that the two systems have 25% of their results in common (as a decaying average over increasing depths). Contrary perhaps to expectations, different search engines are in fact returning quite different results, or at least result orderings, to queries; only a handful have an RBO above 0.25. By the date of this run of queries, Alexa had started to draw its results from Live, which is why the RBO score between them is so high. Previously, Alexa had been an independent search engine, which A9 drew its

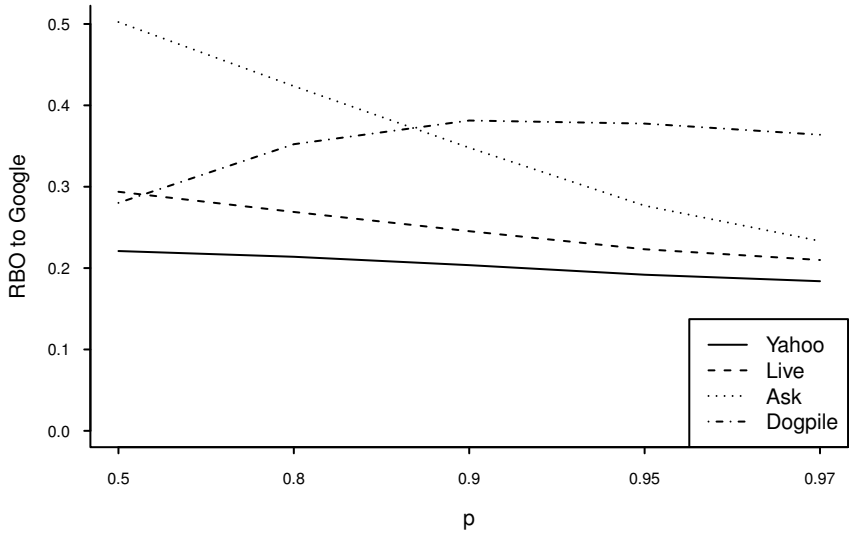


Figure 7.5: Mean RBO between Google and other search engines for different values of the p parameter. Raising p increases the depth of comparison.

search results from, and these two engines had a very high RBO (around 0.9). Not long after December 5th, 2008, A9 stopped offering general-purpose web search and became solely a product search aggregator. The Dogpile engine aggregates results from Google, Yahoo, Live, and Ask. The RBO figures suggest that Google results are given the strongest weighting by Dogpile; the fact that Ask is higher than Yahoo and Live may be because Ask is itself closer to Google. The Sensis search engine is quite unlike all the others, as to a lesser extent is A9. Table 7.3 shows the RBO between the global and Australian-localized search results for the search engines that provide localized variants. Apparently, Google performs much lighter localization than either Yahoo or Live.

Other values than 0.9 could reasonably be chosen for the p parameter in search engine comparisons. The researcher might wish to concentrate more heavily on the user experience of the first few results, in which case p values of 0.8 or even 0.5 might be appropriate, leading to expected comparison depths of 5 and 2, respectively. Con-

	yahoo	live	ask	dogpile	sensis	alexa	a9
google	0.20	0.25	0.35	0.38	0.03	0.23	0.11
yahoo		0.21	0.17	0.24	0.03	0.21	0.08
live			0.18	0.24	0.03	0.76	0.10
ask				0.27	0.04	0.17	0.09
dogpile					0.03	0.23	0.08
sensis						0.03	0.02
alexa							0.09

Table 7.2: Mean RBO, $p = 0.9$, between non-localized search engines across 113 user queries issued on 2008-12-05.

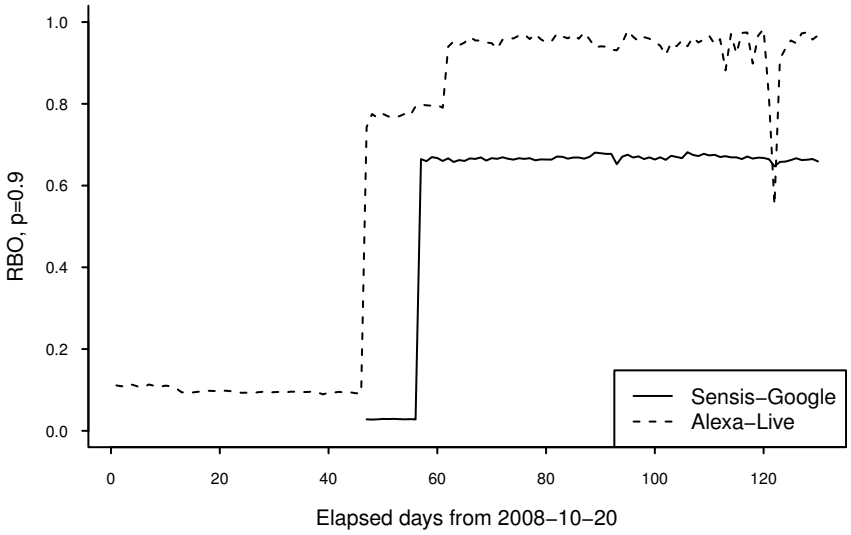


Figure 7.6: Mean RBO, $p = 0.9$, across 113 queries between Sensis and Google, and between Alexa and Live, calculated daily over the experimental period.

versely, a deeper, more system-centric comparison might be preferred, suggesting p values of 0.95 or 0.97 (expected depths of 20 and 33.3). Or the researcher might be interested to contrast a range of comparison depths. Because RBO's top-weightedness is tuneable via the p parameter, such investigations are possible. A question that can be addressed in this way is whether search engines are more similar to each other at the top of their rankings than further down. Raising the p value deepens the comparison, allowing us to explore this hypothesis. Figure 7.5 shows that Yahoo and Live are indeed more similar to Google at higher ranks than lower, but only mildly so. The difference is much stronger for Ask, suggesting that it is (by design or coincidence) strongly tuned towards delivering a similar first-page experience to Google. The rise, with increasing depth, of Dogpile's similarity to Google in Figure 7.5 might on a naive reading lead to the (surprising) interpretation that Dogpile draws more results from Google further down the ranking than higher up. But this interpretation fails to appreciate that aggregated results are supplementary, one engine drawing in another's answers; Dogpile's similarity to Live and Yahoo (not shown) rises even more strongly with depth. The function of RBO here is to alert us to an anomaly: Google's relationship to Dogpile is quite different from its relationship to the other engines.

During the period of the study, Sensis ceased being an independent search engine, and switched to deriving its results from Google. Similarly, Alexa changed to deriving its results from Live. These events can be traced by looking at the mean RBO scores

	google	yahoo	live
RBO	0.77	0.35	0.44

Table 7.3: Mean RBO, $p = 0.9$, between localized and non-localized search engines across 113 queries issued on 2008-12-05.

Search engine	Day-to-day			Start-to-end
	mean	sd	median	mean
google	0.91	0.08	0.94	0.50
yahoo	0.94	0.09	0.98	0.45
live	0.94	0.12	1.00	0.43
ask	0.94	0.13	1.00	0.41

Table 7.4: Rate of change of search engine results over time, as measured by RBO between sequential daily runs (left) and between start and end of experiment (right).

of the respective system pairs over time, as displayed in Figure 7.6. Evidently, Sensis switched to using Google on Day 57 (December 15th, 2008), while Alexa moved to using Live on Day 47 (December 5th, 2008), initially with some modifications, and almost verbatim from Day 62 (December 20th). The dip in similarity between Alexa and Live on Day 122 (February 18th, 2009) is due to Alexa giving idiosyncratic results on this day; why it does so is not clear. (Due to a problem with the query processor, complete results are not available for Sensis prior to Day 47.) Kendall's distance and average overlap detect similar overall trends to those shown in Figure 7.6, but show relatively greater similarity between Sensis and Google after the switch. We hypothesize that Sensis may be seeding (possibly localized) results into the top of the ranking provided by Google. The top-weightedness of RBO would detect such top-heavy seeding more effectively that Kendall's distance or average overlap.

Another question of interest is how much the results of different search engines change over time. This gives a sense of how dynamic a search service is, either in its crawling policy, or through changes in its ranking computation. For each of the 113 queries, the RBO between one day's results and the following day's results was calculated, for all 129 days in the experimental set. For each search engine, the mean and median across all day-to-day RBO scores were calculated, as was the mean of the standard deviation of RBO scores for each query over time. The results are shown in Table 7.4. Results tend to be relatively stable from one day to the next; indeed, for Live and Ask, the "typical" (median) result does not change at all. The results from Google show the highest rate of change. Additionally, changes to Google results, and to a lesser extent those of Yahoo, tend to be continuous and even (median closer to mean, low standard deviation), whereas changes to Live and Ask results are more sporadic (median further from mean, high standard deviation). Also shown is the mean RBO between result lists taken from towards the beginning of the experiment (Days 16 through 19) and then towards the end (Days 111 through 114), 16 pairs in total for each query and each system. Query results have shifted significantly over the three months, but systems are still more similar to the time-shifted versions of themselves than (referring back to Table 7.2) they are to each other. Interestingly, while Google shows more day-to-day change, it shows the least amount of long-term change. This latter result is significant in a two-sample, two-tailed t-test at 0.05 level between Google and each of the other search engines, but differences between the other engines are not significant.

It is informative to compare the RBO results with those obtained by using Kendall's distance at depth $k = 100$, reported in Table 7.5. The large degree of disjointness between results causes Kendall's distance to return negative values for all except the derivative Live–Alexa pair. Negative values make little sense in this application: there

is no sense in which any of these search engines are giving rankings negatively correlated with any other. Kendall's distance gives different relative results than RBO in a number of cases. For instance, RBO reports Dogpile to be closest to Google, but Kendall's distance places it closer to Yahoo; this is because on average Dogpile appears to pull more results from Yahoo than from Google (mean agreement at one hundred is 0.49 for Yahoo, and 0.30 for Google), but seems to give a higher ranking to the results from Google. Similarly, of the independent systems, RBO places Ask as being closest to Google, whereas Kendall's distance places it as being farthest away; again, in this case, Kendall's distance is following agreement at one hundred. Thus, although Kendall's distance is by design a correlation metric, its lack of top-weightedness and the highly non-conjoint nature of these indefinite rankings has it tending towards an unweighted measure of set agreement.

Too much significance should not be attached to these results as they stand. A rigorous examination of search engine similarities would start from these high-level RBO figures, not finish with them. Nevertheless, these comparisons do give a flavour of the analysis that a suitable rank similarity measure allows us to make upon search engine results, and indicate that RBO is uniquely suitable for these purposes.

7.4.2 Experimenting with information retrieval

In this section, we examine the use of RBO in a typical research situation, where an IR system is being modified, and the researcher wishes to measure how much the modification is changing the results. The researcher may be using the rank similarity measure as a proxy for a retrieval effectiveness metric. For instance, an efficiency change might have been made, and the rank similarity comparison is being used as an indicator of the degradation in effectiveness that the change may have caused, as with our first example below. Using RBO is attractive in this situation because performing the relevance assessments needed for effectiveness evaluation is expensive. If an initial examination with RBO determines that only slight changes have occurred in (top-weighted) ranking order for some or all topics, then the expense of relevance assessment on those topics can be avoided. Or the researcher may simply be measuring ranking fidelity as such, as in our second example.

Query pruning was mentioned in Section 7.1. It is a technique in which the amount of memory that is used in query processing is limited, and the amount of processing time reduced, but at a possible cost in retrieval accuracy and effectiveness. Therefore, if the results of a pruned system differ from those of an unpruned one, this sug-

	yahoo	live	ask	dogpile	sensis	alexa	a9
google	−0.60	−0.56	−0.66	−0.20	−0.93	−0.58	−0.80
yahoo		−0.55	−0.75	−0.04	−0.94	−0.56	−0.85
live			−0.73	−0.31	−0.93	0.62	−0.81
ask				−0.41	−0.91	−0.73	−0.83
dogpile					−0.93	−0.35	−0.82
sensis						−0.93	−0.95
alexa							−0.83

Table 7.5: Mean Kendall's distance at depth 100 between non-localized search engines across 113 user queries issued on 2008-12-05.

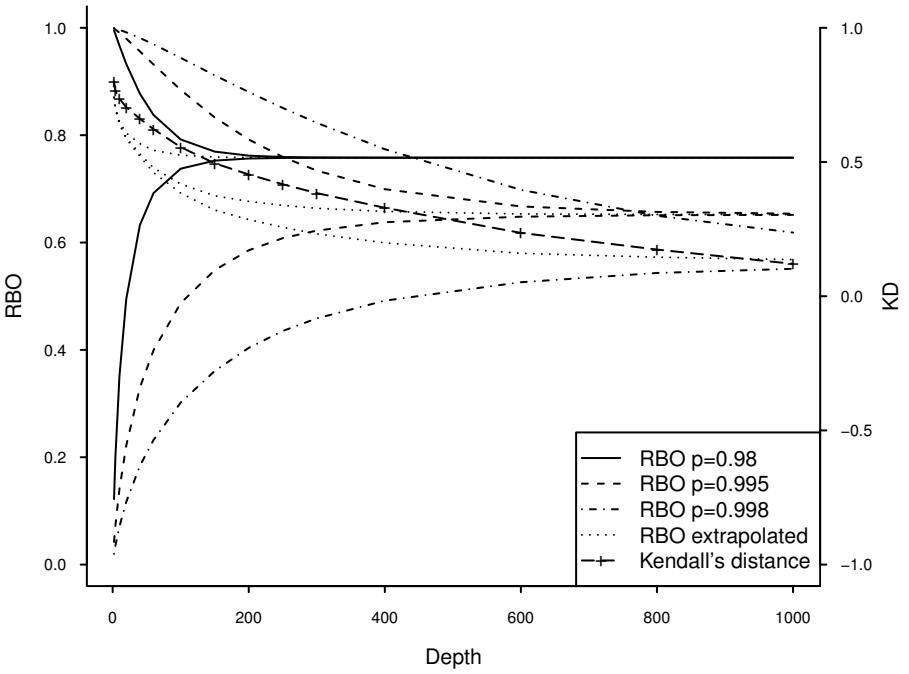


Figure 7.7: Similarity of query-pruned and unpruned runs. Kendall's distance and RBO with different p parameters are calculated at increasing depths, averaged across all topics. The upper and lower bounds and extrapolated values are shown for RBO. The corpus is wt18g, and the queries are TREC queries 551–600, title only. The retrieval engine is Zettair 0.9.3, using the Dirichlet similarity metric. Pruning is as described in Lester, Moffat, Webber, and Zobel (2005), with a limit of 1,000 accumulators, compared to no accumulator limit.

gests (though does not by itself prove) a degradation in effectiveness. Figure 7.7 gives the results of using RBO and Kendall's distance in a query pruning experiment. The query-pruned results are compared to the unpruned results, with evaluation carried out to varying depths. Here the unpruned results are the objective or "gold-standard" ranking, from which the pruned results deviate. All extrapolated RBO values and also Kendall's distance decrease as the depth of evaluation increases. This is because query pruning tends to have a greater effect on late-ranking than top-ranking documents. The extrapolated RBO value asymptotes to its final value relatively quickly, even for the very deep $p = 0.998$ evaluation. On the other hand, the Kendall's distance score is still falling at depth 1,000, and it is not clear what value it is asymptoting to, if any. We see here clearly that Kendall's distance is a measure, not on the full list, but on the prefix. In contrast, base plus residual RBO is a measure on the full list, and even the extrapolated value shows greater stability. It should be noted that all the p values chosen here are quite high. If one were using RBO as a proxy for a retrieval effectiveness metric, $p = 0.98$ would be at the upper end of the values one would be likely to choose, in which case the value has already converged by depth 200.

