Beyond Measuring Performance: Targeted Meta-Evaluations for Diversity and an Information-Theoretic Framework for the Analysis of Search Engines

A dissertation presented

by

Peter Bernard Golbus

to the Faculty of the Graduate School

of the College of Computer and Information Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Northeastern University

Boston, Massachusetts

June, 2014

NORTHEASTERN UNIVERSITY
GRADUATE SCHOOL OF COMPUTER SCIENCE
Ph.D. THESIS APPROVAL FORM

THESIS TITLE:

AUTHOR:

Ph.D. Thesis Approved to complete all degree requirements for the Ph.D. Degree in Computer Science.

_____	_____
Thesis Advisor	*Date*
_____	_____
Thesis Reader	*Date*
_____	_____
Thesis Reader	*Date*
_____	_____
Thesis Reader	*Date*

GRADUATE SCHOOL APPROVAL:

_____	_____
Director, Graduate School	*Date*

COPY RECEIVED IN GRADUATE SCHOOL OFFICE:

_____	_____
Recipient's Signature	*Date*

Distribution: Once completed, this form should be scanned and attached to the front of the electronic dissertation document (page 1). An electronic version of the document can then be uploaded to the Northeastern University-UMI website.

Dedication

To my advisor, Jay Aslam—for teaching me to be a scientist.

To my labmates, past and present—for playing science with me: Jesse Anderton, Maryam Aziz, Maryam Bashir, Keshi Dai, Matthew Ekstrand-Abueg, Moonyoung Kang, Cheng Li, Pavel Metrikov, Virgil Pavlu, Shahzad Rajput, and Stefan Savev.

To my wife, Alissa, and to my son, Aaron—because some things are important.

Abstract

In this work, we address information retrieval evaluation and the methods and metrics used for such evaluations. We consider the relative lack of understanding in this area to be *the* crucial problem in advancing information retrieval. To that end, we introduce several frameworks for meta-evaluation and describe how their unification with evaluation measures can lead to improvements in assessing the quality of information retrieval systems.

For example, many queries, especially in the context of the web, have multiple interpretations; they are *ambiguous* or *underspecified*. To account for this, much recent research has focused on creating systems that produce *diverse* ranked lists that cover as many interpretations in as few documents as possible. Ideally, measures that evaluate these systems would distinguish between them by how many interpretations they cover and how quickly. Unfortunately, diversity is also a function of the collection over which the system is run and a system's ability to retrieve documents relevant to *any* interpretation. To ensure that we are assessing systems by their diversity, we develop (1) a family of evaluation measures that take into account the diversity of the collection and (2) a meta-evaluation measure that explicitly controls for a system's ability to retrieve relevant documents. We demonstrate experimentally that our new measures can achieve substantial improvements in sensitivity to diversity without reducing discriminative power.

Furthermore, we propose a *probabilistic framework* whose utility encompasses both evaluation and meta-evaluation. This allows us to develop new *information-theoretic* evaluation and meta-evaluation metrics that will, hopefully, be more easy to unify in a fashion similar to our family of diversity measures. We demonstrate that these new metrics are powerful and generalizable, enabling evaluations heretofore not possible.

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

Today we live in a world of effectively infinite information. Each of us carries a device in our pocket that allow us access to the sum total of human knowledge. It has become increasingly less important to *know* facts, and increasingly more crucial to be able to *find* them. The gatekeeper to this vast trove of data is the search engine.

Search engines are a modern manifestation of the field known in Computer Science as Information Retrieval (IR). IR is "a field concerned with the structure, analysis, organization, storage, searching, and retrieval of information [65]." As early as the 1950s, IR researchers have been exploring means of locating specific documents in haystacks of text, often as large as whole encyclopedias [35]. While the scale of information retrieval has changed drastically, the underlying tasks themselves, though certainly more complicated to perform in today's world, are essentially unchanged.

Any IR system must solve the following problems:

1. Document collecting—The first step in building an IR system is choosing the set of documents that your system can be used to search. The modern search engine is built to search the web, and finds its documents by performing a web "crawl." Crawling the web is the practice of utilizing the hyperlinked structure of web pages to discover large portions of the web based on smaller sets of seed pages.

2. Information processing—Every time you locate a new document in your crawl, you must process this information in some fashion. In the context of the web, this involves parsing HTML for text, images and other multimedia content, and meta-data. This data, when properly represented, will be used to determine whether the page contains the content the user is searching for.

3. Indexing—The previous steps generate unimaginably vast amounts of data, which must be compressed and stored. Since the web is ever changing, these steps must be efficient enough that they can be run frequently. Furthermore, modern users of the web are often unwilling to wait to load pages with any noticeable delay. Therefore, this index must provide representations of millions of web pages to the search engine in milliseconds.

4. Query Processing—Initially, for research purposes, IR systems were given detailed narratives describing the specific information the user was searching for. Modern search engines are given two or three keywords. Therefore it is necessary to expand queries with additional terms to aid in search. Search engines are also often able to make use of additional information such as location, search history, *etc.* which can be considered alongside the actual query submitted by the user.

5. Document Ranking—At its heart, an IR system is a function that takes representations of documents and representations of "information needs" and returns a number that represents how likely the document is to satisfy that need. This number is then used to present a ranked list of documents to the user. These functions can either be formal, empirically and theoretically derived "retrieval models," or can be learned using so-called "learning to rank" algorithms. Additionally, search engines now provide not just ranked lists of web pages, but entire "search engine results pages" containing, for example, images, news, maps, suggested queries, *etc.*, as well as advertising.

6. Result Evaluation—In order to improve the performance of an IR system, it is necessary to have some means of assessing the quality of the system as it currently exists. Otherwise, while you can certainly alter the system to your heart's content, there is no way for you to know whether your changes were beneficial. The core of this evaluation is some means of utilizing humans to find some sample of the true set of "relevant" information. Historically, this has been provided by hiring trained assessors or performing user studies. The modern search engine has enough people interacting with it that they are also able to make use of observed user behavior, such as observing which documents are clicked on, whether users reformulate their queries and continue searching, *etc.*

Our work is focused on the result evaluation phase of this process. Currently, we consider this to be *the* bottleneck in improving the modern search engine. To a

physicist, defining temperature as that which can be measured by a thermometer is a reasonable and often useful approximation. Under no circumstances would one consider a search engine to be that which can be measured by an evaluation metric such as nDCG [45]. Since the introduction of the World Wide Web, there have been perhaps a half-dozen new evaluation measures specifically aimed at ad hoc retrieval (see Section 2.1). Ad hoc retrieval has become a niche problem that relatively few researchers are still trying to solve, and yet we are still not confident in our ability to recognize high quality ad hoc systems when we see them. The recent past has seen an explosion of new retrieval tasks such as Web search, Diversification, Topic Distillation, Knowledge Acquisition, Temporal Summarization, Legal search, Enterprise search, Microblog search, and on and on. The vast majority of IR research-hours are spent trying to improve systems that solve these problems. It is our contention that without an accurate thermometer, those researchers may have already found adequate solutions, or may even be making their systems worse, and would never know.

To demonstrate the ways in which our thermometers may be leading us in the wrong direction, consider the Diversity task which TREC ran from 2009 to 2012 [28–31], in which systems try to provide ranked lists that are both *diverse*—in that they cover as many search intents as possible—and *novel*—in that each new document should provide previously unseen information [33]. Evaluating these systems requires effectiveness measures that appropriately reward diversity in the result list. Many such measures have been proposed [1, 24, 27, 32, 33, 62–64, 93], all highly correlated and all assumed to measure system's success at diversification. Therefore, researchers strive to build systems that perform well according to these measures, believing that this will lead to systems that satisfy users who search for ambiguous and underspecified queries. The primary meta-evaluation used to demonstrate the effectiveness of these measures is *discriminative power* [60], which assesses how sensitive measures are to changes in ranked lists. Discriminative power is a valuable tool, yet it was originally designed for ad hoc performance measures. While these measures are highly sensitive as measured by discriminative power, discriminative power alone does not tell us if the sensitivity is due to diversity. However, according to Santos *et al.* and as we argue in Sections 3.1 and 5.2, these measures are largely dominated by ad hoc performance.

The goal of this work is to: 1) prove that by using targeted meta-evaluations, one can define the true nature of the problem that is being attacked, 2) prove that evaluation can be unified with meta-evaluation to better identify high-quality systems, and 3) provide a new, powerful, highly interpretable information-theoretic

toolkit that can be used for evaluation and meta-evaluation within a single unified framework. Our framework for evaluation is based on the observation that relevance judgments can be interpreted as a preference between those documents with different relevance grades. This implies that relevance judgments can be treated as a retrieval system, and that evaluation can be considered as the "rank" correlation between systems and relevance judgments. To this end, we develop a probabilistic framework for rank correlation based on the expectation of random variables, which we demonstrate can also be used to compute existing evaluation metrics. However, the true value of our framework lies in its extension to new information-theoretic evaluation tools.

Outline

In Chapter 2, we give a brief history of how academia has, without the access to user behavior enjoyed by major search engines, evaluated information retrieval systems through the use of *test collections*. In Section 2.1, we describe the *ad hoc* testing regime that was in place at the dawn of the Internet era, and how the evaluation measures used were adapted to the presence of the web. In Section 2.2, we describe the *diversity* task, which grew out of the ad hoc task, and was specifically designed to allow researchers to address problems inherent to searching the web. We focus on how these diversity systems are evaluated. In Section 2.3 we describe a few of the questions that were posed by those researching test collection construction and how those answers relate to our work. In Section 2.3.1 we describe the problems of *stability* and *reusability* caused by the size of document collections and cost of obtaining quality relevance assessments. In Section 2.3.2, we describe the means by which IR researchers compare ranked lists, both of documents and of systems. Finally, in Section 2.3.3, we describe the meta-evaluation measures that have been used to evaluate evaluation measures themselves.

In Chapter 3, we demonstrate several new meta-evaluation measures for the diversity task described in Section 2.2. In Section 3.1, we show that the current methodology is dominated by ad hoc performance and introduce *document selection sensitivity*, a new measure of the sensitivity of an evaluation measure to diversity that controls for this. Another factor that influences diversity evaluation is the amount of intrinsic diversity in the collection (Section 3.2). In Section 3.2.1, we briefly describe the related problem of Query Difficulty Prediction [17] before describing our notion of *diversity difficulty*, which quantifies the amount of diversity present in a collection at the topic level. In Section 3.2.2, we introduce an analogous notion, *subtopic miss rate*, that quantifies a collection's intrinsic diversity at the subtopic level. The chapter concludes with a brief summary of results (Section 3.3).

In Chapter 4, we show how to incorporate these meta-evaluation measures into the existing framework of diversity evaluation. We introduce a new family of measures in Section 4.1 that have as much discriminative power as existing measures (Section 4.2). We show that, by amplifying the impact of difficult queries that on which only high quality systems should have good performance, our measures prefer different systems than do existing measures (Section 4.3). However, this does not indicate that our measures prefer more diverse systems, as we do now know why it prefers the systems that it does. Therefore, in Section 4.4, we use our new meta-evaluation measure to show that our family of measures is more sensitive to document selection and ordering than existing measures. We conclude with a brief summary in Section 4.5.

In Chapter 5, we introduce our probabilist framework for evaluation and applications. In Section 5.1, we motivate and define our framework (Section 5.1.1). In Section 5.1.2, we show how it can be used to compute traditional ad hoc evaluation measures and in Section 5.1.3 we demonstrate that our probabilistic framework can be interpreted information theoretically. This leads to our first application in Section 5.2, *information* τ, a measure of *conditional rank correlation* and a useful meta-evaluation tool for demonstrating that a correlation between rankings is not causal. In Section 5.3, we demonstrate how our framework can be manipulated to produce an evaluation measure, which we call *Relevance Information Correlation* (Section 5.3.1). In Section 5.3.2, we show that these evaluation measures are consistent with existing measures currently in use. However, we show that the information-theoretic nature of our evaluation framework allows for novel uses (Section 5.4), such as considering multiple ranked lists simultaneously, effectively computing an upper bound on metasearch. Section 5.5 contains a brief summary.

In Chapter 6, we focus on one particular application of our framework, *information difference*, which can be used to measure the similarity between IR systems based on their *behavior* rather than their *performance* (Section 6.1). In Section 6.2, we use this technology to measure the relative impact of choice of retrieval models versus retrieval model parameter tuning. We describe the models we will compare in Section 6.2.1. In Section 6.2.2, we present our experimental results. We also demonstrate that information difference can be used to select candidate systems to be used in metasearch (Section 6.3), significantly outperforming selecting systems at random. In Section 6.3.1, we describe our methodology for selecting systems in term of Facilities Location Analysis [42] and present our experimental results in Section 6.3.2. We briefly summarize our results in Section 6.4.

We present concluding remarks and suggestions for future work in Chapter 7.

We present additional figures in Appendix A.

Chapter 2

A (Brief) History of Evaluation and Meta-Evaluation

In this chapter we provide a brief history of evaluation and meta-evaluation. In Section 2.1, we describe the TREC ad hoc test collection paradigm, a foundational evaluation paradigm that was in place at the dawn of the Internet era and continues to shape our understanding of search engine evaluation in subtle ways. We describe our framework for understanding this evaluation paradigm in Chapters 5 and 6. In Section 2.2, we describe the diversity task, a refinement of this evaluation paradigm specifically designed to improve the experience of users of modern, commercial search engines. We demonstrate improvements to diversity evaluation created by leveraging meta-evaluation measures in Chapters 3 and 4. In Section 2.3, we describe a few of the meta-evaluation challenges that researches face, especially as they relate to our work.