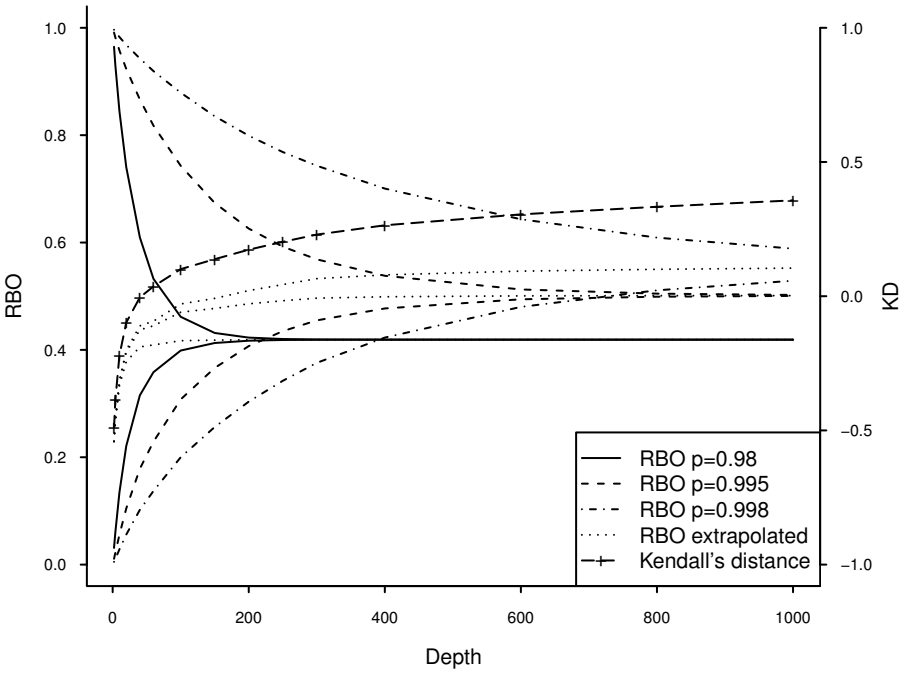


Figure 7.8: Similarity of runs with different similarity metric tuning parameters. Kendall's distance and RBO with different p parameters are calculated at increasing depths, averaged across all topics. The upper and lower bounds and extrapolated values are shown for RBO. The corpus is wt18g, and the queries are TREC queries 551–600, title only. The retrieval engine is Zettair 0.9.3, using the Dirichlet similarity metric. The μ parameter of the Dirichlet metric was set to 500 for one run, and 5,000 for the other.

Figure 7.8 shows a different kind of alteration to an information retrieval system. In this case, a language model smoothed with Dirichlet priors is being used to score the similarity between query and documents. This query–document similarity measure takes a parameter μ, which balances the influence of the relative weighting of terms within a document: with lower μ values, relative weighting is emphasized, meaning some terms have much higher impact than others, whereas with higher μ values, each term tends to have similar weighting and what matters is simply the presence or absence of the term (Zhai and Lafferty, 2004). Two different values of μ are being tried in Figure 7.8 as part of a parameter tuning experiment, with the mean RBO across a set of topics being displayed. Here, neither parameter value is the baseline or objective value, from which the other parameter is deviating and presumably degrading. Rather, the interest is in seeing how much difference is caused by altering the parameter. In contrast to Figure 7.7, the RBO_{EXT} and Kendall's distance scores trend up as depth of evaluation increases, not down. The reason is that parameter tuning tends to cause localized perturbations in ordering; as the depth increases, the degree of overlap increases too. All point measures give rising similarity values with depth, but Kendall's

distance rises considerably more than even the highest-p RBO, and it appears not to have asymptoted by depth $1{,}000$, even though the extrapolated RBO values stabilize well before that. Although Kendall's distance is derived from a metric that is based upon counting perturbations, it seems to be even more strongly affected by overlap than RBO itself is.

Of course, the preceding two cases are only examples. Different ranking perturbations will result in different effects on rank similarity measures. Nevertheless, these examples serve to illustrate two important points. The first is that the values of non-convergent measures evaluated to shallow depths can be very different from those at deeper depths, and so such measures cannot be regarded as adequate similarity measures on indefinite rankings. In contrast, a convergent metric gives hard bounds on infinite evaluation. The second, related point is that Kendall's distance and other top-k metrics cannot be regarded as single measures, but rather as families of measures, with each value of k (that is, depth of evaluation) instantiating a different member of the family. Kendall's distance is at least as dependent on its cutoff depth k as RBO is on its parameter p.

7.4.3 Correlation with effectiveness measures

We conclude by examining the relationship between rank similarity measures and changes in retrieval effectiveness. The metric used to calculate retrieval effectiveness is average precision (AP), which was defined in Section 3.2.2. To calculate the correlation between effectiveness and rank similarity measures, one could take actual retrieval runs, perturb their rankings, and calculate the similarity between the original and perturbed rankings on the one hand, and the change in effectiveness on the other. Actual rankings, however, are typically far from ideal ones, so randomly perturbing them, while decreasing the ranking similarity, has a rather noisy influence on effectiveness. Instead, we take a simulated approach. An ideal ranking of 10 relevant and 90 irrelevant documents is progressively degraded. The degradation consists of a sequence of 25 swaps between a relevant and a non-relevant document, chosen at random. After each such swap, the AP of the degraded ranking, and similarity of the degraded to the ideal ranking, is calculated and plotted. For calculating AP, the total number of relevant documents is set to 10 (that is, all relevant documents are contained in the depth 100 ranking).

The results of this simulated experiment are given in Figure 7.9. A total of 100 degradations were performed; each of the above figures therefore consists of $2{,}500$ points. The Kendall's τ between the AP score and the similarity value of the data points is also displayed. Kendall's distance shows a weaker correlation with AP than either of the top-weighted metrics. Moreover, it is more sensitive to the cutoff point. Cutoff at 10 gives the best correlation with AP across the whole sequence, but poor correlation at the top, and insensitivity to relationships beyond depth 10. Evaluation to depth 100 shows quite poor correlation. Average overlap shares some of this sensitivity to evaluation depth, whereas RBO has high fidelity at high similarity, regardless of the p value chosen. A comparison between the average overlap and RBO figures illustrates how intimately average overlap is linked with the choice of cutoff depth. Cutoff depth has at least as strong an effect on average overlap as changes in the p parameter has on RBO, even though as argued before cutoff depth is essentially arbitrary in an indefinite ranking.

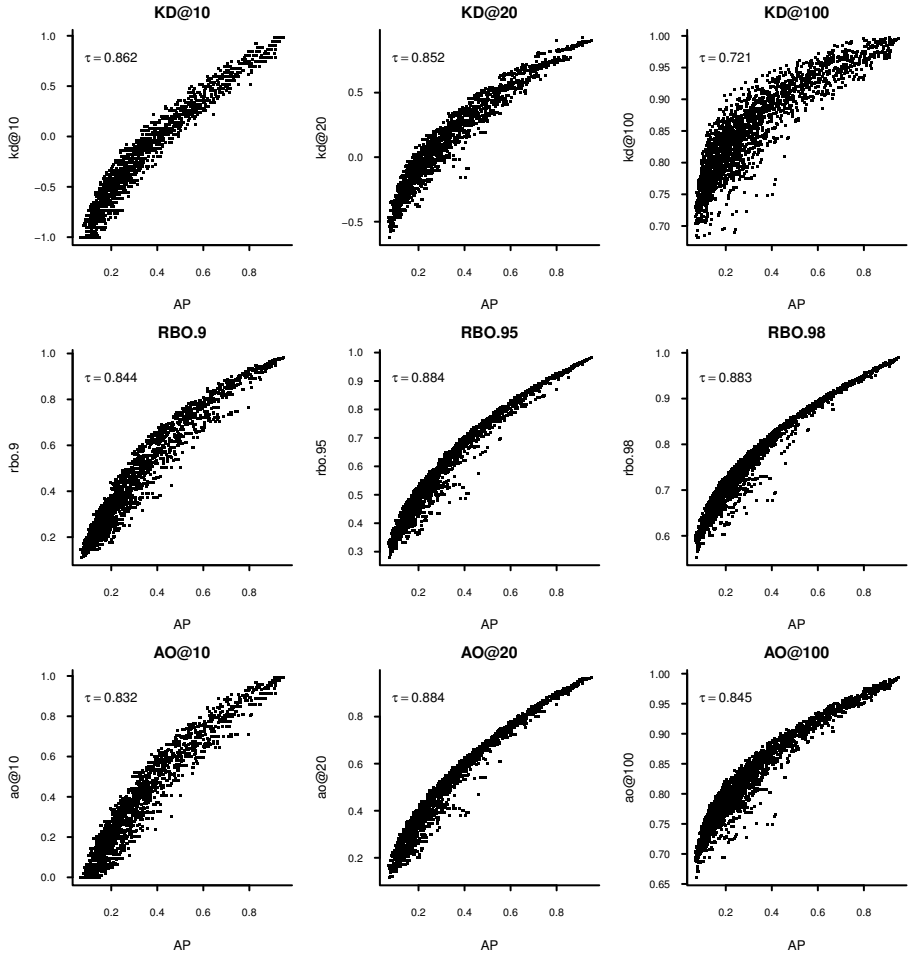


Figure 7.9: Correlation between the average precision (AP) of a degraded ranking on the one hand, and the rank similarity between the degraded and the ideal ranking on the other, for the experiment in which we start with a ranking of 10 relevant followed by 90 non-relevant documents, then randomly swap relevant and non-relevant elements 25 times, recording similarity and AP at each iteration, with 100 independent repetitions. The similarity metrics used are Kendall's distance (KD) at different depths; rank-biased overlap (RBO) with different p values; and average overlap (AO) at different depths.

7.5 Summary

Non-conjoint, top-weighted, and incomplete ranked lists—what we have called *indefinite* rankings—are encountered in the document rankings returned by retrieval systems, and in many other domains as well. An appropriate measure of similarity between indefinite rankings has, however, been lacking. Such a measure must recognize the peculiar characteristics of indefinite rankings. It must be top-weighted, giving more emphasis to the degree of similarity at the top of the ranking than further down. It must handle non-conjointness in the rankings, neither requiring every item to appear in both rankings, nor making arbitrary assumptions about where items uniquely seen in one

ranking are located (past prefix depth) in the other. And finally, it must recognize that the observed rankings are incomplete prefixes of much longer full rankings, and that the cutoff depth of the prefix is essentially arbitrary. A corollary of this incompleteness is that what is desired is a measure of the similarity of the full rankings, not merely of the observed prefixes. No existing similarity measure on ranked lists meets all of the above requirements.

In this chapter, we have introduced a new similarity measure on ranked lists, namely rank-biased overlap, or RBO. It is tuneably top-weighted, handles non-conjointness in the rankings, and is not tied to a particular prefix length. Most importantly, it is a similarity measure on the full rankings, even when only a prefix of each is available for comparison. It achieves this by using a convergent set of weights across successive prefixes, preventing the weight of the unseen tail from dominating that of the observed head. As a result, partial evaluation allows us to set strict upper and lower bounds on the similarity of the full rankings—a similarity whose exact value could only be calculated by evaluating the rankings in full. The RBO measure is parameterized to tune the degree of top-weightedness, and we have provided guidelines on the parameter choice. An extrapolated RBO value has been derived to give a reasonable point estimate on this similarity. This extrapolated value is itself monotonic on agreement. If the degree of agreement increases with deeper evaluation, the extrapolated value will go up; if agreement decreases, the extrapolated value will go down. Naturally, the extrapolated value is bounded by the upper and lower bounds of the RBO range. We have also proved that the distance measure $1-\text{RBO}$ is a metric, and extended RBO in a consistent way to handle tied ranks and prefix rankings of different lengths. Finally, we have illustrated the use of RBO in comparing public search engines and in the IR researcher's laboratory, demonstrating that it gives stabler and more intuitive results than alternative measures.

Rank-biased overlap can properly be considered as a branch of a family of measures on indefinite rankings, which are overlap-based measures using a convergent set of weights over prefixes. We have argued that an overlap-based measure makes more sense for indefinite rankings than do measures derived from the notion of correlation. Indeed, our illustrative examples suggest that, in the presence of high and variable degrees of non-conjointness, correlation-based metrics tend in practice to degenerate into unweighted measures of set overlap.

Chapter 8

Conclusions

The test collection method of retrieval evaluation is half a century old, and the method's realization in TREC as a community activity has been underway for nearly twenty years. The experience of TREC has inspired researchers to tackle many questions in the analysis of retrieval evaluation. How reliable are evaluation results, and how can this reliability be increased? How many queries do we need? What is the effect of pooling's incompleteness, and how can we compensate for it? How deep should pooling be? What constitutes statistical significance for retrieval experiments? What can we learn about system performance without using human assessment? This thesis has contributed answers to these questions: score standardization to improve result clarity and comparability (Chapter 4); power analysis for the design of reliable and efficient experiments (Chapter 5); score adjustment for the correction of pooling bias (Chapter 6); and rank-biased overlap for the comparison of document rankings without human assessment (Chapter 7).

As important as these technical questions is the role that evaluation plays in research practice. Reusable collections enable comparable experiments, but the comparisons need to be made, recorded, and referred to. We need continually to ask whether evaluation methods are measuring the desired properties, as retrieval domains and applications change. Preciseness of measurement is of little value, and is even misleading, if the wrong thing is being measured.

This concluding chapter begins by listing the thesis's outcomes in advancing the state of the art in test collection evaluative practice (Section 8.1). The chapter then continues to examine the use and influence of the test collection methodology. Section 8.2 poses a basic, but rarely asked, question: is retrieval effectiveness, as measured against test collections, improving over time? Then, in Section 8.3, we examine the challenges facing collection-based evaluation, and ask whether the method's strong influence has been entirely beneficial.

8.1 Thesis outcomes

The thesis began by proposing the use of *score standardization* to improve result clarity and comparability (Chapter 4). Topics typically vary in difficulty more than retrieval systems do in effectiveness, leading to instability in mean scores, difficulty in interpreting these scores, and obstacles to comparing scores between collections. The existing approach to varying topic difficulty is to normalize per-topic scores by the

theoretical maximum score that could be achieved for the topic, given the number of relevant documents. But the theoretical maximum score is a poor indicator of topic difficulty. Instead, we proposed that topic difficulty should be directly observed on the scores achieved by a set of reference systems run against the topic and collection. The mean and standard deviation of the per-topic reference system scores are then used to *standardize* system scores achieved on that topic. Standardization removes inter-topic variability for the reference systems, and greatly reduces it for new systems, too. Standardized system mean and per-topic scores are interpretable in isolation, as expressions of the system's performance relative to the reference baseline. Moreover, if the one reference set is run across multiple collections, it becomes possible to compare scores achieved on different collections. Standardization-enabled inter-collection comparison can even be performed retrospectively, as is illustrated in Section 8.2.1 below. Score standardization is the first contribution of the thesis.

The importance of applying tests of statistical significance to retrieval experiment results has been recognized in the retrieval community for at least a decade, but no work has been done on design-phase statistical analysis. Chapter 5 of this thesis introduces the use of *power analysis* for the design of reliable and efficient experiments. The power of a significance test is the probability that it will detect a specified true difference in effectiveness between systems, for a certain number of topics. Key to design-phase power analysis is an estimation of the likely standard deviation of experimental subject values; in retrieval evaluation, of per-topic, between-system score deltas. We have demonstrated that there is no single, typical (say) AP delta standard deviation, and that the value needs to be estimated afresh for each system pair. The most efficient approach is that of incrementally adding new topics until the desired power is achieved. We have shown, however, that the incremental approach leads to a slight bias in results, and we have provided an empirical quantification of this bias. Finally, power analysis provides a handy tool for assessing the degree of technological improvement that an experimental setup will be able reliably to detect. Using power analysis as an analytical tool, we argue that the standard 50-query topic sets are too small to reliably detect incremental improvements in a mature technology such as information retrieval. Power analysis is the second contribution of the thesis.

As corpora and topic set sizes grow, the same expenditure of resources in pooling leads to increasingly incomplete test collections. Assuming unpooled documents to be irrelevant is biased against new systems; ignoring them is biased in new systems' favour. In Chapter 6, we propose the method of *score adjustment for the correction of pooling bias*. The idea is to perform a leave-one-out experiment on topics for which a system is fully pooled, and use the results to estimate the bias the system suffers for topics on which it is unpooled. Unpooled scores for the system are then adjusted to counteract this bias. The score adjustment method is simple, statistically unbiased, and metric agnostic, and can be applied post-hoc to existing assessments. It is in particular suited to a dynamic evaluation environment, to which new topics are continually being added, and in which retrieval algorithms are constantly being developed and tuned; the method allows existing topics and their qrels to be reused, without have to go back and reassess them. Score adjustment is the third contribution of the thesis.

The methodology of system comparison in retrieval experiments has overwhelming been focused on effectiveness evaluation. There are, though, many circumstances in which we wish to compare the output of systems without performing relevance assessments, either as a cheaper proxy to effectiveness evaluation, or because we are interesting in system similarity in its own right. The document rankings produced by retrieval systems, however, have a number of special features. The top of the ranking

is more important than the tail; the rankings are incomplete and therefore mutually disjoint; and the rankings are cut off at an essentially arbitrary depth. In Chapter 7, we identify these features as characterizing an *indefinite ranking*, and argue that there are no satisfactory indefinite rank similarity measures described in the literature. We therefore propose our own: *rank-biased overlap* (RBO). The RBO measure is based on set overlap, a more intuitive foundation for indefinite rankings than rank correlation methods. The measure has a simple user model underlying it, and an informative probabilistic interpretation. Through experiments with the result lists of commercial web search engines, we demonstrate that RBO is a suitable and flexible measure of similarity between document rankings. The proposal of RBO, as the first true measure on indefinite rankings, is the fourth contribution of the thesis.

These four contributions constitute an important technical advance to the test collection method of retrieval evaluation. As important as technical correctness in evaluation methodology, however, is the use that evaluation methodology is put to in practice. In the light of these technical contributions, the remainder of this chapter examines the current and likely future direction of retrieval evaluation.

8.2 Trends in retrieval effectiveness

Standard evaluation methods and data sets allow results to be compared across systems, groups, and time. In TREC, between-group comparability is achieved by having multiple teams participate in each year's tasks, with the same availability of training data and opportunity for preparation. For comparisons of technology at a point in time, the TREC process has worked well. The rapid adoption of many early advances, such as the BM25 similarity metric and document length normalization in similarity scoring, can be attributed to the superiority of these approaches as demonstrated in TREC tasks.

Contrasting systems at the one time and on the one task is, however, only a single form of comparison. We also need to assess changes in effectiveness over time. Surprisingly, though, such longitudinal assessments have been rare and incomplete. Here we investigate two related questions about the development of retrieval effectiveness: first, whether effectiveness has been measurably improving amongst TREC participants (Section 8.2.1); and second, whether published results on TREC collections have been improving over time (Section 8.2.2). The answer to both questions is a fairly clear no, raising questions about experimental and reviewing practice (Section 8.2.3).

8.2.1 Result trends at TREC

Comparing results within a given year of a TREC task is straightforward, but doing so between years is not. An immediate problem is that no track has continued throughout TREC, and different tracks vary considerably in their tasks. The longest-running track was the AdHoc Track, from TREC 1 through TREC 8, substantially continued as the Robust Track from TREC 2003 to TREC 2005. Attention in this section is restricted to these two tracks, which will be referred to as "the ad hoc tracks". The analysis omits TREC 1, for which runs are not available, and TREC 2, for which many are malformed.

Even within the one track, comparing results between years is tricky. The high variability in topic difficulty means that, even if each year's task was the same, collection difficulty would vary. But in most tracks, the task is not precisely the same each year. In the AdHoc Track, for instance, the document collection varied between years, and topic formats changed repeatedly until TREC 6. Because of the varying

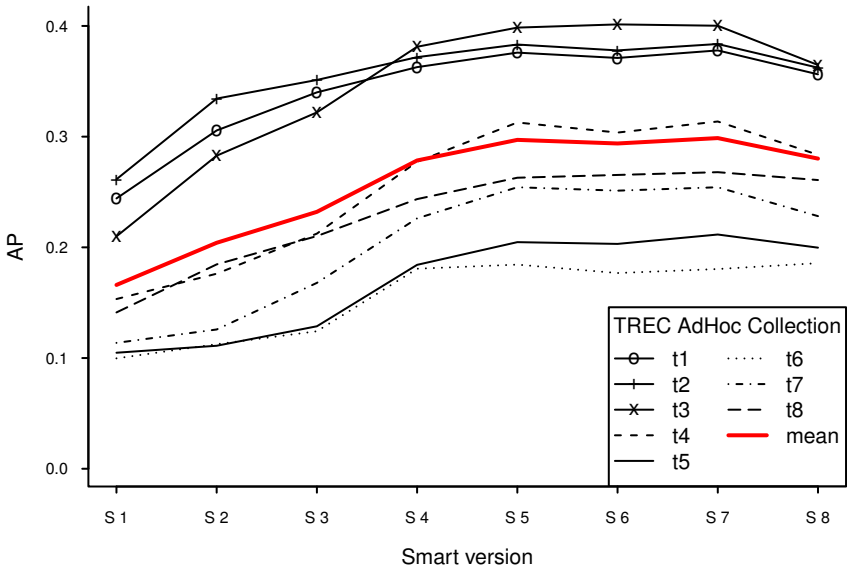


Figure 8.1: The performance of the first 8 TREC-participating versions of SMART against the first 8 TREC AdHoc test collections. The data is taken from Buckley (2005, page 311).

difficulty of the ad hoc tasks, scores cannot directly be compared between years. One solution would be to run each year's systems again on the following year's task, but this was not done systematically at TREC. The SMART team, however, did undertake their own analysis, evaluating SMART versions from the first eight TRECs against the first eight AdHoc collections. The outcome is shown in Figure 8.1, using data taken directly from Buckley (2005). Variance between collections exceeds variance between system versions. Taken as a whole, though, the figure shows a steady increase in effectiveness up to TREC 4 or TREC 5, and a plateau in performance from then on. These results were cited in justification of the suspension of the AdHoc Track following TREC 8 (Voorhees and Harman, 1999).

The complete crossing of systems with collections makes the SMART analysis persuasive, but it only demonstrates the development of that team's technology. It is possible that systems based on different principles would exhibit different behaviour. What of the other participants in TREC? Answering this question requires a method of controlling for collection difficulty. Fortunately, we have developed just such a method, namely score standardization (Chapter 4).

Score standardization requires a set of reference systems. To create this reference set, five publicly available retrieval systems (Lucene, MG, Zettair, Indri, and Terrier) were run against the ad hoc collections in a total of seventeen different configurations (Armstrong, Moffat, Webber, and Zobel, 2009c). Standardized scores were mapped to the normal CDF (Section 4.7.3), to bring them within the $[0, 1]$ range. Despite the different systems and varying configurations, the reference set provided a narrow distribution of middling scores, relative to the original systems; today's out-of-the-box retrieval technology still cannot match the best (albeit highly-tuned) original TREC runs. Two virtual reference systems were therefore added for smoothing: one defective system scoring 0 for every topic, and another, perfect system scoring 1.

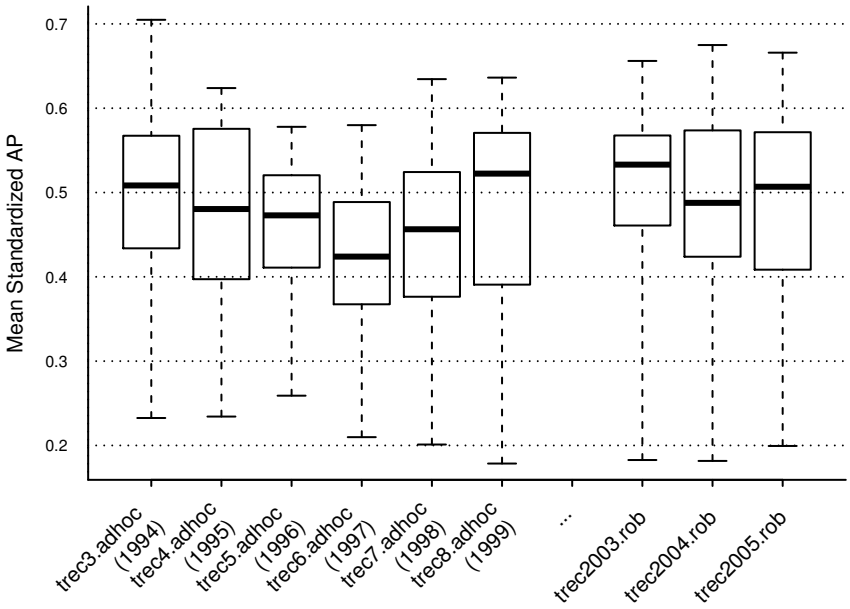


Figure 8.2: Standardized scores, under normal CDF mapping, of automatic TREC participant systems in the TREC 3 through TREC 8 AdHoc track, and the TREC 2003 through TREC 2005 Robust track. The thick horizontal line through each box shows the median of the system mean standardized AP scores; the edges of the box show the upper and lower quartiles; and the ends of the whiskers show the minimum and maximum scores. A standardized score of 0.5 equates to the mean performance of the nineteen reference systems, including the synthetic defective and perfect systems.

With the reference set constructed, standardized mean AP scores for all automatic AdHoc and Robust participant systems were calculated, and their distribution in each year compared. The result is shown in Figure 8.2. There is no sign of an upward trend from TREC 3 onwards. Indeed, performance seems to fall off slightly in the following few iterations, before recovering later. And the best system overall, across the entire nine iterations and twelve years examined, is the top-performing system from TREC 3 (the original Okapi BM25 system, as it happens). In short, there is no evidence in Figure 8.2 of a measurable increase in retrieval effectiveness amongst ad hoc TREC participants since at least TREC 3, in 1994.

There is some disagreement between Figure 8.2 and the SMART analysis in Figure 8.1, which showed continuing improvement until TREC 4 or 5. It may be that SMART took a year or two to catch up with advances in the state of the art. But while the two analyses differ slightly in dating the start of the performance plateau, they agree in finding no increase in effectiveness in the latter half of the AdHoc track, in the late 1990s; and the standardized results extend this finding to Robust track in the mid 2000s.

So there does not seem to have been an improvement in ad hoc retrieval effectiveness at TREC itself since the mid 1990s. But what about outside of TREC? We examine this question next, through published retrieval results.

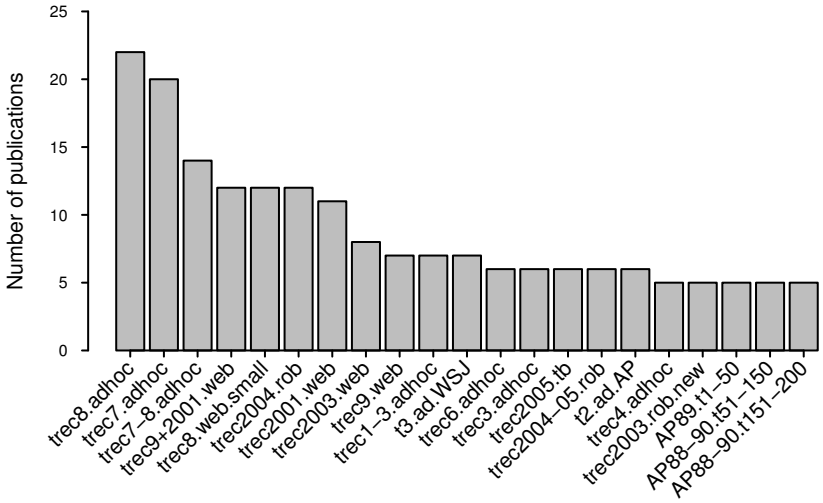


Figure 8.3: Frequency of use of TREC test collections and pseudo-collections in SIGIR and CIKM papers. Only collections used in 5 or more publications are shown; another 101 usages (38% of all usages) were for collections used in fewer than 5 publications.

8.2.2 Result trends in published research

The collections produced at each year's TREC have been extensively used in subsequent research. By collecting published results, longitudinal performance information can be gathered. An important difference between runs at TREC and subsequent experiments is that the latter had the official qrels available to them, which can (inadvertently) lead to a training effect. And of course later researchers know how well earlier approaches worked on each collection. These factors give post-TREC runs an advantage that is independent of genuine technological improvement; experimental scores reported on TREC collections may need to be discounted in response.

To examine performance trends, we surveyed published results on TREC ad hoc style collections (Armstrong, Moffat, Webber, and Zobel, 2009a). The surveyed venues were SIGIR from 1998 to 2008, and CIKM from 2004 to 2008. Both full papers and posters were included. All reported mean system AP scores on the TREC AdHoc, Robust, Web, and Terabyte collections, and subsets thereof, were recorded, excluding cases where training was directly performed on the test set, or where the reported results appeared incorrect (such as incompatible AP and P@10 scores). Some 22 publications were excluded on this basis, leaving a total of 106 papers, 85 from SIGIR and 21 from CIKM. A list of the surveyed papers is given in Armstrong et al. (2009a). For each combination of publication, collection, and query type, the strongest baseline and the best "improved" score was recorded. A total of 83 different collection variants were used in these 106 papers; the more frequently used ones are displayed in Figure 8.3. Of the 83 collections, 12 were used often enough, and over a long enough period, for some indication of trends to be discerned. Here, we look at two of the most frequently used collections; figures for all 12 are given in Armstrong et al. (2009a). (Note that standardization cannot be used to combine this data into a single analysis, as per-query scores, required for the standardization process, are missing.)

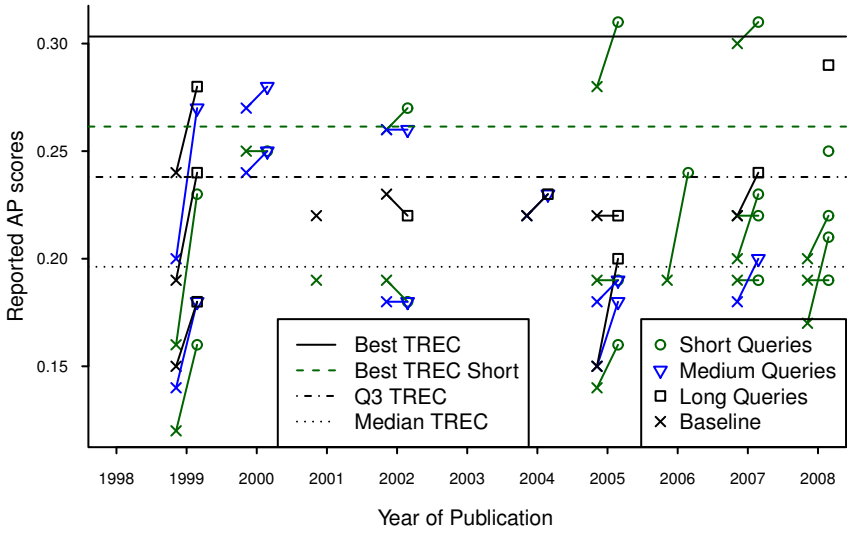


Figure 8.4: Mean system AP scores reported in SIGIR and CIKM papers on the TREC 7 AdHoc collection. Paired baseline and improved scores in each paper are joined by a line. Short queries are title-only; medium are title plus description; and long are all fields. The best original TREC automatic and automatic title-only scores are marked by horizontal lines, along with third-quartile and median automatic scores. Of the papers surveyed, 22 reported results on the TREC 7 AdHoc collection, totalling 36 baseline and 36 improved scores. (One paper could report results on more than one query type.)

The AP scores reported against the TREC 7 AdHoc collection are shown in Figure 8.4. There is no clear upwards trend in reported scores over time. Indeed, the mean of the system scores is lower from 2005 onwards than prior to 2005, both for improved systems (0.23 in the latter period, 0.22 in the former) and for baselines (0.21 to 0.20). In addition, published results are rarely competitive with the best original TREC systems. Only eight of the thirty-six baselines are at or above the third quartile of the TREC participant runs, and over half the baselines are below the median. Only two improved results, both from the same research group (Liu et al. (2005) and Zhang et al. (2007)), beat the best original TREC automatic run. The pattern is one of results that fall below the best TREC participant runs, with the same low baselines being improved upon by similar amounts each year, and no trend of improvement over time. Nevertheless, a statistically significant improvement is claimed in 35% of these publications. These figures are representative of results published on all AdHoc collections.

The pattern for the (more recent) Web and Robust collections is broadly similar, although there is a greater tendency for an upwards trend in scores over time, and also more examples of published results exceeding the best TREC systems, at least for comparable (for instance, short) query types. The results for the most frequently-used of the non-AdHoc collections, the TREC 8 Small Web collection, are given in Figure 8.5. The slightly better trends in published results for these collections may be due to the newness of the tasks, which leaves more room for improvement (and also makes the original TREC benchmarks less demanding). But still, the trend in scores is mixed and far from cumulative.