2.1 TREC and the Ad Hoc Test Collection

In order to evaluate a search system, it is necessary to define a *search task* that the system will be asked to perform and to provide a *test collection*: a corpus of documents and a set of searches to perform, as well as a set of *relevance assessments*—for each search, a subset of those documents which are considered "relevant" by some definition defined by that task. For over 20 years, one of the main sources of these tasks and collections used in academia has been the annual, NIST-sponsored Text REtrieval Conference (TREC). One of the early tasks used by TREC is the ad hoc retrieval task. In this task, trained assessors are provided detailed statements of the information need underlying each search query, denoted as topics (see Figure 2.1 for an example). Assessors are tasked with marking documents as *relevant* if "any

```
<top>

<num> Number: 405
<title> cosmic events

<desc> Description:
What unexpected or unexplained cosmic events or
celestial phenomena, such as radiation and
supernova outbursts or new comets, have been detected?

<narr> Narrative:
New theories or new interpretations concerning
known celestial objects made as a result of new
technology are not relevant.

</top>
```

Figure 2.1: Example ad hoc topic from TREC 8.

piece of it is relevant (regardless of how small the piece in in relation to the rest of the document)." Assessors provide binary relevance judgments, *i.e.* they simply mark documents as relevant or non-relevant with no regard for quality or degree of relevance. The set of all relevance judgments are known as QRELs. Participants are asked to rank up to 1,000 documents for each topic. TREC in general, and the ad hoc task in particular, have hugely influenced IR research [66].

Systems are evaluated for the ad hoc task based on their trade-off between *precision*, the percentage of retrieved documents that are relevant, and *recall*, the percentage of relevant documents that are retrieved. For a given topic, Let $g_i \in \{0, 1\}$ be the relevance grade of the document at rank i, and let R be the number of relevant documents in the collection. At rank k,

$$\text{precision@}k = \frac{\sum_{i=1}^{k} g_i}{k} \tag{2.1}$$

$$\text{recall@}k = \frac{\sum_{i=1}^{k} g_i}{R} \tag{2.2}$$

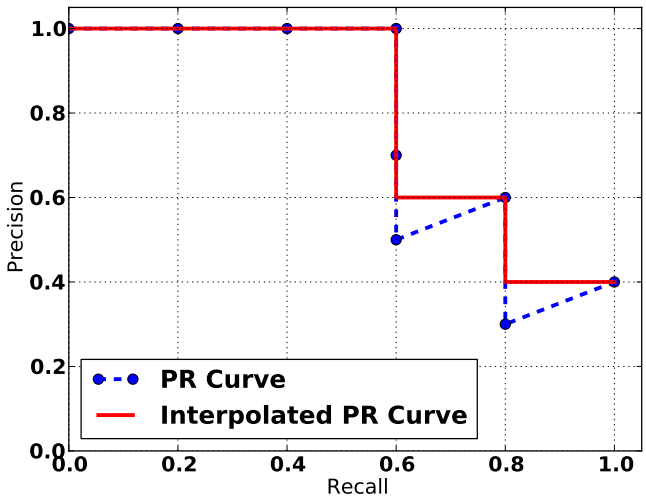


Figure 2.2: Precision-recall curves are a visualization of their tradeoff. These curves are typically interpolated so as to be non-increasing, ensuring that precision is defined for every recall.

It is trivial to design a system that is perfect with respect to one of these by simply adjusting the number of documents that are retrieved. A system that retrieves a single document is highly likely to have perfect precision, whereas a system that retrieves all documents is guaranteed to have perfect recall. The trade-off between the two can be measured by *average precision*, which is the average of the observed precision at the rank of each relevant document if non-retrieved relevant documents are assumed to appear at rank ∞.

$$AP = \frac{\sum_{i=1}^{\infty} g_i \times \text{precision@}i}{R} \tag{2.3}$$

Average precision can also be interpreted as the area under the precision-recall curve (see Figure 2.2).

Average precision does not include information about document quality and degrees of relevance, and is an inherently recall-oriented measure. It is therefore not suitable for evaluating commercial web search engines. With the growth of the World Wide Web, test collections began to include graded, non-binary relevance judgments, e.g. $G = \{\text{non-relevant, relevant, highly relevant}\}$ or $G = \{0, \ldots, 4\}$. To make use of these graded assessments, Järvelin and Kekäläinen developed *normalized discounted cumulative gain* (nDCG) [45]. nDCG also has the advantage that it can be evaluated at arbitrary ranks, and can therefore be used for precision-oriented

tasks such as web search.

Unlike average precision, which has a technical interpretation, nDCG can be best understood in terms of a model of a hypothetical user. In this model, a user will read the first k documents in a ranked list, deriving utility from each document. The amount of utility is proportional to the document's relevance grade and inversely proportional to the rank at which the document is encountered. We first define discounted cumulative gain (DCG).

$$DCG@k = \sum_{i=1}^{k} \frac{2^{g_i} - 1}{\log_2(i+1)} \tag{2.4}$$

Since the range of DCG will vary from topic to topic, it is necessary to normalize these scores so that an average can be computed. Normalization is performed with regard to an ideal ranked list. If $DCG'@k$ is the maximum possible DCG of any ranked list of documents in the collection then

$$nDCG@k = \frac{DCG@k}{DCG'@k} \tag{2.5}$$

However, one does not always know how many documents are relevant at each level, and therefore the ideal list used for normalization is only an approximation. Moffat and Zobel [50] introduced a measure, *rank-biased precision* (RBP), that addresses this issue. In RBP, the probability that a user will read the document at rank k is drawn from a geometric distribution, whose parameter, $\beta \in [0, 1)$, models the user's persistence. Given a utility function $u\colon G \to [0, 1]$, commonly defined as

$$u(g) = \frac{2^g - 1}{2^d} \tag{2.6}$$

where d is the maximum possible relevance grade, RBP is defined as the expected utility of a user who browses according to this model.

$$RBP = (1 - \beta) \sum_{i=1}^{\infty} u(g_i) \times \beta^{i-1} \tag{2.7}$$

Since RBP is guaranteed to be in the range [0,1) for any topic and β, it does not require normalization.

These user models were derived theoretically, and were not validated experimentally. This led Craswell *et al.* [34] to introduce the Cascade model of user behavior. In this model, a user is still assumed to browse documents in order, but the probability that a user will view a particular document is no longer assumed to be independent of the documents that were viewed previously, i.e. a user is not

assumed to stop at a particular rank, or at each rank with some probability. Instead, the user is assumed to stop after finding a relevant document. This implies that if a user reaches rank k, then all of the $k-1$ documents ranked before it were non-relevant. Craswell *et al.* demonstrated empirically that this model corresponds well to observed user behavior in terms of predicting the click-through behavior observed on a commercial search engine.