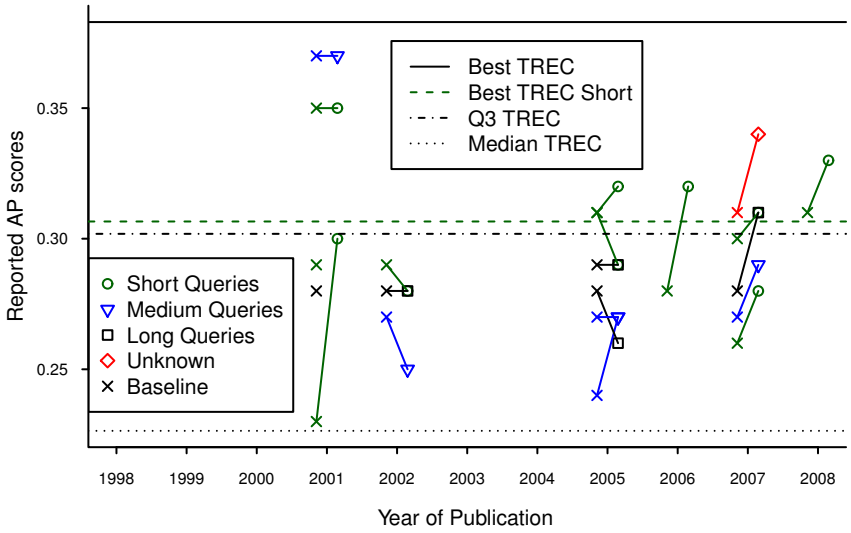


Figure 8.5: Mean system AP scores reported in SIGIR and CIKM papers against the TREC 8 Small Web collection. The "best TREC short" line is a lower bound, as not all query types are specified in the Track reports. Of the papers surveyed, 12 reported results on the TREC 8 Small Web collection, totalling 21 baseline and 19 improved scores.

The survey of published retrieval results on TREC collections in this section renders a similar conclusion to the analysis of the standardized scores of the original TREC participants in Section 8.2.1. This conclusion is that there has been no clear, measurable improvement in ad hoc retrieval performance over the past decade or more; certainly not on the AdHoc newswire collections, and not convincingly so on web data. In the following section, we consider the ramifications of these findings for the field of information retrieval.

8.2.3 The practice of IR evaluation

Test collections enable comparable experiments, but comparisons over time have not been regularly made. The longitudinal studies described in the previous sections uncover some less than encouraging findings. Dozens of papers that claim improvements in retrieval effectiveness have been published in leading venues, adhering to the field's experimental standards, and frequently achieving statistical significance. And yet ad hoc retrieval effectiveness has not measurably improved over the past decade and a half—with little explicit public recognition of the fact.

Much of the benefit of test collection evaluation is lost if results are not recorded over time. One should not have to spend a fortnight skimming conference proceedings to determine the current state of the art, not least if one is a time-pressed paper reviewer. We have created EvaluatIR, an online repository for retrieval results, as an example of what best practice in this area should be (Armstrong, Moffat, Webber, and Zobel, 2009b).[1]

[1] `http://evaluatir.org/`

The availability of retrieval scores over time not only informs practitioners of the general trend in performance. It also forces individual researchers to place their effectiveness results in a broader context. There may be good reasons why this year's method does not take last year's best result as its baseline; but the previous high water mark should at least be stated, and the reason for not starting at it justified. Otherwise, researchers are left repeating the same effectiveness increments, without anyone being properly aware of it.

There is, though, a more worrying interpretation of the combination of weak baselines and statistically significant improvements found in published results. The importance of testing for statistical significance has been established for over a decade (Zobel, 1998), and of comparing against a baseline for longer still. The review and publication process means that we only see the work that has cleared this hurdle; and choosing weak baselines makes it an easier hurdle to clear. Researchers who choose more competitive baselines will have more difficulty in achieving significant improvements, or any improvements at all. Their work may be more likely to be rejected by reviewers, or self-censored by the researchers. Thus, it may be that experimental practice is not as undemanding as the published results suggest. Rather, the formal, but incomplete, requirement for experimental rigour may, perversely, be filtering the more rigorous work out, and favouring the less demanding attempts. This interpretation of the published record is another strong argument for making a collation of TREC and other published results readily available for the use of reviewers.

Perversion of the publication review process aside, we are left with the question of why there is no measurable upwards trend in ad hoc retrieval effectiveness, when mean AP scores remain stuck around 0.35, a third of their nominal potential. It could be that the field awaits a technological breakthrough in, say, deep natural-language processing. It could also be that the problem lies in the task itself. Inter-assessor agreement is often low (Voorhees, 1998), and title-only queries are frequently ambiguous and underspecified. Perhaps retrieval is operating closer to its real effectiveness limit than scores imply. Or, finally, it could be that real progress is achievable, but our test collections lack the experimental power to detect it, as Chapter 5 suggests.

Our survey of published and TREC results has raised questions about the use and interpretation of test collection evaluation. The formalisms of the methodology are being observed, but retrieval effectiveness is not measurably improving, and publication practice is as much obscuring as revealing this lack of progress. Meanwhile, the nature of information access is changing rapidly, on the web and elsewhere. Researchers must keep abreast, and ideally ahead, of this change; and this places pressure on evaluation technology. The next section discusses the challenges and opportunities facing the field of information retrieval as it responds to these pressures.

8.3 Challenges and opportunities for IR evaluation

Evaluation practice is well-developed within information retrieval, placing the field in advance of many other areas of computer science. But many challenges remain, and new ones present themselves. This section is an examination of some of these challenges. We begin in Section 8.3.1 with the problems of scale, data access, and coherence facing the test collection method. Section 8.3.2 then considers what IR has to teach, and learn from, other domains and modes of evaluation. Finally, in Section 8.3.3, we examine the place of evaluation in the research economy, and ask whether methodological strength is impeding the field's flexibility and innovation.

8.3.1 Extending test collection evaluation

The test collection model of evaluation, which is the focus of this thesis, has been central to the achievements of the field of information retrieval over the past decades, and will remain important to its progress in the future. As a panel of prominent researchers noted recently, "robust and well-accepted experimental methodologies are significant research accomplishments that do not happen often" (Callan et al., 2007). Such a methodology, once built, is a valuable possession, worth much effort to renovate and extend. This thesis is part of that effort. Many challenges face the test collection methodology at present. Among these is reflecting the diversity and ambiguity of keyword web search; handling the scale of data that the web provides; and accessing and employing the usage data that is key to analyzing behaviour on the web.

Diversity, ambiguity, and coherence

The standard test collection model treats the relevance of a document to a query as a discrete, independent, and absolute event: a document is either relevant to a topic, or it is not, and its relevance is not affected by what else appears in the ranking (Section 3.1.2). The reality of retrieval is more complex. Few users like to see the same information repeated; and many queries are ambiguous. A good retrieval system will make its results list diverse, to cover different aspects of complex information needs, and different senses of ambiguous queries; and a good retrieval evaluation method will reward such diversity.

The benefit of result diversity has been appreciated for many decades (Boyce, 1982), and the manual evaluation of topic aspects and sentence-level novelty has been undertaken previously in the TREC Interactive and Novelty tracks (Over, 1997; Harman, 2002). But it is only recently that a concerted effort has been made to develop test collections and metrics for the automated evaluation of diversity in search results (Clarke et al., 2008), culminating in 2009 in the Diversity Task of the Web Track (Clarke et al., 2009). The long delay in implementing even such a seemingly minor (though in practice complex) extension to test collection evaluation underlines how difficult and important the development of an experimental methodology is.

The pre-eminence of web search emphasizes the importance of result diversity, due both to the range of information sources and types on the web, and to the terse, ambiguous nature of web queries. Diversity also fits within a wider emphasis upon whole-of-page relevance: the organization and presentation of information in the results page to maximize user satisfaction and minimize effort and confusion. One component of the task is integrating information from multiple data sources, such as blog posts, news items, and targetted advertisements, alongside traditional search results (Callan et al., 2007). How should the effectiveness of such integration be evaluated? The technique of side-by-side evaluation has been developed for assessing users' whole-of-page preference (Thomas and Hawking, 2006), but unlike qrels in standard test collections, side-by-side assessments are not reusable. Another aspect of organization and presentation is in the summarization of search results; and while result snippets are a long-standing feature of web engines, work on incorporating snippets into automated test collection evaluation has only just begun (Turpin et al., 2009).

In whole-of-page relevance at least, practice is outstripping research. The major search engines are increasingly competing on the presentation and organization of their results, rather than simply their topical relevance. Moreover, new online search needs are rapidly emerging, such as real-time search and micro-blogging. The challenge

for the research community is to retain the relevance of their work to contemporary practice, without abandoning the methodological rigour that has developed around the established keyword and ranked list retrieval model. As we argue in Section 8.3.3, the tension between rigour and innovation is not resolvable as a rational optimization along a smooth continuum, by calmly trading off one for the other; there is a strong inertia of structure and convention.

Scale and dynamism

Retrieval evaluation is challenged not just by the web's diversity, but also by its scale. Gathering a web-scale document corpus itself is within the resources of the public research community, as shown by the billion-page ClueWeb collection, used in the TREC Web Track since 2009 (Clarke et al., 2009). The problem lies in creating a reusable qrel set for such a large collection—a problem increased by the need for larger topic sets to reflect the diversity of web search. Sampling and inference techniques have been developed to estimate, with reduced effort, the scores that participant systems would achieve under full pooling (Aslam et al., 2006; Carterette, 2007); but the reusability of qrel sets created in this way is unclear. The need to estimate such deep assessment is itself questionable: shallow but broad evaluation gives greater statistical power (Chapter 5), and is equally predictive of user satisfaction (Sanderson et al., 2010); but again, such shallow assessments, gathered once for a static collection, are not reusable.

In addition to scale is the problem of change, in documents and in queries. The web is highly dynamic, and search companies place increasing emphasis upon the freshness of their results.[2] A static web collection quickly becomes out of date. Staleness is of particular concern where static and live resources are integrated in experiments, as is increasingly done. For instance, a recent user study used the ClueWeb collection for document retrieval, and fetched the corresponding snippets from a live web search engine; but 35% of the results retrieved from ClueWeb were no longer in the search engine's index (Sanderson et al., 2010).

The issues both of scale and of change suggest the need to design a protocol for extensible collections, rather than the static collections currently employed. In an extensible collection, assessments would be performed only shallowly, and on results actually returned. Partial incompleteness in assessments would be dealt with initially by inferential methods such as score adjustment (Chapter 6); once incompleteness had become serious enough, more assessments would be performed. Crowdsourcing (discussed in Section 8.3.2) offers a possible, independent resource for such assessments. Extensible collections would also allow for the addition of new documents to the corpus and queries to the topic set. There are many problems with such an approach, including the comparability of retrieval scores achieved on different versions of the collection; and extensible collections would not meet all research needs. Nevertheless, they seem an attractive option for research work on web-scale, and web-dynamic, data.

The availability of user data

The user is represented in traditional test collections through topics and relevance assessments. But much of the most interesting data for research into search behaviour is found elsewhere, in search logs, click-through records, and the information captured by web browser toolbars. These sources provide a wealth of user data: when and in

[2] `http://googleblog.blogspot.com/2010/06/our-new-search-index-caffeine.html` (published 8th June, 2010; last retrieved 20th July, 2010).

what context users decide to search; how they reformulate their queries; how queries relate to each other; when searches are successful and unsuccessful; and so on. These are some of the most pressing questions for current retrieval research.

Such usage data is collected by search engines, but is not readily available to public researchers. Privacy concerns are part of the problem, as the debacle of the AOL query log release illustrates (in 2008, AOL publicly released a query log that had been superficially anonymized; but sessions were readily linked back to users through personally identifying queries[3]). As importantly, usage data is proprietary information of considerable business value; search companies are willing to release such data only incompletely and with delay, if they are willing to release it at all. An effort to collect equivalent information for public research, through the Lemur Toolbar,[4] has been abandoned due to lack of participation.

The asymmetry between public research groups and commercial search labs in their access to usage data provides the latter with a clear research advantage. Public researchers (and company researchers doing fully public research) seem destined to rely upon commercial search providers for (frequently unsatisfactory) data sets to work with. The danger is that the data paucity will feed a perception, if not a reality, of growing irrelevance of public research to actual search engine practice. This is on the face of it an undesirable state of affairs. But it might have the desirable side-effect of forcing upon public researchers a renewed innovation in the problems they tackle, the methods they use, and the solutions they propose.

8.3.2 Beyond Cranfield and outside IR

The previous section examined the challenges facing the relevance-centric, test collection model of retrieval evaluation. There are, though, others possibilities for retrieval evaluation besides the test collection, and other fields besides information retrieval in which the lessons of Cranfield and TREC can be applied. We investigate some of these possibilities here.

Crowdsourcing

Test collection evaluation is founded on the assumption that human assessment is expensive and time-consuming. Evaluation must be designed to re-use this precious resource, even if that constrains the evaluation model's realism and flexibility. But a recent development has challenged these economic assumptions. The development is that of *crowdsourcing*: the online distribution of human intelligence tasks (HITs), via a service company such as Mechanical Turk,[5] to casual piece-workers. Crowdsourcing creates efficiencies of scale and automation in human studies. More significantly, though, crowdsourced workers are willing to work cheaply, more cheaply even than nominally remunerated in-laboratory subjects. A recent IR study reports paying an effective rate of 80 cents an hour. Several thousand rank-pair comparisons were performed by three hundred different workers for a total experimental budget of just sixty dollars (Sanderson et al., 2010). Whether quite such a low level of payment is sustainably consistent with reliable human effort may be questioned; nevertheless, the convenience (or perhaps multi-nationality) of crowdsourced work does seem to substantially reduce the required remuneration.

[3] `http://sifaka.cs.uiuc.edu/xshen/aol_querylog.html` (last retrieved 25th July, 2010.)

[4] `http://lemurstudy.cs.umass.edu/` (last retrieved 20th July, 2010).

[5] `http://www.mturk.com/`

Crowdsourcing could simply be used to produce static test collections at reduced expense. But the deeper change that it offers to retrieval evaluation is to make large-scale live experiments, involving non-reusable HITs, affordable to projects on even modest budgets. The growing interest in whole-of-page relevance, for instance, has been mentioned previously. In the past, the chief obstacle has been how to re-use discrete relevance assessments to perform this holistic evaluation. With cheap HITs, though, reusing existing assessments can be foregone, and subjects can be asked to compare results pages in their entirety—as indeed was done in the study by Sanderson et al. (2010) cited above. Moreover, the crowdsourced setup provides human cognition on tap within an almost-automated task. Crowd-sourced workers have even been integrated as a text revision module within an interactive word processor (Bernstein et al., 2010). The experimental environment, data, code, and even subject recruitment process could be packaged up and made available on call, making human-cognition experiments automated and reproducible. The proposal in Section 8.3.1 for extensible collections driven by crowdsourced evaluation is an example of the software–wetware experimental hybrid that is possible.

As attractive as possibilities of crowdsourcing appear, there are potential obstacles. A basic one is ethics approval. Crowdsourcers work voluntarily; but is it ethical to exploit this resource with so little remuneration? A second problem is assuring the reliability of the data collected. Outright spam can be filtered by setting *trap tasks*, ones to which the answer is known and which no attentive human would get wrong. Spam aside, human subjects have varying levels of diligence, and assessing this in the automated, faceless crowdsourced environment poses different challenges to doing so in a lab. As cheap as crowdsourced evaluation is, it is hard to believe that it is done to a high quality. In particular, evaluation tasks such as relevance assessment require the subject to first imagine themselves performing a task in a real-life context, such as searching to address an information need, and then to assess their satisfaction with the results, given their imagined context. Such contextualized, subjective tasks are cognitively more complex than the stereotypical HIT of, say, identifying whether a picture is of a bird or a plane. How realistically they are performed by a throughput-focused, online piece-worker may be questioned, and requires empirical investigation.

Beyond IR

Information retrieval was one of the first computing fields to employ human evaluation, and it has a strong tradition of empirical work, built on well-developed methodologies, particularly in automated, collection-based evaluation. Experience in retrieval can contribute to the development of evaluation in other computing fields, as more and more of them become enmeshed in human and social interaction.

An example of the scope for improved evaluation practice comes from the field of keyword retrieval on structured data and databases (Webber, 2010). Database retrieval has traditionally been performed using formal query languages, such as SQL, which precisely determine which records match the query. But the spread of free-text querying through web search has provoked interest in applying keyword queries to database retrieval. The initial practicalities of keyword search having been resolved (Agrawal et al., 2002), attention has turned to achieving and measuring effective retrieval from these informal queries.