Chapelle *et al.* [25] developed an evaluation measure, *expected reciprocal rank* (ERR), based on the Cascade model. Let R_i denote the probability that a user will find the document at rank i to be relevant. Then in the Cascade model, which assumes that documents do not interact with one another, the likelihood that a user will terminate his or her search at rank r is

$$P(\text{user stops at rank } r) = R_r \prod_{i=1}^{r-1}(1 - R_i). \tag{2.8}$$

If we interpret the previously defined utility function (Equation 2.6) as the probability that a user will find a document relevant, i.e. $R_i = u(g_i)$, then we can computed the expected reciprocal rank at which a user will terminate his or her search as

$$ERR = \sum_{r=1}^{\infty} \frac{1}{r} R_r \prod_{i=1}^{r-1}(1 - R_i). \tag{2.9}$$

This whole evaluation paradigm is predicated upon the Probabilistic Ranking Principle (PRP) [58], which dictates that documents should be ranked by their probability of relevance to the user's intent, with the simplifying assumption of independent document relevance. Chen and Kargar [26] showed that the PRP is actually sub-optimal if the user is interested in a limited number of relevant documents, rather than all relevant documents. Also contrary to the PRP, Carbonell and Goldstein's Maximal Marginal Relevance method [16], which iteratively selects documents that are most similar to the query and least similar to the documents previously shown to the user, was shown to be more effective at tasks such as automatic summarization. Zhai *et al.* [93] explicitly rejected the assumption of independent relevance to define the *subtopic retrieval* problem. These insights led to the creation of what has come to be known as the diversity task.

2.2 The Diversity Task

Information retrieval research traditionally assumes that a given query can be associated with a single underlying user intent, or information need. In reality—

especially in the context of Web search—users with very different intents may enter identical queries. In many cases, queries may be *ambiguous*, with multiple unrelated interpretations [27]. For example, a user entering the query "zeppelin" may be interested in either the band or the type of airship. Even when a query is unambiguous it may be *underspecified*, and may not precisely express the user's information need. For example, a user entering the query "led zepplin" may be seeking a discography, biographies of the members, and/or news about a possible reunion. When a query gives rise to many possible interpretations, the ideal ranked result list should be both *diverse*—it should cover as many search intents as possible—and *novel*—each new document should provide previously unseen information [33]. Producing such a ranked list has come to be known as the diversity task.

From 2009 to 2012, the TREC Web Track [28–31] has included a diversity task alongside the traditional ad-hoc task. The track organizers constructed 50 topics for each collection. For each topic, the organizers created a number of subtopics corresponding to example search intents by extracting information from the logs of a commercial search engine. Each subtopic is given a type, "navigational" or "informational," denoting whether the user is interested in finding a specific web page or any web page with the correct content. Figure 2.3 presents two topics from the 2011 collection and their subtopics. Topic 114 is "faceted," i.e. "underspecified" in the same sense as our "led zeppelin" example; we know what an adobe indian house is, but we do not know which facet of this broad topic the user is interested in. Topic 140 is "ambiguous," similar to our "zeppelin" example. There are many high schools named East Ridge; without additional information, there is no way of knowing which one the user intended.

Track participants were given the queries (and not the subtopics) associated with each topic. Through 2011, systems were run on the ClueWeb09 corpus,[1] a general web crawl from 2009 containing approximately 1 billion documents. The submissions were pooled (see Section 2.3) and judged to a depth of 20 (2009, 2010) or 25 (2011). Hired assessors made binary relevance judgments with respect to each subtopic.[2] All of our evaluations will be performed using these relevance assessments. We focus on the 2010 and 2011 collections, since participants in those years had time to work with the 2009 data to better understand how to diversify runs.

[1] `lemurproject.org/clueweb09/`

[2] In 2011, the judgments were actually graded. For this work, we consider any document with relevance grade greater than zero to be relevant and any document with a relevance grade of zero or marked as spam to be non-relevant.

```
<topic number="114" type="faceted">
  <query>adobe indian houses</query>
  <description>
    How does one build an adobe house?
  </description>
  <subtopic number="1" type="inf">
    How does one build an adobe house?
  </subtopic>
  <subtopic number="2" type="inf">
    information about Indian tribes that used adobe houses
  </subtopic>
  <subtopic number="3" type="nav">
    I'd like to order books or videos/CDs about how to construct
    adobe buildings.
  </subtopic>
</topic>

<topic number="140" type="ambiguous">
  <query>east ridge high school</query>
  <description>
    demographics of East Ridge High School in Lick Creek, Kentucky
  </description>
  <subtopic number="1" type="inf">
    demographics of East Ridge High School in Lick Creek, Kentucky
  </subtopic>
  <subtopic number="2" type="nav">
    home page for East Ridge High School in Chattanooga, Tennessee
  </subtopic>
  <subtopic number="3" type="inf">
    information about the sports program at East Ridge High School
    in Clermont, Florida
  </subtopic>
  <subtopic number="4" type="inf">
    description of the sports facilities at East Ridge High School
    in Woodbury, MN
  </subtopic>
</topic>
```

Figure 2.3: Examples of TREC 2011 Web track diversity task topics.

1. $\mathbf{ERR}(k) = \frac{1}{k}$
2. $\mathbf{DCG}(k) = \frac{1}{\log(k+1)}$
3. $\mathbf{RBP}(k) = \frac{1}{\beta}^{k-1}$

Table 2.1: Discount vectors used in evaluation measures.

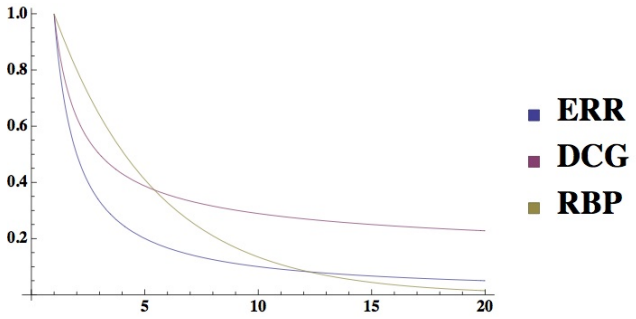


Figure 2.4: Discount vectors used in evaluation measures.

Following Zhang *et al.* [94], we note that most measures can easily be described as the cross-product of a gain vector with a discount vector, normalized in some fashion. Following Clarke *et al.* [32], we begin by highlighting three particular functions for producing discount vectors based on ad hoc performance measures (see Table 2.1 and Figure 2.4).

The first diversity evaluation measures are due to Zhai *et al.* [93]. One of these measures, S-Recall, is still in use. S-Recall does not fit into cross-product framework. Let X be the number of subtopics covered by at least one document at or before rank k. Assume a topic has M subtopics. Then

$$\text{S-Recall@}k = \frac{X}{M}. \tag{2.10}$$

The next measures we choose to highlight are the Cascade measures [27,32,33]. Clarke *et al.* [33] modified nDCG by using the diminishing returns of cascade measures to model the user's tolerance for redundancy in a ranked list. By penalizing redundancy, α-nDCG rewards both novelty and diversity. Cascade measures use a cascading gain function where a document's gain with respect to some subtopic is

decreased each time the subtopic is encountered. To define the gain function, let I_i^r be an indicator variable representing whether the document at rank r is relevant to subtopic i. Let

$$c_i^k = \sum_{r=1}^{k-1} I_i^r \tag{2.11}$$

represent the number of documents relevant to subtopic i seen prior to rank k. Let g_i^k be the relevance grade, or a function thereof, of the document at rank k with respect to subtopic i. Let w_i represent the probability that a user's intent is subtopic i. Then the gain of the document at rank k is

$$\text{Gain}(k) = \sum_{i=1}^{M} w_i \times g_i^k (1-\alpha)^{c_i^k}, \tag{2.12}$$

where α is a parameter in $[0, 1]$ which models the users tolerance for redundancy. The gain can be combined with any discount vector from Table 2.1. Normalization is performed relative to a single ideal ranked list. A drawback of the Cascade measures is that the ideal ranked list required for normalization is NP-hard to compute, and is therefore usually approximated.

Agrawal *et al.* [1] proposed the IA measure family that computes the weighted average of ad hoc metrics computed separately for each intent. These measures incorporate both degrees of relevance, known as graded relevance, and the likelihood that a user is interested in a particular subtopic, known as intent probability. As an example of intent probability, suppose that more users who enter the query "zeppelin" are interested in the band than the mode of travel. Then systems should receive higher rewards for retrieving documents that refer to the former than the later.

Intent aware measures model diversity by computing the weighted average of an evaluation measure with respect to each subtopic. As an example of an intent aware measure, consider nDCG-IA. Let nDCG_i represent nDCG [45] evaluated with respect to subtopic i. Then the intent aware measure nDCG-IA would be:

$$\text{nDCG-IA@}k = \sum_{i=1}^{M} w_i \times \text{nDCG}_i@k. \tag{2.13}$$

Notice that normalization is computed separately for each subtopic. Each subtopic requires its own ideal ranked list, but these lists can be computed directly.

A drawback of IA measures is that they tend to prefer systems that perform well

on the most likely subtopics over systems that are more diverse [32, 63]. Partially in an attempt to correct this problem, Sakai *et al.* [63] introduced the D# family of measures [62–64]. The authors begin by returning to the Probabilistic Ranking Principle. If we assume that:

1. intents are mutually exclusive, *i.e.* $\sum_{i=1}^{M} w_i = 1$, and
2. the binary probability of relevance is proportional to the relevance grade,

then the gain of a document, which the authors refer to as *global gain*, is

$$\text{Global Gain}(k) = \sum_{i=1}^{M} w_i g_i^k. \tag{2.14}$$

Global gains are computed for each subtopic and normalized with regard to a single ideal ranked list. Unlike Cascade measures, this ideal ranked list can be found by a simple greedy algorithm. Measures using global gain are called D-measures. To increase the correlation with subtopic coverage, D#-measures are D-measures combined linearly with S-Recall. The mixture is controlled by a parameter $\lambda \in [0, 1]$, with $\lambda = 1$ being equivalent to pure S-Recall.

Additionally, noting that D# measures seemed to perform differently on subtopics depending on their taxonomy, i.e. *navigational* vs. *informational* [12], Sakai [62] developed additional measures in the style of D# that explicitly take subtopic taxonomy into account. When the intent is informational, one of these, the P+Q# measures, uses what we would call a #-IA measure in the sense of Section 4.1.

2.3 Meta-Evaluations

In the previous sections, we have described how test collections are *used* for evaluation. We have not discussed how test collections are *created*, nor the meta-evaluations used in guiding this process. In this section, we will discuss three of the main problems that test collection creation research must address:

1. minimizing the impact of missing relevance judgments,
2. comparing ranked lists, be they lists of documents or lists of systems, and
3. assessing the relative merits of evaluation metrics.

Our work was inspired by applying the probabilistic approach of computing evaluation measures in expectation, first employed in addressing the problem of missing judgments, to the problem of comparing ranked lists. The goal of our work is to introduce a framework for assessing evaluation metrics.

2.3.1 Missing Relevance Judgments

In Section 2.1 we discussed how human assessors separate documents into two classes for each query: relevant and non-relevant. However, given the size of even a modest test collection, there is a third, far, far larger class: documents that assessors do not read. To limit the impact of these *unjudged* documents, test collections are usually created via *pooling* [77]. In depth pooling, a number of retrieval systems are selected and run over the corpus for each topic. The union of the top k results from each system are then presented to assessors for judging. It is assumed that this forms a reasonable sample that will contain a large fraction of the relevant documents present in the corpus. In this way, it is reasonable to consider any unjudged document as non-relevant; if it were relevant, it would have been included in the pool.

This assumption, that pooling leads to effectively complete judgments, was challenged as early as 1998. Zobel [97] estimated that fixed-depth pooling strategies at relatively large depths such as rank 100 can lead to as few as 50 to 70% of the relevant documents being judged. However, Zobel concluded that while these evaluations vastly over-estimated recall, they were not biased and therefore systems are ranked correctly. However, Buckley and Voorhees [14] showed that TREC style evaluation is not robust to gross violations of the completeness assumption, *i.e.* that system ranking can change drastically as the number of relevance judgments are artificially reduced. To combat this, they introduced the BPref measure, so-called because "it uses binary relevance judgments to define the preference relation." The definition of BPref, refined by Soboroff [74], is:

$$BPref = \frac{1}{R} \sum_{r} \left(1 - \frac{|n \text{ ranked higher than } r|}{\min(R, N)} \right) \tag{2.15}$$

where N is the number of judged non-relevant documents, r is the set of retrieved, relevant documents, and n is the set of retrieved, non-relevant documents. This measure was found to be far more robust to drastically reduced judgment sets as non-judged documents are not used in computing evaluation scores, and was briefly in very wide use [66].

The TREC Million Query Track [22] was created to explore the boundaries of accurate evaluation with minimal assessment. The two main approaches to evaluation were the so-called "MTC" and "statAP" paradigms. The minimal test collection (MTC) algorithm [21] is a greedy, on-line approach to selecting the set of documents that are most useful for accurately ranking systems. This is very similar to the approach of Moffet *et al.* [50] that targets for judgment the documents that

1. Pick a random relevant document,
2. Pick a random document ranked at or above the rank of the document selected in step 1.
3. Output 1 if the document from step 2 is relevant, otherwise output 0.

Table 2.2: A random experiment whose expected value is equal to average precision.

are most useful for determining the best systems. statAP [7] is the culmination of a research project [8, 87, 88, 90] in using various sampling methodologies to create unbiased, minimum-variance estimates of average precision. We describe an intermediate result, *inferred* AP (infAP), due to its popularity, ease of explanation, and similarity to our probabilistic framework.