To date, however, evaluation practice in keyword retrieval from databases has fallen well below the standards of information retrieval. While there are de facto some common corpora, each research group develops their own queries and performs their own

assessments, hindering independent comparisons. Query sets are small, as few as 5 queries for each corpus and rarely more than 20, and unstable metrics such as MRR and P@1 are favoured (Li et al., 2008; Luo et al., 2007), leaving experimental results inconclusive. The home-grown queries are frequently slanted towards highlighting particular features of a research group's approach. For instance, a retrieval method that matches queries against schema terms will be tested with queries containing schema keywords (Liu et al., 2006). Perhaps in consequence, each new method reports near-perfect scores, and doubles or triples those of their re-implemented baselines, which themselves had originally achieved flawless results (Liu et al., 2006; Luo et al., 2007; Xu et al., 2009).

The prescriptions that the evaluation experience of information retrieval would draw up for this situation are fairly clear: standard collections, with independently determined information needs and relevance assessments; larger topic sets; and deeper, more stable metrics (Webber, 2010). The need for more thorough, objective evaluation along these lines has been recognized by the database community (Coffman and Weaver, 2010). To bridge the gap between the communities, the 2010 INEX evaluation forum has added a track for data-centric XML to its traditional focus on semi-structured XML documents.[6] But even in so similar a field as keyword retrieval on databases, it is not simply a matter of taking the IR methodology and copying it wholesale, since keyword retrieval on databases has many distinctive features. A satisfactory evaluation model must, for instance, take account of the structure inherent in the data, and the possibilities this offers for query formulation, query processing, and results presentation. For fields more distant from information retrieval than keyword retrieval on databases, the required customization of method will be still greater. Nevertheless, the retrieval evaluation experience offers valuable lessons for all fields of computer science on the power (and pitfalls) of a standardized, repeatable, automated evaluation methodology.

8.3.3 Evaluation in the research economy

This thesis began with an historical view of retrieval evaluation's past; it is appropriate that it conclude with a sociological interpretation of its present. Evaluation plays a crucial rule in the verification of scientific progress; but it also plays a central role in the conventions of scientific practice, and in the exactions and disbursements of the research economy. A rigorous evaluation methodology provides the verification and measurement of improvements; but the existence of a strong, established methodology can impede the flexibility, progress, and ultimately relevance of a research field.

The sociology of methodology

The applied sciences are directed towards scientifically validated technological advance and innovation. But the pursuit of this goal takes place within a structure of convention, appraisal, and reward that shapes the field's development. These influences converge on the research nexus of publication. Publication is not only the means of communicating ideas; it also the emblem of an idea's validity. And publication plays a primary role in the success both of research projects and of the individual researcher.

Academic publication is governed by well-established conventions. Some disciplines have stricter conventions than computer science, dictating not just methodological standards, but paper layout, terminology, and even section headings. Computer

[6] `http://www.inex.otago.ac.nz/tracks/strong/strong.asp`

science, though, has its own standards, and within the discipline each field has its variants. These conventions determine what is good science, what is expected, what fits the model, and what is unexceptionable.

A research methodology is rooted and grows within this conventional and economic system. A clearly defined, familiar, and strong methodology scientifically validates experimental results, by making them testable and reproducible (though not necessarily useful or important). The possession of a strong methodology makes a field a more rigorous science, places it further up the ladder of scientific status, and enables it to look down on the fields below it and associate with the more respectable disciplines above. Once a field has taken possession of such a methodology, it is reluctant to relinquish it.

Above all, an established methodology plays a crucial role in the review process. Studies that employ such a ready-made methodology already have a head-start on work that must develop its own methods from scratch. Also, work that uses an established methodology is both familiar and (socially, if not scientifically) validated. It is, therefore, all the more readily published. The advantage that methodologically conventional work has over unconventional work is not dependent, though, on the real significance and innovation of the research. On the contrary, conventional work is likely to be less innovative and (once a methodology has been established long enough) more abstruse and over-fitted to the methodology's peculiarities.

The Cranfield "paradigm"

The preceding description of a science bound to a prescriptive methodology is an idealized, or perhaps demonized, one. It would be a Kuhnian caricature (Kuhn, 1970) to describe the field of information retrieval as constricted to the relevance-centric, test collection model of what constitutes the field's "science". Other questions are asked in the area, and other methods of evaluation are accepted and pursued. But certainly the test collection methodology exerts a pervasive, normative influence. The strongest advocates of the Cranfield tradition have even borrowed Kuhn's language, and taken to referring to "the Cranfield paradigm" (Voorhees, 2002, 2009a; Harman, 2010).

In Kuhn's account, however, a paradigm is both hindrance and asset: a strong model that validates a field, it is also a hard and inflexible one that holds it fixed and prevents it from changing. The tendency of the test collection methodology to perpetuate a line of research and publication that follows conventional lines, well after that line of research appears to have been exhausted—the methodology's *paradigmatic* tendency—can be seen when we observe the continuing, and even increasing, popularity of collection-based publications in Figure 8.6, despite the lack of improvement in the results they are reporting (observed in Section 8.2.2). Thus, when papers refer to their investigations as following "the Cranfield paradigm", as Figure 8.7 shows they are doing with increasing frequency, the self-characterization may be truer and more revealing than intended.

The test collection model exerts its influence directly and indirectly: directly, in the status it imparts to publications that employ the methodology (and withholds from those that eschew it); and indirectly, in setting a benchmark for empirical rigour. If researchers do not follow the standard methodology, they must at least provide comparable experimental validation. This can be viewed as "maintaining our standards": the field of IR has reached a high level of rigour, and the tradition must be sustained. But, necessarily, new fields will have less developed methodologies, and less satisfactory test data. Indeed, in the newest fields, particularly those inspired by application rather than theory, it will still be unclear precisely what the correct questions are.

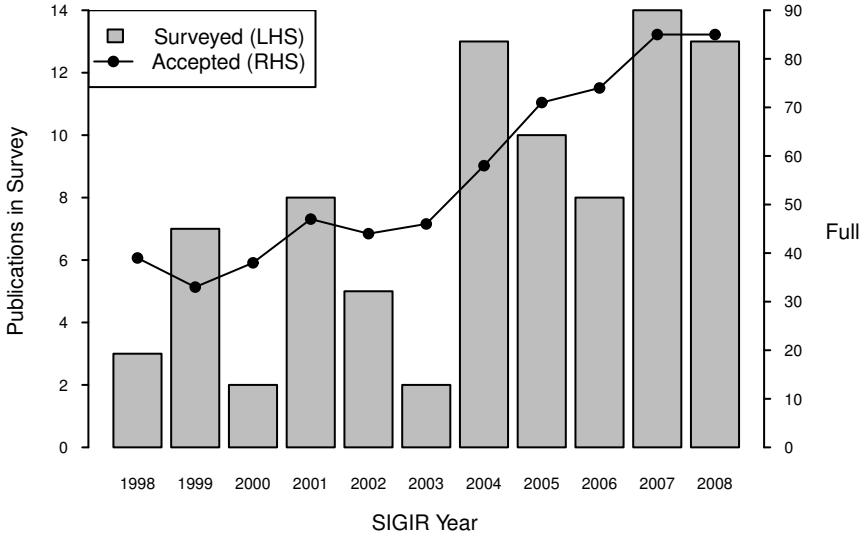


Figure 8.6: Number of papers at SIGIR using TREC ad hoc style collections, as included in the survey reported in Section 8.2.2 (left axis). The number of full papers accepted at each year's SIGIR is given (right axis), as a reference for conference size. The surveyed counts are not directly a proportion of the full paper counts, as the former include posters. Conversely, the survey only counted papers meeting the survey's particular criteria, and understates the number of publications using TREC ad hoc style test collections.

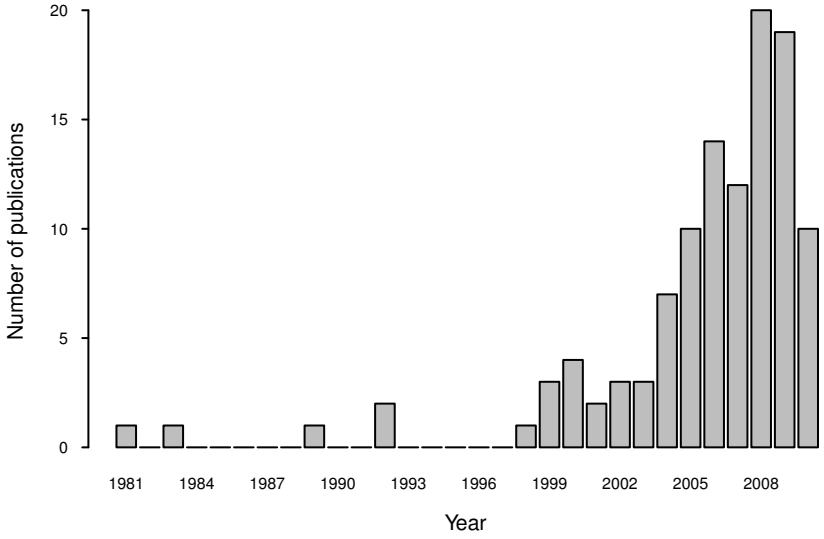


Figure 8.7: Number of academic publications using the phrase "Cranfield paradigm" per year, as reported by Google Scholar (retrieved: 13th July, 2010). Note that the final bar, for 2010, covers only half a year.

Finding evidence of an unwelcome publication bias caused by paradigmatic methodological preconceptions is difficult, because detailed conference and journal submission and rejection information is not available, and even if it were, it would be hard to analyze objectively. We met the same obstacle in Section 8.2.2 to testing the hypothesis that weak baselines help a paper achieve publication. As Church (2006) notes, it is much easier to calculate the precision than the recall of a conference. A notorious (and controversial) example of alleged methodological inflexibility is the rejection of the original Page Rank paper from SIGIR in 1998, for (amongst other things) a lack of experimental validation (Hersh, 2009), when there was at the time no suitable, publicly available dataset on which to test the new algorithm—though the Google project doubtless had the data internally.[7] As a technical report (Page et al., 1998), the paper has gone on to garner over 3,000 citations.[8] There are also less concrete, but more pervasive, complaints from researchers working outside the test collection model about the alleged rigidity of SIGIR's evaluation hurdle.[9]

It could be argued that our implicit methodological meta-standards (standards for what a methodology should be) have been developed in fields such as genetics and pharmacology, where the subject matter (in the latter case, humans) and type of question asked (does drug X cure disease Y?) change only gradually over time; and that such meta-standards are inappropriate in an applied information science, where both subject matter and questions of interest change rapidly. The dynamic environment of information science may make it impossible for a methodology to be both long-established and relevant. Technology evolves; scale balloons; information changes rapidly; and the nature of online communication between humans and organizations is in constant flux. By the time a problem has become concrete enough, and generated enough publicly available data, to be addressed to the highest standards of empirical thoroughness, the problem itself may have been passed by. To continue making useful progress, we may have to live with being a softer science than we would like.

Measurability and innovation

Much of this thesis has involved the methods of statistics; and one of the key practical lessons of statistics is to think not just about the data you see, but about the data you do not. Although quantitative methods themselves fail us on such a high-level domain as the sociology of methodology, considering the evidence you cannot see is crucial when examining the role of an experimental methodology in a scientific discipline. A strong and well-developed methodology is desirable, because it provides validation of results and measurement of progress. Desirable too are improvements in the correctness and rigour of method, such as have been proposed in this thesis. But neither rigour nor methodology are ends in themselves. And the presence of a strong, but dated, methodology within a scientific discipline can have a detrimental influence on that discipline, by straightjacketing research performed within the confines of the methodology, and depreciating research performed outside it. A field can find itself fixated on what is

[7] I owe this observation to Jeremy Pickens; see `http://blog.codalism.com/?p=984&cpage=1` (published 23rd September, 2009; last retrieved 25th July, 2010).

[8] Reported by Google Scholar, 16th July, 2010.

[9] "The SIGIR community is trapped by a very successful paradigm. People can do complex work, the quality of that work can be measured, and progress made. [...] The bigger problem is that a successful paradigm stifles innovation [...] We need to balance innovation (with its attendant imperfections) with more methodologically-mature work." Gene Golovchinsky, `http://palblog.fxpal.com/?p=4283` (published 22nd July, 2010; last retrieved 25th July, 2010).

measurable, not what is innovative, interesting, or important; statistical significance can drive out true significance.

In the dynamic environment of the information and computing sciences, it is difficult for a strong methodology to take hold. Researchers have only just delineated what the real problems underlying a new field are, when the field itself is transformed. When a strong methodology does emerge, it is understandable that information scientists are very reluctant to release their hold of it, and be thrown back upon experimental improvisation and uncertainty. In such an environment, a firm methodology is a real achievement, and a valuable one, but one in general of a limited life span. But when its proper life span is over, the scientific respectability it bestows to its adherents may impart the methodology with a long-extended ghostly afterlife.

What, then, are the practical consequences of this picture for the field of information retrieval, and for rigorous empiricists in information science in general? Not, certainly, to give up the pursuit of rigour, nor precipitously to abandon the relevance-based, ad hoc test collection model. But nor is it to idolize methodological rigour blindly. Rather, the need is to be practical, flexible, and enterprising. The achievements of methodologically developed fields should be used as a source of tools, examples, and experience, to sharpen the empirical edge of less experimentally developed disciplines. The innovative should be consciously preferred, and the stale consciously deprecated, because we work in an area where technology and applications become dated very quickly.

In other words, it is perhaps time for us to measure something new.

Appendix A

Proofs

A.1 Standardized score limits

A.1.1 Maximum standardized score of reference system

Proposition A.1 *The maximum absolute standardized system-topic score that a reference system can achieve is $\sqrt{n-1}$, where n is the number of reference systems (Section 4.3).*

Proof.

Let s_i be the raw score achieved by reference system i. Consider the situation in which $n-1$ reference systems achieve one score, x: $s_i = x, i \in \{1, \ldots, n-1\}$; and the remaining reference system achieves another score, y: $s_n = y, y \neq x$. The standardized score s'_n is the maximum absolute score achievable by any reference system; this will be proved later.

We wish to calculate the standardization score that system n achieves:

$$\begin{aligned} s'_n &= \frac{y - \bar{s}}{\sigma(s)} \\ &= \frac{y - \bar{s}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(s_i - \bar{s})^2}} \qquad \text{(A.1)} \\ &= \sqrt{\frac{n(y-\bar{s})^2}{(y-\bar{s})^2 + (n-1)(x-\bar{s})^2}} \qquad \text{(A.2)} \end{aligned}$$

Now:

$$\bar{s} = \frac{(n-1)x}{n} + \frac{1}{n}y$$

so:

$$\begin{aligned} x - \bar{s} &= \left(1 - \frac{n-1}{n}\right)x - \frac{y}{n} \\ &= \frac{1}{n}(x-y) \qquad \text{(A.3)} \end{aligned}$$

and:

$$\begin{aligned} y - \bar{s} &= \left(1 - \frac{1}{n}\right) y - \frac{n-1}{n} x \\ &= \frac{n-1}{n}(y - x) \end{aligned} \tag{A.4}$$

Substituting Equation A.3 and A.4 in Equation A.2 gives:

$$\begin{aligned} s'_n &= \sqrt{\frac{n\left(\frac{n-1}{n}\right)^2 (y-x)^2}{\left(\frac{n-1}{n}\right)^2 (y-x)^2 + \frac{n-1}{n^2}(x-y)^2}} \\ &= \sqrt{n-1} \end{aligned}$$

as required (note that $(x-y)^2 = (y-x)^2$). The positive root is taken if $y > \bar{s}$ (see steps A.1 and A.2 above); the negative if $y < \bar{s}$.

Thus, when all reference systems but one achieve the same score, the standardized score of the other is $\pm\sqrt{n-1}$. The final step is to prove that this situation is the one in which the maximum absolute standardized score is achieved.

Consider an alternative set of scores, t, with the same mean score $\bar{t} = \bar{s}$ and the same maximum raw score $t_n = s_n$. (Note that because the maximum standardized score of s depends solely on n, we can always find an equivalent set of scores s^* having the desired relationship to any set of scores t.) We wish to prove that $s'_n \geq t'_n$. Now:

$$s'_n = \frac{s_n - \bar{s}}{\sigma(s)}$$

while

$$\begin{aligned} t'_n &= \frac{t_n - \bar{t}}{\sigma(t)} \\ &= \frac{s_n - \bar{s}}{\sigma(t)} \end{aligned}$$

Thus, to prove that $s'_n \geq t'_n$, we need only prove that $\sigma(s) \leq \sigma(t)$, or in other words that the minimum standard deviation for t, given the conditions stated above, is that $t_i = t_j$ for all $i, j \neq n$. This actually follows from Jensen's inequality and the fact that $\sigma(s)$ is a convex function.

For instance, consider the case that t differs from s in having one value $t_k < s_i$. Let the difference $s_i - t_k$ be d. Therefore, $t_i = s_i + d/(n-2), i \notin \{n, k\}$. Now:

$$\sigma(s) = \frac{1}{n}\sqrt{(s_n - \bar{s})^2 + (n-1)(s_i - \bar{s})^2}$$

and:

$$\begin{aligned} \sigma(t) &= \frac{1}{n}\sqrt{(t_n - \bar{t})^2 + (t_k - \bar{t})^2 + (n-2)(t_i - \bar{t})^2} \\ &= \frac{1}{n}\sqrt{(s_n - \bar{s})^2 + (s_i - d - \bar{s})^2 + (n-2)(s_i + \frac{d}{n-2} - \bar{s})^2} \end{aligned}$$

Therefore, after some algebraic manipulation:

$$\sigma(t)^2 - \sigma(s)^2 = \frac{1}{n} \cdot \frac{(n-1)\, d^2}{n-2} > 0 \tag{A.5}$$

(since $d > 0$ and $n > 2$), from which it follows (since $\sigma(s) > 0$ and $\sigma(t) > 0$) that $\sigma(s) < \sigma(t)$.

□

A.2 Tail dominates prefix in AO

In this section, we prove that the tails of infinite rankings dominate the heads in the calculation of AO, as stated in Section 7.2.2.

Consider the weight given to each rank by the AO measure on lists of depth n. Rank 1 is contained in each of the n subsets. In the first subset, it determines the entire overlap; in the second subset, it determines half the overlap; in the third, a third of the overlap; and so forth. Therefore the weight of rank 1 is:

$$W_{\mathrm{AO}}(1,n) = \frac{1}{n}\left(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n-1} + \frac{1}{n}\right) = \frac{1}{n}\sum_{d=1}^{n}\frac{1}{d} = \frac{H_n}{n}$$

where $H_n \approx \gamma + \ln n + 1/(2n)$ is the nth Harmonic number, and $\gamma = 0.52771\ldots$ is the Euler-Mascheroni constant (see Knuth (1997, Section 1.2.7)). It follows that $W_{\mathrm{AO}}(2,n) = (H_n - H_1)/n$, that $W_{\mathrm{AO}}(3,n) = (H_n - H_2)/n$, and in general:

$$W_{\mathrm{AO}}(i,n) = \frac{H_n - H_{(i-1)}}{n}\ .$$

If only the prefix $k < n$ elements are available for each list, then the $\{1,\ldots,k\}$ heads of each list have contributed to the similarity measure, but the $\{k+1,\ldots,n\}$ tails have not. The cumulative weight of the head is:

$$\begin{aligned}
W_{\mathrm{AO}}^{head} &= \sum_{i=1}^{k} W_{\mathrm{AO}}(i,n) = \frac{1}{n}\sum_{i=1}^{k}\left(H_n - H_{(i-1)}\right) \approx \frac{1}{n}\ln\frac{n^k}{(k-1)!} \\
&= \frac{1}{n}\left[\ln n^k - \ln(k-1)!\right] \\
&\approx \frac{1}{n}\left[k\ln n - (k-1)\ln(k-1) + k - 1\right] \qquad \text{(A.6)} \\
&\approx \frac{k}{n}\left[\ln n - \ln k + 1\right] \\
&= \frac{k}{n}\ln\frac{n}{k} + \frac{k}{n} \qquad \text{(A.7)}
\end{aligned}$$

where the simplification at Equation A.6 uses Stirling's approximation, $\ln x! \approx x\ln x - x$. Equation A.7 goes to 0 as $n \to \infty$ and k is fixed.

The cumulative weight of the tail, following a similar line of simplification, is:

$$\begin{aligned}
W_{\mathrm{AO}}^{tail} &= \sum_{i=k+1}^{n} W_{\mathrm{AO}}(i,n) = \frac{1}{n}\sum_{i=k+1}^{n}\left(H_n - H_{(i-1)}\right) \\
&\approx \frac{1}{n}\ln\frac{n^{(n-k)}(k-1)!}{(n-1)!} \\
&\approx 1 - \frac{k}{n}\ln\frac{n}{k} - \frac{k}{n} \qquad \text{(A.8)}
\end{aligned}$$

which goes to 1 as $n \to \infty$ and k is fixed. Therefore, for an infinite list, the weight of the tail is 1, and of the head is 0, proving that the tail dominates the head.

□

Bibliography

S. Agrawal, S. Chaudhuri, and G. Das. DBXplorer: a system for keyword-based search over relational databases. In R. Agrawal, K. Dittrich, and A. H. H. Ngu, editors, *Proc. 18th International Conference on Data Engineering*, pages 5–16, San Jose, California, Feb. 2002.

A. Al-Maskari, M. Sanderson, and P. Clough. The relationship between IR effectiveness measures and user satisfaction. In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 773–774, Amsterdam, the Netherlands, July 2007.

J. Allan, B. Carterette, J. Aslam, V. Pavlu, B. Dachev, and E. Kanoulas. Million query track 2007 overview. In E. Voorhees and L. P. Buckland, editors, *Proc. 16th Text REtrieval Conference*, pages 6:1–20, Gaithersburg, Maryland, USA, Nov. 2007. NIST Special Publication 500-274.

T. G. Armstrong, A. Moffat, W. Webber, and J. Zobel. Improvements that don't add up: Ad-hoc retrieval results since 1998. In D. Cheung, I.-Y. Song, W. Chu, X. Hu, J. Lin, J. Li, and Z. Peng, editors, *Proc. 18th ACM International Conference on Information and Knowledge Management*, pages 601–610, Hong Kong, China, Nov. 2009a.

T. G. Armstrong, A. Moffat, W. Webber, and J. Zobel. EvaluatIR: An online tool for evaluating and comparing IR systems. In J. Allan, J. Aslam, M. Sanderson, C. Zhai, and J. Zobel, editors, *Proc. 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 834, Boston, Massachusetts, USA, July 2009b.

T. G. Armstrong, A. Moffat, W. Webber, and J. Zobel. Has adhoc retrieval improved since 1994? In J. Allan, J. Aslam, M. Sanderson, C. Zhai, and J. Zobel, editors, *Proc. 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 692–693, Boston, Massachusetts, USA, July 2009c.

J. Aslam and E. Yilmaz. A geometric interpretation and analysis of R-precision. In O. Herzog, H.-J. Schek, N. Fuhr, A. Chowdhury, and W. Teiken, editors, *Proc. 14th ACM International Conference on Information and Knowledge Management*, pages 664–671, Bremen, Germany, Nov. 2005.

J. Aslam, V. Pavlu, and R. Savell. A unified model for metasearch, pooling, and system evaluation. In D. Kraft, O. Frieder, J. Hammer, S. Qureshi, and L. Seligman, editors, *Proc. 12th ACM International Conference on Information and Knowledge Management*, pages 484–491, New Orleans, Louisiana, USA, Nov. 2003.

J. Aslam, E. Yilmaz, and V. Pavlu. The maximum entropy method for analyzing retrieval measures. In G. Marchionini, A. Moffat, J. Tait, R. Baeza-Yates, and N. Ziviani, editors, *Proc. 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 27–34, Salvador, Brazil, Aug. 2005.

J. Aslam, V. Pavlu, and E. Yilmaz. A statistical method for system evaluation using incomplete judgments. In S. Dumais, E. Efthimiadis, D. Hawking, and K. Järvelin, editors, *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 541–548, Seattle, Washington, USA, Aug. 2006.

P. Bailey, N. Craswell, and D. Hawking. Engineering a multi-purpose test collection for Web retrieval experiments. *Information Processing & Management*, 39(6):853–871, 2003.

D. Banks, P. Over, and N.-F. Zhang. Blind men and elephants: six approaches to TREC data. *Information Retrieval*, 1(1):7–34, Apr. 1999.

J. Bar-Ilan. Comparing rankings of search results on the Web. *Information Processing & Management*, 41(6):1511–1519, 2005.

J. Bar-Ilan, M. Mat-Hassan, and M. Levene. Methods for comparing rankings of search engine results. *Computer Networks*, 50(10):1448–1463, July 2006.

J. R. Baron, D. D. Lewis, and D. W. Oard. TREC-2006 legal track overview. In E. Voorhees and L. P. Buckland, editors, *Proc. 15th Text REtrieval Conference*, pages 79–98, Gaithersburg, Maryland, USA, Nov. 2006. NIST Special Publication 500-272.

N. J. Belkin. Ineffable concepts in information retrieval. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 4, pages 44–58. Butterworths, 1981.

M. Bernstein, R. C. Miller, G. Little, M. Ackerman, B. Hartmann, D. R. Karger, and K. Panovich. Soylent: A word processor with a crowd inside. In *Proc. 23rd Annual ACM Symposium on User Interface Software and Technology*, New York, New York, USA, Oct. 2010. to appear.

D. C. Blair. Some thoughts on the reported results of TREC. *Information Processing & Management*, 38(3):445–451, 2002.

D. Bodoff and P. Li. Test theory for assessing IR test collections. In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 367–374, Amsterdam, the Netherlands, July 2007.

B. Boyce. Beyond topicality: A two stage view of relevance and the retrieval process. *Information Processing & Management*, 18(3):105–109, 1982.

R. L. Brennan. *Generalizability Theory*. Springer, New York, 2001.

C. Buckley. The SMART project at TREC. In Voorhees and Harman (2005a), chapter 13.

C. Buckley. Topic prediction based on comparative retrieval rankings. In M. Sanderson, K. Järvelin, J. Allan, and P. Bruza, editors, *Proc. 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 506–507, Sheffield, United Kingdom, Aug. 2004.

C. Buckley. TREC 6 high-precision track. In E. Voorhees and D. Harman, editors, *Proc. 6th Text REtrieval Conference*, pages 69–72, Gaithersburg, Maryland, USA, Nov. 1997. NIST Special Publication 500-240.

C. Buckley. The TREC 7 query track. In E. Voorhees and D. Harman, editors, *Proc. 7th Text REtrieval Conference*, pages 73–78, Gaithersburg, Maryland, USA, Nov. 1998. NIST Special Publication 500-242.

C. Buckley and S. Robertson. Relevance feedback track overview: TREC 2008. In E. Voorhees and L. P. Buckland, editors, *Proc. 17th Text REtrieval Conference*, pages 4:1–13, Gaithersburg, Maryland, USA, Nov. 2008. NIST Special Publication 500-277.

C. Buckley and E. Voorhees. Evaluating evaluation measure stability. In E. Yannakoudis, N. J. Belkin, M.-K. Leong, and P. Ingwersen, editors, *Proc. 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 33–40, Athens, Greece, Aug. 2000.

C. Buckley and E. Voorhees. Retrieval evaluation with incomplete information. In M. Sanderson, K. Järvelin, J. Allan, and P. Bruza, editors, *Proc. 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 25–32, Sheffield, United Kingdom, Aug. 2004.

C. Buckley and E. Voorhees. Retrieval system evaluation. In Voorhees and Harman (2005a), chapter 3.

C. Buckley, A. Singhal, and M. Mitra. Using query zoning and correlation within SMART: TREC 5. In E. Voorhees and D. Harman, editors, *Proc. 5th Text REtrieval Conference*, pages 105–118, Gaithersburg, Maryland, USA, Nov. 1996. NIST Special Publication 500-238.

C. Buckley, D. Dimmick, I. Soboroff, and E. Voorhees. Bias and the limits of pooling for large collections. *Information Retrieval*, 10(6):491–508, Dec. 2007.

C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In L. de Raedt and S. Wrobel, editors, *Proc. 22nd International Conference on Machine Learning*, pages 89–96, Bonn, Germany, Aug. 2005.

J. Callan and M. Hoy. The ClueWeb09 dataset, 2009. URL `http://boston.lti.cs.cmu.edu/Data/clueweb09`. Last accessed: 2009-11-13.

J. Callan, J. Allan, C. L. A. Clarke, S. Dumais, D. A. Evans, M. Sanderson, and C. Zhai. Meeting of the minds: an information retrieval research agenda. *SIGIR Forum*, 41 (2):25–34, 2007.

B. Carterette. Robust test collections for retrieval evaluation. In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 55–62, Amsterdam, the Netherlands, July 2007.

B. Carterette. On rank correlation and the distance between rankings. In J. Allan, J. Aslam, M. Sanderson, C. Zhai, and J. Zobel, editors, *Proc. 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 436–443, Boston, Massachusetts, USA, July 2009.

B. Carterette and J. Allan. Incremental test collections. In O. Herzog, H.-J. Schek, N. Fuhr, A. Chowdhury, and W. Teiken, editors, *Proc. 14th ACM International Conference on Information and Knowledge Management*, pages 680–687, Bremen, Germany, Nov. 2005.

B. Carterette and M. D. Smucker. Hypothesis testing with incomplete relevance judgments. In M. J. Silvaa, A. A. F. Laender, R. Baeza-Yates, D. L. McGuinness, B. Olstad, Ø. H. Olsen, and A. O. Falcão, editors, *Proc. 16th ACM International Conference on Information and Knowledge Management*, pages 643–652, Lisboa, Portugal, Nov. 2007.

B. Carterette, J. Allan, and R. Sitaraman. Minimal test collections for retrieval evaluation. In S. Dumais, E. Efthimiadis, D. Hawking, and K. Järvelin, editors, *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 268–275, Seattle, Washington, USA, Aug. 2006.

B. Carterette, V. Pavlu, E. Kanoulas, J. Aslam, and J. Allan. Evaluation over thousands of queries. In Myaeng et al. (2008), pages 651–658.

O. Chapelle, D. A. Metzler, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In D. Cheung, I.-Y. Song, W. Chu, X. Hu, J. Lin, J. Li, and Z. Peng, editors, *Proc. 18th ACM International Conference on Information and Knowledge Management*, pages 621–630, Hong Kong, China, Nov. 2009.

K. Church. Reviewing the reviewers. *Computational Linguistics*, 31(4):575–578, 2006.

C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2004 terabyte track. In E. Voorhees and L. P. Buckland, editors, *Proc. 13th Text REtrieval Conference*, pages 7:1–9, Gaithersburg, Maryland, USA, Nov. 2004. NIST Special Publication 500-261.

C. L. A. Clarke, F. Scholer, and I. Soboroff. The TREC 2005 terabyte track. In E. Voorhees and L. P. Buckland, editors, *Proc. 14th Text REtrieval Conference*, pages 8:1–11, Gaithersburg, Maryland, USA, Nov. 2005. NIST Special Publication 500-266.

C. L. A. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In Myaeng et al. (2008), pages 659–666.

C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 web track. In E. Voorhees and L. P. Buckland, editors, *Proc. 18th Text REtrieval Conference*, pages 1:4:1–9, Gaithersburg, Maryland, USA, Nov. 2009. NIST Special Publication 500-278.

C. W. Cleverdon. *Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems*. Aslib Cranfield Research Project, Cranfield, 1962.

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

C. W. Cleverdon. The significance of the Cranfield tests on index languages. In A. Bookstein, Y. Chiaramella, G. Salton, and V. V. Raghavan, editors, *Proc. 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12, Chicago, Illinois, USA, Oct. 1991.

C. W. Cleverdon and E. M. Keen. *Factors Determining the Performance of Indexing Systems. Volume 2.*, volume 2. Aslib Cranfield Research Project, Cranfield, 1966.

C. W. Cleverdon, J. Mills, and E. M. Keen. *Factors Determining the Performance of Indexing Systems. Volume 1: Design*, volume 1. Aslib Cranfield Research Project, Cranfield, 1966.

N. Cliff. *Ordinal Methods for Behavioural Data Analysis*. Lawrence Erlbaum Associates, 1996.

D. B. H. Cline and J. D. Hart. Kernel estimation of densities with discontinuities or discontinuous derivatives. *Statistics*, 22(1):69–84, 1991.

J. Coffman and A. C. Weaver. A framework for evaluating keyword search strategies. In *Proc. 19th ACM International Conference on Information and Knowledge Management*, Toronto, Canada, Oct. 2010. to appear.

J. Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, 2nd edition, 1988.

W. S. Cooper. On selecting a measure of retrieval effectiveness. *Journal of the American Society for Information Science*, 24(2):87–100, 1973.