Yilmaz and Aslam [87] begin with the observation that average precision can be seen as the expectation of the random experiment described in Table 2.2. For a given relevant document, in expectation, steps two and three compute precision at the rank of the chosen document. Step one averages this in the manner of average precision. To accurately estimate average precision, for each system Yilmaz and Aslam [90] split the unjudged documents into two sets based on whether the documents would have contributed to the test collection's pool. If an unjudged document would not have contributed to the pool, than it is simply deemed non-relevant. Unjudged documents that would have contributed to the pool are considered relevant with a probability proportional to the ratio of retrieved, judged documents that were considered relevant to the number of retrieved documents that were judged. If we assume that systems are pooled at depth 100, then at rank k

$$infAP(k) = \frac{1}{R} \sum_{r} \left[\frac{1}{k} + \frac{k-1}{k} \left(\frac{|d100|}{k-1} \cdot \frac{|rel| + \epsilon}{|rel| + |nonrel| + 2\epsilon} \right) \right] \tag{2.16}$$

where $|d100|$ is the number of judged documents found above rank k plus the number of unjudged documents above rank k that would have contributed to the document pool; $|rel|$ is the number of documents above rank k that are judged relevant; $|nonrel|$ is the number of documents above rank k that are judged non-relevant; and ϵ is a smoothing constant. Notice that the value of infAP given complete judgments, as well as the expected value of infAP if we assume that documents are selected for judging at random, is equal to AP. Experimentally, infAP was found to

be substantially more stable than BPref.

One drawback to the approaches used in the Million Query Track is that they are not *reusable*, in that they are biased estimators of the effectiveness of new systems that did not contribute to the pool. This is due to the on-line, targeted nature of the judgments used by the MTC algorithm and the additional information statAP requires to compute scaling factors necessary for the creation of an unbiased estimate. Additional work since then to combat this problem has been to, for example, use machine learning techniques trained on relevance judgments [15], or to match relevant pieces of information with portions of documents [3, 56, 57], to algorithmically assess unjudged documents. Also, the recent emergence of crowdsourcing platforms such as Amazon's Mechanical Turk[3] has led to the exploration of using inexpensive crowdworkers either in place of, or in addition to, trained assessors to judge large numbers of documents at the same or lesser cost than traditional methods. For example, see the TREC Crowdsourcing task which ran in 2012 and 2013 [72, 73].

2.3.2 Comparing Ranked Lists

Common rank correlation measures such as Kendall's τ are not ideally suited for comparing the output of search engines since they treat all objects equally independent of rank. Multiple solutions to this have been proposed. We focus on two, Kumar and Vassilvitskii [48] and Yilmaz et al. [89], that demonstrate how researchers have extended Kendall's τ to behave more like IR evaluation measires, nDCG and AP respectively, in the past.

Kumar and Vassilvitskii call their measure *generalized* Kendall's τ (K^*). Generalized Kendall's τ encodes positional information by modeling the cost of swapping adjacent documents, denoted δ. In traditional Kendall's τ, δ is uniform independent of rank. Kumar and Vassilvitskii propose several δs; we focus on one based on nDCG. Let δ_r denote the cost of swapping the document at rank r with the document at rank $r-1$. If n objects are ranked, then

$$\delta_r = \frac{1}{\log(r)} + \frac{1}{\log(r+1)}, \tag{2.17}$$

which is defined for $2 \leq r \leq n$. Let σ and σ^* be two rankings. Element i's displacement weight $\bar{p}_i(\sigma, \sigma^*)$ is given by the average cost incurred in moving from rank

[3] `www.mturk.com`

1. Pick a random document from σ ranked after the first document.
2. Pick a random document from σ ranked above the document selected in step 1.
3. Output 1 if these two documents are in the same relative order in σ^*, otherwise output 0.

Table 2.3: A random experiment for comparing ranked lists.

$\sigma(i)$ to rank $\sigma^*(i)$ in terms of adjacent swaps. If $p_r = \sum_{2}^{r} \delta_r$, then

$$\bar{p}_i(\sigma, \sigma^*) = \begin{cases} 1 & \text{if } \sigma(i) = \sigma^*(i) \\ \frac{p_{\sigma(i)} - p_{\sigma^*(i)}}{\sigma(i) - \sigma^*(i)} & \text{otherwise.} \end{cases} \tag{2.18}$$

If $I[\sigma(i) > \sigma(j)]$ is an indicator variable, then the K^* distance is given by

$$K^*(\sigma, \sigma^*) = \sum_{\sigma^*(i) < \sigma^*(j)} \bar{p}_i(\sigma, \sigma^*) \bar{p}_j(\sigma, \sigma^*) I[\sigma(i) > \sigma(j)] \tag{2.19}$$

Yilmaz *et al.* [89] define a version of Kendall's τ based on average precision, τ_{ap}, in terms of the random experiment whose expectation can be used as a definition for AP. Their random experiment, described in Table 2.3, is designed to ask the analogous question about the correlation between ranked lists. Compare this experiment to the one that is used to compute AP (see Table 2.2). The expectation of this random experiment, p' can be computed as

$$p' = \frac{1}{n-1} \sum_{i=2}^{n} \frac{C(i)}{i-1}, \tag{2.20}$$

where $C(i)$ is the number of items in σ^* that are ranked above the document at rank i in σ. This number is then normalized to fall within the interval $[-1, 1]$.

$$\tau_{ap}(\sigma, \sigma^*) = p' - (1 - p') = 2p' - 1 = \frac{2}{n-1} \sum_{i=2}^{n} \left(\frac{C(i)}{i-1} \right) - 1. \tag{2.21}$$

Note that unlike K^*, $\tau_{ap}(\sigma, \sigma^*) \neq \tau_{ap}(\sigma^*, \sigma)$, and therefore one usually reports the

average of the two, *i.e.*

$$\tau'_{ap} = \frac{\tau_{ap}(\sigma, \sigma^*) + \tau_{ap}(\sigma^*, \sigma)}{2}. \tag{2.22}$$

Both of these measures, K^* and τ_{ap}, are undefined when documents do not appear in both lists; in practice, all such documents are simply ignored. However, Webber et al. [86] observe that in information retrieval, ranked lists are *incomplete*, i.e. not all documents are ranked. Therefore, they introduced Rank-Biased Overlap (RBO), an adaptation of RBP (Equation 2.7) to the comparison of ranked lists. The key insight of RBO is that a ranked list can be considered to be a sequence of sets indexed by rank. At any given rank, the set intersection between two ranked lists is defined even if the two ranked lists are not over the same set of objects. In this way, two ranked lists can be compared by the average size of their set intersection at progressively deeper ranks. To make the comparison appropriately top-heavy, they weight their average according to the persistence based user-model of RBP. Let σ_k represent the object appearing in list σ at rank k. Then

$$RBO(\sigma, \sigma^*) = (1 - \beta) \sum_{i=1}^{\infty} \beta^{k-1} |\{\sigma_1, \ldots, \sigma_k\} \cap \{\sigma_1^*, \ldots, \sigma_k^*\}|. \tag{2.23}$$

Note that RBO$(\sigma, \sigma^*) \in [0, 1)$ for all $\beta \in [0, 1)$.