G. V. Cormack and T. R. Lynam. Statistical precision of information retrieval evaluation. In S. Dumais, E. Efthimiadis, D. Hawking, and K. Järvelin, editors, *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 533–540, Seattle, Washington, USA, Aug. 2006.

G. V. Cormack and T. R. Lynam. Validity and power of t-test for comparing MAP and GMAP. In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 753–754, Amsterdam, the Netherlands, July 2007.

G. V. Cormack, C. R. Palmer, and C. L. A. Clarke. Efficient construction of large test collections. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proc. 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 282–289, Melbourne, Australia, Aug. 1998.

N. Craswell and D. Hawking. Overview of the TREC-2002 web track. In E. Voorhees and L. P. Buckland, editors, *Proc. 11th Text REtrieval Conference*, pages 8:1–10, Gaithersburg, Maryland, USA, Nov. 2002. NIST Special Publication 500-251.

N. Craswell, D. Hawking, R. Wilkinson, and M. Wu. Overview of the TREC 2003 web track. In E. Voorhees and L. P. Buckland, editors, *Proc. 12th Text REtrieval Conference*, pages 7:1–15, Gaithersburg, Maryland, USA, Nov. 2003. NIST Special Publication 500-255.

A. C. Davison and D. V. Hinkley. *Bootstrap Methods and their Application*. Cambridge University Press, 1997.

J. De Beer and M.-F. Moens. Rpref – a generalization of Bpref towards graded relevance judgments. In S. Dumais, E. Efthimiadis, D. Hawking, and K. Järvelin, editors, *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 637–638, Seattle, Washington, USA, Aug. 2006.

S. Dumais and N. J. Belkin. The TREC interactive tracks: putting the user into search. In Voorhees and Harman (2005a), chapter 6.

B. Efron and R. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1993.

R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. *SIAM Journal on Discrete Mathematics*, 17(1):134–160, 2003.

E. A. Fox. Characterization of two new experimental collections in computer and information science containing textual and bibliographic concepts. Technical Report TR 83-561, Cornell University, Ithaca, New York, 1983.

J. D. Gibbons and S. Chakraborti. *Nonparametric Statistical Inference*. CRC, 4th edition, 2003.

L. A. Goodman and W. H. Kruskal. Measures of association for cross classifications. *Journal of the American Statistical Association*, 49(268):732–764, 1954.

N. Gövert and G. Kazai. Overview of the initiative for the evaluation of XML retrieval (INEX) 2002. In N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, editors, *Proceedings of the First Workshop of the INitiative for the Evaluation of XML Retrieval*, pages 1–17, Schloss Dagstuhl, Germany, Dec. 2002.

D. Harman. The TREC test collections. In Voorhees and Harman (2005a), chapter 2.

D. Harman. The TREC ad hoc experiments. In Voorhees and Harman (2005a), chapter 4.

D. Harman. Beyond English. In Voorhees and Harman (2005a), chapter 7.

D. Harman. Is the Cranfield paradigm outdated? In H.-H. Chen, E. N. Efthimiadis, J. Savoy, F. Crestani, and S. Marchand-Maillet, editors, *Proc. 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 1, Geneva, Switzerland, July 2010.

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

D. Harman. Overview of the TREC 2002 novelty track. In E. Voorhees and L. P. Buckland, editors, *Proc. 11th Text REtrieval Conference*, pages 5:1–1, Gaithersburg, Maryland, USA, Nov. 2002. NIST Special Publication 500-251.

D. Harman. Overview of the first text REtrieval conference (TREC-1). In D. Harman, editor, *Proc. 1st Text REtrieval Conference*, pages 1–30, Gaithersburg, Maryland, USA, Nov. 1992b. NIST Special Publication 500-207.

D. Harman. Overview of the third text REtrieval conference (TREC-3). In D. Harman, editor, *Proc. 3rd Text REtrieval Conference*, pages 1–19, Gaithersburg, Maryland, USA, Nov. 1994. NIST Special Publication 500-225.

D. Harman. Overview of the fourth text REtrieval conference (TREC-4). In D. Harman, editor, *Proc. 4th Text REtrieval Conference*, pages 1–23, Gaithersburg, Maryland, USA, Nov. 1995. NIST Special Publication 500-236.

D. Hawking. Overview of the TREC-9 web track. In E. Voorhees and D. Harman, editors, *Proc. 9th Text REtrieval Conference*, pages 87–102, Gaithersburg, Maryland, USA, Nov. 2000. NIST Special Publication 500-249.

D. Hawking and N. Craswell. The very large collection and web tracks. In Voorhees and Harman (2005a), chapter 9.

D. Hawking and N. Craswell. Overview of the TREC-2001 web track. In E. Voorhees and D. Harman, editors, *Proc. 10th Text REtrieval Conference*, pages 7:1–8, Gaithersburg, Maryland, USA, Nov. 2001. NIST Special Publication 500-250.

D. Hawking, N. Craswell, and P. Thistlewaite. Overview of the TREC-7 very large collection track. In E. Voorhees and D. Harman, editors, *Proc. 7th Text REtrieval Conference*, pages 1–13, Gaithersburg, Maryland, USA, Nov. 1998. NIST Special Publication 500-242.

D. Hawking, E. Voorhees, N. Craswell, and P. Bailey. Overview of the TREC-8 web track. In E. Voorhees and D. Harman, editors, *Proc. 8th Text REtrieval Conference*, pages 1–18, Gaithersburg, Maryland, USA, Nov. 1999. NIST Special Publication 500-246.

W. L. Hays. *Statistics*. Harcourt Brace, Fort Worth, 4th edition, 1991.

W. Hersh. *Information Retrieval: a health and biomedical perspective*. Springer, 2009.

W. Hersh and R. T. Bhupatiraju. TREC genomics track overview. In E. Voorhees and L. P. Buckland, editors, *Proc. 12th Text REtrieval Conference*, pages 14–23, Gaithersburg, Maryland, USA, Nov. 2003. NIST Special Publication 500-255.

S. B. Huffman and M. Hochster. How well does result relevance predict session satisfaction? In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 567–574, Amsterdam, the Netherlands, July 2007.

K. Järvelin and J. Kekäläinen. IR evaluation methods for retrieving highly relevant documents. In E. Yannakoudis, N. J. Belkin, M.-K. Leong, and P. Ingwersen, editors, *Proc. 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 41–48, Athens, Greece, Aug. 2000.

K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In G. Marchionini, A. Moffat, J. Tait, R. Baeza-Yates, and N. Ziviani, editors, *Proc. 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 154–161, Salvador, Brazil, Aug. 2005.

M. C. Jones. Simple boundary correction for kernel density estimation. *Statistics and Computing*, 3(3):135–146, 1993.

N. Kando, editor. *Proc. 1st NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, Tokyo, Japan, Aug. 1999.

E. Kanoulas and J. Aslam. Empirical justification of the gain and discount function for nDCG. In D. Cheung, I.-Y. Song, W. Chu, X. Hu, J. Lin, J. Li, and Z. Peng, editors, *Proc. 18th ACM International Conference on Information and Knowledge Management*, pages 611–620, Hong Kong, China, Nov. 2009.

M. G. Kendall. *Rank Correlation Methods*. Charles Griffin, London, 1st edition, 1948.

L. Kirkup and R. B. Frenkel. *An Introduction to Uncertainty in Measurement*. Cambridge University Press, 2006.

D. Knuth. *The Art of Computer Programming*, volume 1. Addison Wesley, 3rd edition, 1997.

T. S. Kuhn. *The Structure of Scientific Revolutions*. University of Chicago Press, Chicago, 1970.

R. Kumar and S. Vassilvitskii. Generalized distances between rankings. In M. Rappa, P. Jones, J. Freire, and S. Chakrabarti, editors, *Proc. 19th International Conference on World Wide Web*, pages 571–580, Raleigh, North Carolina, USA, Apr. 2010.

R. Ledwith. On the difficulties of applying the results of information retrieval research to aid in the searching of large scientific databases. *Information Processing & Management*, 28(4):451–455, 1992.

M. E. Lesk. Operating instructions for the SMART text processing and document retrieval system. In Salton (1966b), chapter 2, pages 1–63. Issued as Scientific Report No. ISR-11.

J. T. Lessler and W. D. Kalsbeek. *Nonsampling Error in Surveys*. John Wiley & Sons, 1992.

N. Lester, A. Moffat, W. Webber, and J. Zobel. Space-limited ranked query evaluation using adaptive pruning. In A. H. Ngu, M. Kitsuregawa, E. J. Neuhold, J.-Y. Chung, and Q. Z. Sheng, editors, *Proc. 6th International Conference on Web Informations Systems Engineering*, volume 3806 of *Lecture Notes in Computer Science*, pages 470–477, New York, New York, USA, Nov. 2005.

G. Li, B. C. Ooi, J. Feng, J. Wang, and L. Zhou. EASE: an effective 3-in-1 keyword search method for unstructured, semi-structured and structured data. In D. Shasha, L. V. S. Lakshmanan, R. T. Ng, and J. Wang, editors, *Proc. 32nd ACM SIGMOD International Conference on Management of Data*, pages 903–914, Vancouver, Canada, June 2008.

F. Liu, C. Yu, M. Weiyi, and A. Chowdhury. Effective keyword search in relational databases. In S. Chaudhuri, V. Hristidis, and N. Polyzotis, editors, *Proc. 32nd ACM SIGMOD International Conference on Management of Data*, pages 563–574, Chicago, Illinois, USA, June 2006.

S. Liu, C. Yu, and M. Weiyi. Word sense disambiguation in queries. In O. Herzog, H.-J. Schek, N. Fuhr, A. Chowdhury, and W. Teiken, editors, *Proc. 14th ACM International Conference on Information and Knowledge Management*, pages 525–532, Bremen, Germany, Nov. 2005.

H. P. Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4):309–317, 1957.

Y. Luo, X. Lin, W. Wang, and X. Zhou. SPARK: top-k keyword query in relational databases. In C. Y. Chan, B. C. Ooi, and A. Zhou, editors, *Proc. 33rd ACM SIGMOD International Conference on Management of Data*, pages 115–126, Beijing, China, June 2007.

M. E. Maron and J. L. Kuhns. On relevance, probabilistic indexing and information retrieval. *Journal of the ACM*, 7(3):216–244, 1960.

J. S. Marron and D. Ruppert. Transformations to reduce boundary bias in kernel density estimation. *Journal of the Royal Statistical Society, Series B*, 56(4):653–671, 1994.

M. Melucci. Weighted rank correlation in information retrieval evaluation. In G. G. Lee, D. Song, C.-Y. Lin, A. Aizawa, K. Kuriyama, M. Yoshioka, and T. Sakai, editors, *Proc. 5th Asia Information Retrieval Symposium*, volume 5839 of *Lecture Notes in Computer Science*, pages 75–86, Sapporo, Japan, Oct. 2009.

S. Mizzaro. Relevance: The whole history. *Journal of the American Society for Information Science and Technology*, 48(3):810–832, 1997.

A. Moffat. Seven properties of effectiveness metrics, 2010. Manuscript.

A. Moffat and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems*, 27(1):1–27, 2008.

A. Moffat, W. Webber, and J. Zobel. Strategic system comparisons via targeted relevance judgments. In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 375–382, Amsterdam, the Netherlands, July 2007.

N. Mukhopadhyay and B. M. de Silva. *Sequential Methods and Their Applications*. Chapman and Hall/CRC, 2009.

S.-H. Myaeng, D. W. Oard, F. Sebastiani, T.-S. Chua, and M.-K. Leong, editors. *Proc. 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Singapore, Singapore, July 2008.

M. Najork and N. Craswell. Efficient and effective link analysis with precomputed SALSA maps. In J. G. Shanahan, S. Amer-Yahia, Y. Zhang, A. Kolcz, A. Chowdhury, and D. Kelly, editors, *Proc. 17th ACM International Conference on Information and Knowledge Management*, pages 53–61, Napa, California, USA, Oct. 2008.

NIST. Guidelines for TREC-8, 1999. URL `http://trec.nist.gov/act_part/guidelines/trec8_guides.html`. Last accessed: 2009-11-24.

R. Oddy. Laboratory tests: automatic systems. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 9, pages 156–178. Butterworths, 1981.

I. Ounis, M. de Rijke, C. Macdonald, G. Mishne, and I. Soboroff. Overview of the TREC-2006 blog track. In E. Voorhees and L. P. Buckland, editors, *Proc. 15th Text REtrieval Conference*, pages 17–31, Gaithersburg, Maryland, USA, Nov. 2006. NIST Special Publication 500-272.

P. Over. TREC-6 interactive track report. In E. Voorhees and D. Harman, editors, *Proc. 6th Text REtrieval Conference*, pages 73–82, Gaithersburg, Maryland, USA, Nov. 1997. NIST Special Publication 500-240.

L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

C. Peters, editor. *Proc. 1st Workshop of the Cross-Lingual Evaluation Forum*, volume 2069 of *Lecture Notes in Computer Science*, Lisbon, Portugal, Sept. 2000.

J. D. Prange. Evaluation driven research: The foundation of the TIPSTER text program. In *Proceedings of the TIPSTER Text Program: Phase II*, pages 13–22, Vienna, Virginia, USA, May 1996. Association for Computational Linguistics.

(Pseudo-)Google. Google guidelines for quality raters, April 2007. URL `http://www.mauriziopetrone.com/blog/wp-content/uploads/quality-rater-guidelines-2007.pdf`

S. D. Ravana and A. Moffat. Exploring evaluation metrics: GMAP versus MAP. In Myaeng et al. (2008), pages 687–688.

S. D. Ravana and A. Moffat. Score aggregation techniques in retrieval experimentation. In A. Bouguettaya and X. Lin, editors, *Proc. 20th Australasian Database Conference*, volume 92 of *Conferences in Research and Practice in Information Technology*, pages 59–67, Wellington, New Zealand, Jan. 2009.

S. Robertson. On the early history of evaluation in IR. In J. Tait, editor, *Charting a New Course: Natural Language Processing and Information Retrieval – Essays in Honour of Karen Spärck Jones*, pages 13–22. Springer, 2005.

S. Robertson. On GMAP: and other transformations. In P. S. Yu, V. Tsotras, E. A. Fox, and B. Liu, editors, *Proc. 15th ACM International Conference on Information and Knowledge Management*, pages 78–83, Arlington, Virginia, USA, Nov. 2006.

S. Robertson. On the history of evaluation in IR. *Journal of Information Science*, 34 (4):439–456, 2008a.

S. Robertson. A new interpretation of average precision. In Myaeng et al. (2008), pages 689–690.

S. Robertson. Richer theories, richer experiments. In S. Geva, J. Kamps, C. Peters, T. Sakai, A. Trotman, and E. Voorhees, editors, *Proc.SIGIR 2009 Workshop on the Future of IR Evaluation*, page 1, Boston, Massachusetts, USA, July 2009.

S. Robertson. The probability ranking principle in IR. *Journal of Documentation*, 33 (4):294–304, 1977.

S. Robertson. The methodology of information retrieval experiment. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 2, pages 9–31. Butterworths, 1981.

S. Robertson. On samples sizes for non-matched-pair IR experiments. *Information Processing & Management*, 26(6):739–753, 1990.

S. Robertson and J. Callan. Routing and filtering. In Voorhees and Harman (2005a), chapter 5.

S. Robertson and M. M. Hancock-Beaulieu. On the evaluation of IR systems. *Information Processing & Management*, 28(4):457–466, 1992.

S. Robertson and I. Soboroff. The TREC 2002 filtering track report. In E. Voorhees and L. P. Buckland, editors, *Proc. 11th Text REtrieval Conference*, pages 3:1–13, Gaithersburg, Maryland, USA, Nov. 2002. NIST Special Publication 500-251.

S. Robertson and C. L. Thompson. Weighted searching: the CIRT experiment. In K. P. Jones, editor, *Informatics 10: Prospects for Intelligent Retrieval*, pages 75–89, London, 1990.

S. Robertson, S. Walker, M. M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at TREC. In D. Harman, editor, *Proc. 1st Text REtrieval Conference*, pages 21–30, Gaithersburg, Maryland, USA, Nov. 1992. NIST Special Publication 500-207.

S. Robertson, S. Walker, and M. M. Beaulieu. Experimentation as a way of life: Okapi at TREC. *Information Processing & Management*, 36(1):95–108, 2000.

J. J. Rocchio. *Document retrieval systems – optimization and evaluation*. PhD thesis, Harvard University, 1966. Issued as Scientific Report No. ISR-10.

T. Sakai. Evaluating evaluation metrics based on the bootstrap. In S. Dumais, E. Efthimiadis, D. Hawking, and K. Järvelin, editors, *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 525–532, Seattle, Washington, USA, Aug. 2006.

T. Sakai. On penalising late arrival of relevant documents in information retrieval with graded relevance. In T. Sakai, M. Sanderson, and D. K. Evans, editors, *Proc. 1st International Workshop on Evaluating Information Access*, pages 32–43, Tokyo, Japan, May 2007a.

T. Sakai. Alternatives to Bpref. In C. L. A. Clarke, N. Fuhr, N. Kando, W. Kraaij, and A. de Vries, editors, *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 71–78, Amsterdam, the Netherlands, July 2007b.

T. Sakai. Comparing metrics across TREC and NTCIR: The robustness to system bias. In J. G. Shanahan, S. Amer-Yahia, Y. Zhang, A. Kolcz, A. Chowdhury, and D. Kelly, editors, *Proc. 17th ACM International Conference on Information and Knowledge Management*, pages 581–590, Napa, California, USA, Oct. 2008.

G. Salton. The SMART system – retrieval results and future plans. In *Information Storage and Retrieval* Salton (1966b), chapter 1, pages 1–9. Issued as Scientific Report No. ISR-11.

G. Salton, editor. *The SMART Retrieval System – Experiments in Automatic Document Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1971.

G. Salton. The Smart environment for retrieval system evaluation–advantages and problem areas. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 15, pages 316–329. Butterworths, 1981.

G. Salton. The state of retrieval system evaluation. *Information Processing & Management*, 28(4):441–449, 1992.

G. Salton, editor. *Information Storage and Retrieval*. Cornell University, Ithaca, New York, 1966b. Issued as Scientific Report No. ISR-11.

G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 44(4):288–297, 1990.

G. Salton and M. E. Lesk. The SMART automatic document retrieval systems–an illustration. *Communications of the ACM*, 8(6):391–398, 1965.

M. Sanderson and H. Joho. Forming test collections with no system pooling. In M. Sanderson, K. Järvelin, J. Allan, and P. Bruza, editors, *Proc. 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 33–40, Sheffield, United Kingdom, Aug. 2004.

M. Sanderson and J. Zobel. Information retrieval system evaluation: effort, sensitivity, and reliability. In G. Marchionini, A. Moffat, J. Tait, R. Baeza-Yates, and N. Ziviani, editors, *Proc. 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169, Salvador, Brazil, Aug. 2005.

M. Sanderson, M. L. Paramita, P. Clough, and E. Kanoulas. Do user preferences and evaluation measures line up? In H.-H. Chen, E. N. Efthimiadis, J. Savoy, F. Crestani, and S. Marchand-Maillet, editors, *Proc. 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 555–562, Geneva, Switzerland, July 2010.

T. Saracevic. In memoriam: Gerald Salton (1927–1995). *Information Processing & Management*, 31(6):787–788, 1995.

T. Saracevic, P. Kantor, A. Y. Chamis, and D. Trivison. A study of information seeking and retrieving. i. background and methodology. *Journal of the American Society for Information Science and Technology*, 39:161–176, 1988.

J. Savoy. Statistical inference in retrieval effectiveness evaluation. *Information Processing & Management*, 33(4):495–512, 1997.

G. S. Shieh. A weighted Kendall's tau statistic. *Statistics & Probability Letters*, 39: 17–24, 1998.

D. Siegmund. *Sequential Analysis: Tests and Confidence Intervals*. Springer, 1985.

B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

A. F. Smeaton, P. Over, and R. Taban. The TREC-2001 video track report. In E. Voorhees and D. Harman, editors, *Proc. 10th Text REtrieval Conference*, pages 6:1–9, Gaithersburg, Maryland, USA, Nov. 2001. NIST Special Publication 500-250.

M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In M. J. Silvaa, A. A. F. Laender, R. Baeza-Yates, D. L. McGuinness, B. Olstad, Ø. H. Olsen, and A. O. Falcão, editors, *Proc. 16th ACM International Conference on Information and Knowledge Management*, pages 623–632, Lisboa, Portugal, Nov. 2007.

I. Soboroff. Does WT10g look like the web? In K. Järvelin, M. Beaulieu, R. Baeza-Yates, and S.-H. Myaeng, editors, *Proc. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 423–424, Tampere, Finland, Aug. 2002.

I. Soboroff and S. Robertson. Building a filtering test collection for TREC 2002. In C. L. A. Clarke, G. V. Cormack, J. Callan, D. Hawking, and A. Smeaton, editors, *Proc. 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 243–250, Toronto, Canada, July 2003.

I. Soboroff, C. Nicholas, and P. Cahan. Ranking retrieval systems without relevance judgments. In W. B. Croft, D. J. Harper, D. H. Kraft, and J. Zobel, editors, *Proc. 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 66–73, New Orleans, Louisiana, USA, Sept. 2001.

I. Soboroff, A. de Vries, and N. Craswell. Overview of the TREC 2006 enterprise track. In E. Voorhees and L. P. Buckland, editors, *Proc. 15th Text REtrieval Conference*, pages 32–51, Gaithersburg, Maryland, USA, Nov. 2006. NIST Special Publication 500-272.

K. Spärck Jones. Further reflections on TREC. *Information Processing & Management*, 36(1):37–85, 2000.

K. Spärck Jones. Retrieval system tests 1958–1978. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 12, pages 213–255. Butterworths, 1981a.

K. Spärck Jones. The Cranfield tests. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 13, pages 256–284. Butterworths, 1981b.

K. Spärck Jones and C. J. van Rijsbergen. Report on the need for and provision of an 'ideal' test collection. Technical report, University Computer Laboratory, Cambridge, 1975.

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

S. S. Stevens. On the theory of scales of measurement. *Science*, 103(2684):677–680, 1946.

M. Sun, G. Lebanon, and K. Collins-Thompson. Visualizing differences in web search algorithms using the expected weighted Hoeffding distance. In M. Rappa, P. Jones, J. Freire, and S. Chakrabarti, editors, *Proc. 19th International Conference on World Wide Web*, pages 931–940, Raleigh, North Carolina, USA, Apr. 2010.

J. M. Tague. The pragmatics of information retrieval experimentation. In K. Spärck Jones, editor, *Information Retrieval Experiment*, chapter 5, pages 59–102. Butterworths, 1981.

J. Tague-Sutcliffe. The pragmatics of information retrieval experimentation, revisited. *Information Processing & Management*, 28(4):467–490, 1992.

J. Tague-Sutcliffe and J. Blustein. A statistical analysis of the TREC-3 data. In D. Harman, editor, *Proc. 3rd Text REtrieval Conference*, pages 385–398, Gaithersburg, Maryland, USA, Nov. 1994. NIST Special Publication 500-225.

P. Thomas and D. Hawking. Evaluation by comparing results sets in context. In P. S. Yu, V. Tsotras, E. A. Fox, and B. Liu, editors, *Proc. 15th ACM International Conference on Information and Knowledge Management*, pages 94–101, Arlington, Virginia, USA, Nov. 2006.

S. K. Thompson. *Sampling*. John Wiley & Sons, New York, 2nd edition, 2002.

A. Trotman. Learning to rank. *Information Retrieval*, 8(3):359–381, 2005.

A. Turpin and F. Scholer. User performance versus precision measures for simple search tasks. In S. Dumais, E. Efthimiadis, D. Hawking, and K. Järvelin, editors, *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 11–18, Seattle, Washington, USA, Aug. 2006.

A. Turpin, F. Scholer, K. Järvelin, M. Wu, and J. S. Culpepper. Including summaries in system evaluation. In J. Allan, J. Aslam, M. Sanderson, C. Zhai, and J. Zobel, editors, *Proc. 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 508–515, Boston, Massachusetts, USA, July 2009.

E. Voorhees. Overview of TREC 2005. In E. Voorhees and L. P. Buckland, editors, *Proc. 14th Text REtrieval Conference*, pages 1:1–15, Gaithersburg, Maryland, USA, Nov. 2005a. NIST Special Publication 500-266.

E. Voorhees. Overview of TREC 2007. In E. Voorhees and L. P. Buckland, editors, *Proc. 16th Text REtrieval Conference*, pages 1:1–16, Gaithersburg, Maryland, USA, Nov. 2007. NIST Special Publication 500-274.

E. Voorhees. Overview of the TREC 2003 robust retrieval track. In E. Voorhees and L. P. Buckland, editors, *Proc. 12th Text REtrieval Conference*, pages 69–77, Gaithersburg, Maryland, USA, Nov. 2003. NIST Special Publication 500-255.

E. Voorhees. Overview of the TREC 2004 robust retrieval track. In E. Voorhees and L. P. Buckland, editors, *Proc. 13th Text REtrieval Conference*, pages 6:1–10, Gaithersburg, Maryland, USA, Nov. 2004. NIST Special Publication 500-261.

E. Voorhees. Overview of the TREC 2005 robust retrieval track. In E. Voorhees and L. P. Buckland, editors, *Proc. 14th Text REtrieval Conference*, pages 6:1–9, Gaithersburg, Maryland, USA, Nov. 2005b. NIST Special Publication 500-266.

E. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. *Information Processing & Management*, 36(5):697–716, Sept. 2000.

E. Voorhees. The philosophy of information retrieval evaluation. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Proc. 2nd Workshop of the Cross-Lingual Evaluation Forum*, volume 2406 of *Lecture Notes in Computer Science*, pages 355–370, Darmstadt, Germany, Sept. 2002. Springer.

E. Voorhees. Evaluation by highly relevant documents. In W. B. Croft, D. J. Harper, D. H. Kraft, and J. Zobel, editors, *Proc. 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 74–82, New Orleans, Louisiana, USA, Sept. 2001.

E. Voorhees. Question answering in TREC. In Voorhees and Harman (2005a), chapter 10.

E. Voorhees. On test collections for adaptive information retrieval. *Information Processing & Management*, 44(6):1879–1885, Nov. 2008.

E. Voorhees. I come not to bury Cranfield, but to praise it. In B. Kules, D. Tunkelang, and R. White, editors, *Proc. 3rd Annual Workshop on Human-Computer Interaction and Information Retrieval*, pages 1–4, Washington, DC, USA, Oct. 2009a.

E. Voorhees. Topic set size redux. In J. Allan, J. Aslam, M. Sanderson, C. Zhai, and J. Zobel, editors, *Proc. 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 806–807, Boston, Massachusetts, USA, July 2009b.

E. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proc. 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–323, Melbourne, Australia, Aug. 1998.

E. Voorhees and C. Buckley. The effect of topic set size on retrieval experiment error. In K. Järvelin, M. Beaulieu, R. Baeza-Yates, and S.-H. Myaeng, editors, *Proc. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 316–323, Tampere, Finland, Aug. 2002.

E. Voorhees and J. Garofolo. Retrieving noisy text. In Voorhees and Harman (2005a), chapter 8.

E. Voorhees and D. Harman. Overview of the fifth text REtrieval conference (TREC-5). In E. Voorhees and D. Harman, editors, *Proc. 5th Text REtrieval Conference*, pages 1–28, Gaithersburg, Maryland, USA, Nov. 1996. NIST Special Publication 500-238.

E. Voorhees and D. Harman. Overview of the sixth text REtrieval conference (TREC-6). In E. Voorhees and D. Harman, editors, *Proc. 6th Text REtrieval Conference*, pages 1–24, Gaithersburg, Maryland, USA, Nov. 1997. NIST Special Publication 500-240.

E. Voorhees and D. Harman. Overview of the eighth text REtrieval conference (TREC-8). In E. Voorhees and D. Harman, editors, *Proc. 8th Text REtrieval Conference*, pages 1–24, Gaithersburg, Maryland, USA, Nov. 1999. NIST Special Publication 500-246.

E. Voorhees and D. Harman. Overview of the sixth text REtrieval conference (TREC-6). *Information Processing & Management*, 36(1):3–35, 2000.

E. Voorhees and D. Harman, editors. *TREC: Experiment and Evaluation in Information Retrieval*. The MIT Press, 2005a.

E. Voorhees and D. Harman. The Text REtrieval Conference. In *TREC: Experiment and Evaluation in Information Retrieval* Voorhees and Harman (2005a), chapter 1.

L. Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer, New York, 2004.

W. Webber. Evaluating the effectiveness of keyword search. *IEEE Data Eng. Bull.*, 33 (1):54–59, 2010.

W. Webber and L. A. F. Park. Score adjustment for correction of pooling bias. In J. Allan, J. Aslam, M. Sanderson, C. Zhai, and J. Zobel, editors, *Proc. 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 444–451, Boston, Massachusetts, USA, July 2009.

W. Webber, V. N. Anh, and A. Moffat. The University of Melbourne in the million query track of TREC 2007. In E. Voorhees and L. P. Buckland, editors, *Proc. 16th Text REtrieval Conference*, pages 86:1–5, Gaithersburg, Maryland, USA, Nov. 2007a. NIST Special Publication 500-274.

W. Webber, A. Moffat, and J. Zobel. Score standardization for robust comparison of retrieval systems. In M. Wu, A. Turpin, and A. Spink, editors, *Proc. 12th Australasian Document Computing Symposium*, pages 1–8, Melbourne, Dec. 2007b.

W. Webber, A. Moffat, and J. Zobel. Statistical power in retrieval experimentation. In J. G. Shanahan, S. Amer-Yahia, Y. Zhang, A. Kolcz, A. Chowdhury, and D. Kelly, editors, *Proc. 17th ACM International Conference on Information and Knowledge Management*, pages 571–580, Napa, California, USA, Oct. 2008a.

W. Webber, A. Moffat, and J. Zobel. Score standardization for inter-collection comparison of retrieval systems. In Myaeng et al. (2008), pages 51–58.

W. Webber, A. Moffat, J. Zobel, and T. Sakai. Precision-at-ten considered redundant. In Myaeng et al. (2008), pages 695–696.

W. Webber, A. Moffat, and J. Zobel. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems*, 2010. to appear.

E. B. Wilson, Jr. *An Introduction to Scientific Research*. McGraw-Hill, 1952.

S. Wu and F. Crestani. Methods for ranking information retrieval systems without relevance judgments. In G. B. Lamont, H. Haddad, G. A. Papadopoulos, and B. Panda, editors, *Proc. 2003 ACM Symposium on Applied Computing*, pages 811–816, Melbourne, Florida, USA, Mar. 2003.

Y. Xu, Y. Ishikawa, and J. Guan. Effective top-k keyword search in relational databases considering query semantics. In *Proc. APWeb-WAIM 2009 International Workshops*, volume 5731 of *Lecture Notes in Computer Science*, pages 172–184, Suzhou, China, Apr. 2009.

E. Yilmaz and J. Aslam. Estimating average precision with incomplete and imperfect judgments. In P. S. Yu, V. Tsotras, E. A. Fox, and B. Liu, editors, *Proc. 15th ACM International Conference on Information and Knowledge Management*, pages 102–111, Arlington, Virginia, USA, Nov. 2006.

E. Yilmaz, J. Aslam, and S. Robertson. A new rank correlation coefficient for information retrieval. In Myaeng et al. (2008), pages 587–594.

E. Yilmaz, E. Kanoulas, and J. Aslam. A simple and efficient sampling method for estimating AP and NDCG. In Myaeng et al. (2008), pages 603–610.

C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.

W. Zhang, S. Liu, C. Yu, C. Sun, F. Liu, and M. Weiyi. Recognition and classification of noun phrases in queries for effective retrieval. In M. J. Silvaa, A. A. F. Laender, R. Baeza-Yates, D. L. McGuinness, B. Olstad, Ø. H. Olsen, and A. O. Falcão, editors, *Proc. 16th ACM International Conference on Information and Knowledge Management*, pages 711–720, Lisboa, Portugal, Nov. 2007.

Y. Zhang, L. A. F. Park, and A. Moffat. Click-based evidence for decaying weight distributions in search effectiveness metrics. *Information Retrieval*, 13(1):46–69, Feb. 2010.

J. Zobel. How reliable are the results of large-scale information retrieval experiments? In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proc. 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 307–314, Melbourne, Australia, Aug. 1998.

J. Zobel, A. Moffat, and L. A. F. Park. Against recall: Is it persistence, cardinality, density, coverage, or totality? *SIGIR Forum*, 43(1):3–15, June 2009.