Rank correlation measures can also be used to evaluate ranked lists based on their distance from a reference list. This is often done by aggregating preferences on subsets of objects, as in Voting Theory. For example, Arguello et al. [4] showed that user's preferences between ranked lists with multiple *verticals*, e.g. image search, news search, items for sale, etc., were correlated with the lists' similarity to a reference list built from preferences on individual verticals using the Shulze voting method [69]. However, since many preferences are missing and not all preferences are transitive, building a reference list is an instance of the feedback arc set problem, the decision version of which was one of Karp's original 21 NP-complete problems [46]. Building these lists is a very active area of research, e.g. Chen et al. [26], Volkovs et al. [82], and many others.

2.3.3 Evaluating Evaluation Metrics

This is the most crucial question in the understanding of the modern information retrieval test collection paradigm: how do we interpret evaluation scores and how should we produce them. Unfortunately, this is an incredibly difficult question to even formulate, let alone answer.

One of the early TREC-era attempts to address this question was to quantify measure *stability*: could measurements of system quality on one test collection be used to predict measurements of system quality on another? Zobel [97] simulated this test by splitting test collections into two sets and comparing the rankings of systems induced by evaluation measures on the two sets. Zobel then counted the proportion of systems that had swapped orders between one set and the other. Voorhees and Buckley [85] applied this *swap method* to additional test collections. In another experiment, Buckley and Voorhees [13], measured the consistency of induced rankings across test collections as the set of topics are reformulated into alternate queries.

When of the most commonly used meta-evaluation techniques appearing in the literature is discriminative power [60]. For a given evaluation measure, discriminative power quantifies how *sensitive* the measure is, *i.e.* how often one would expect a measure to be able to distinguish between different systems based on their performance. This is achieved via a paired bootstrap hyphothesis test using a Studentised test statistic [38, 60].

The following discussion, notation, and examples are due to Sakai [60]. Let Q be the set of topics in the test collection. Let $|Q| = n$. Let $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ be the per-topic evaluation scores of two systems X and Y according to evaluation metric M. Typically systems are compared by their *sample means*, *e.g.* $\bar{x} = \sum_{i=1}^{n} \frac{x_i}{n}$. However, what we truly wish to compare is the *population means* for X and Y, μ_X and μ_Y, based on the population of topics, P, of which Q is assumed to be a uniformly random sample. We regard X and Y as *paired* data, and let $\mathbf{z} = (z_1, \ldots, z_n)$ where $z_i = x_i - y_i$ and $\mu = \mu_X - \mu_Y$. In this way, we have replaced the two-sample question of whether $\mu_X = \mu_Y$ with the one-sample question of whether $\mu = 0$. We wish to perform a two-tailed test of the following hypothesis

$$\begin{aligned} H_1 &: \mu \neq 0 \\ &vs \\ H_\emptyset &: \mu = 0. \end{aligned} \tag{2.24}$$

under the assumption that $\mathbf{z}$ is an independent and identically distributed sample drawn from some unknown probability distribution.

To conduct our hypothesis test, we use a *Studentised test statistic*, t

$$t(\mathbf{z}) = \frac{\bar{z}}{s/\sqrt{n}} \tag{2.25}$$

Algorithm 2.1 Algorithm for creating Bootstrap samples $\mathbf{w}^{*^b} = (w_1^{*^b}, \ldots w_n^{*^b})$.

1: **for** $b = 1, \ldots, B$ **do**
2: Let Q^{*^b} be n topics randomly sampled from Q with replacement
3: **for** $i = 1, \ldots, n$ **do**
4: $q = i^{\text{th}}$ topic from Q^{*^b}
5: $w_i^{*^b} =$ observed value in $\mathbf{w}$ for topic q

Algorithm 2.2 Algorithm for estimating the achieved significance level (ASL)

1: $count = 0$
2: **for** $b = 1, \ldots, B$ **do**
3: $t(\mathbf{w}^{*^b}) = \frac{\bar{w}^{*^b}}{s^{*^b}/\sqrt{n}}$
4: **if** $|t(\mathbf{w}^{*^b})| \geq |t(\mathbf{z})|$ **then**
5: $count++$
6: $ASL = \frac{count}{B}$

where s is the sample standard deviation of $\mathbf{z}$ given by

$$s = \sqrt{\sum_{i=1}^{n} \frac{(z_i - \bar{z})^2}{n-1}}. \tag{2.26}$$

If we let $\mathbf{w} = (w_1, \ldots, w_n)$ where $w_i = z_i - \bar{z}$, then $\mathbf{w}$ is a random variable drawn from the *null hypothesis distribution*. We let $\mathbf{w}^{*^b}$ denote a *bootstrap sample* of per-topic performance over topics Q^{*^b} generated by sampling with replacement from the set of topics Q. Algorithm 2.1 shows how to obtain B bootstrap samples of topics Q^{*^b} and the corresponding values of $\mathbf{w}^{*^b}$. For example, assume we have 5 topics, $Q = \{1, 2, 3, 4, 5\}$, and that $\mathbf{w} = (0.2, 0.0, 0.1, 0.4, 0.0)$. Suppose that for some trial b, sampling with replacement from Q yields $Q^{*^b} = (1, 3, 1, 2, 5)$. Then $\mathbf{w}^{*^b} = (0.2, 0.1, 0.2, 0.0, 0.0)$.

For each b, let $\bar{w}^{*^b}$ and s^{*^b} denote the sample mean and sample standard deviation of $\mathbf{w}^{*^b}$. Algorithm 2.2 shows how to compute the achieved significance level (ASL) using $\mathbf{w}^{*^b}$. If $ASL < \alpha$, then we conclude that the observed difference would be sufficiently rare under the null hypothesis, that $\mu_X = \mu_Y$, and we reject it. Throughout this paper we let B = 1000 and $\alpha = 0.05$.

While sensitivity is a necessary condition for an evaluation measure, it is not sufficient. While it does guarantee that the ranking of systems is almost total, is does not guarantee that the systems are ranked by quality. For example, systems submitted to TREC are required to have a unique identifier. Imagine an evalua-

tion measure that computed a hash function from these identifiers to $[0, 1]$. This measure would have perfect discriminative power, and yet be completely useless.[4] Aslam *et al.* [10] proposed to assess evaluation measures by their *informativeness*. Using the maximum entropy method, evaluation measures can be viewed as inducing constraints on where in a ranked lists it is possible to find relevant documents. The more informative a measure, the more accurate and specific these constraints are. While Ashkan and Clarke [5] applied this method to several families of diversity measures, this method can be quite difficult to employ in practice and it has seen little use. It is hoped that the ease of use and expressiveness of our framework (introduced in Chapter 5) can be used to create many more meta-evaluation measures, allowing us to better interpret evaluation in the future and create better evaluation measures, as in Chapter 4.

[4] This argument is due to private conversation with Javed A. Aslam.

Chapter 3

Meta-Evaluation Metrics for Diversity

Measuring the quality of a diversity system is very difficult, since diversity also depends on a system's performance at basic ad-hoc retrieval—how many documents are relevant to any reasonable intent, especially at the top of the ranked list. Poor ad-hoc performance implies poor diversity; a system that returns few documents relevant to any intent cannot present a diverse ranked list to the user. For example, consider the following experiment. In work by Santos *et al.* [68], the authors investigate the relative impact of increasing subtopic coverage versus reducing redundancy. In one particular experiment, they show that taking a quality ad-hoc run and diversifying it using state-of-the-art algorithms can increase the α-nDCG@100 from roughly 0.35 to 0.45. However, removing all non-relevant documents from the list without any attempts at diversification increases the α-nDCG@100 to almost 0.6. This demonstrates that when we perform diversity evaluation, we must take great care to ensure that we are actually measuring the quantity of interest: the quality of the system's ordering of documents, not the quality of the documents the system retrieved.

In order to isolate a measure's sensitivity to diversity—the order in which documents are presented to the user—from its ad-hoc performance—whether the documents presented to the user are relevant—in Section 3.1 we consider artificial ranked lists created by randomly permuting relevant documents. We show experimentally that, as measured by discriminative power, existing diversity measures are insensitive to the changes in these lists, *i.e.* discriminative power alone is not sufficient to assess the quality of a diversity metric. Therefore, we develop a new meta-evaluation measure, *document selection sensitivity*, which is able to distinguish

between evaluation measures applied to perfect-performance ranked lists and can therefore be used to assess diversity metrics based on their sensitivity to diversity.

Furthermore, diversity necessarily depends on the collection over which the search is being run. If we search back-issues of the Journal of Aerospace Engineering for the query "zeppelin," we are unlikely to find many references to the band. When a collection only covers a single interpretation, even the best search engine will be unable to create a diverse ranked list. Alternatively, consider searching a corpus like Wikipedia. Since so many of the documents provide broad overviews, any relevant document is likely to be at least partially relevant to several intents, and almost any system will produce a diverse ranked list.

In order to further isolate the quantity of interest we leverage the intrinsic diversity of the collection. To do so, we develop a notion analogous to query difficulty that measures the diversity present in a collection, independent of any ranked list. *Diversity difficulty* (Section 3.2.1) measures this property at the topic (*i.e.* query) level, and *subtopic miss rate* (Section 3.2.2) measures this property at the subtopic (*i.e.* interpretation) level. Our intuition is that every system should be able to provide diverse lists for diverse topics and cover prevalent subtopics, whereas only the best systems will be able to provide diverse lists for difficult topics and cover rare subtopics. It is our hypothesis that by viewing measures from the perspective of the collection and leveraging as much information as we can, no matter how opaque to the end-user, we will be best able to distinguish between systems. While a user neither knows nor cares whether a particular query is hard, we believe that if we can find systems that are still able to perform reasonably on the most difficult queries, they will tend to best satisfy users over all.

3.1 Document Selection Sensitivity: Quantifying the Sensitivity of Diversity Evaluation to Diversity

A system's diversity is necessarily conflated with the system's ad-hoc performance. If no relevant documents are retrieved, then there is no way to present them to the user in a diversified order. A consequence of this is that a system's score with respect to a diversity evaluation measure can be increased simply by improving the systems underlying ad hoc performance [68]. Therefore, when using meta-evaluation tools to determine which diversity measure to use in practice, we must ensure that we are choosing based on the measure's sensitivity to diversity.

One way to control for the impact of ad hoc performance on diversity evaluation is to only consider lists with the same performance. We do so using artificial ranked lists created by randomly permuting the set of relevant documents. These

ERR-IA	2010	2011
Actual	0.571	0.544
Artificial	0.039	0.031

D#-nDCG	2010	2011
Actual	0.583	0.600
Artificial	0.036	0.019

Table 3.1: Discriminative power on actual submitted runs and an equal number of artificial ranked lists with perfect combined precision. Discriminative power is an order of magnitude smaller on artificial lists than on actual systems.

lists will all have the same ad-hoc performance—perfect. For a given topic, whatever difference exists between ranked lists with perfect combined precision—the percentage of retrieved documents relevant to at least one subtopic [32]—must be due solely to novelty and diversity.

We use the binary TREC subtopic relevance judgments to create artificial runs by (uniformly) randomly ranking the relevant documents of each topic. Figure 3.1 shows the discriminative power of ERR-IA and D#-nDCG at rank 20, the baseline measures reported by the TREC and NTCIR contests, on these simulated runs. We observe that discriminative power is an order of magnitude smaller than it is over actual runs. Therefore, the quality measured by discriminative power is dominated by ad hoc performance. If we wish to assess the extent to which measures are impacted by novelty and diversity, we must use an alternative framework.

For some measure M, imagine evaluating randomly selected permutations of the set of relevant documents. The observed set of evaluations has some sample mean, $\bar{x}$, and sample standard deviation, s. We define the **document selection sensitivity** of a measure as the coefficient of variation—the standard deviation divided by the mean—of the set of evaluations.

$$dss(M) = \frac{s}{\bar{x}} \tag{3.1}$$

This produces a normalized measure of the variance of ranked lists with perfect performance. A low document selection sensitivity means that it is unlikely that a system which assigns relevant documents at random will be different than the mean.[1] The larger this number, the more sensitive the measure is to the documents in ranked lists and their order. If this number is small, it does not mean that the

[1]For normally-distributed data, a coefficient of variation of $c\%$ means that roughly 68% of the population is within $+/-c\%$ of the mean.

DSS	2010	2011
ERR-IA	0.026	0.024
D#-nDCG	0.013	0.010

Table 3.2: Document Selection Sensitivity of the baseline measures reported by TREC and NTCIR.

measure did not evaluate some lists as substantially better than others: the maximum and minimum scores achieved by any of the randomly generated ranked lists could be quite different; it simply means that the majority of ranked lists have highly similar scores. For example, Table 3.2 shows the document selection sensitivity of our baseline measures. These numbers indicate that the majority of artificial, perfect ranked lists had evaluation scores quite near the mean, which is in accordance with the observed vast reduction in discriminative power.

We note that as described, document selection sensitivity does not use relevance grades or intent probabilities. We suggest that one way to use this information would be to select documents iteratively by first using the intent probability distribution to choose a subtopic, and then randomly drawing from the most relevant documents remaining for that subtopic. We hypothesize that not using relevance grades and subtopic probabilities may be beneficial, in that some diversity measures will be better able to leverage this information to distinguish between lists than others. We will address this question in future work.

A further limitation is that, as defined, DSS can only be used to evaluate measures that are always positive. While this is almost always the case, it is not universal, e.g. logit(AP). Also, DSS is not invariant to even the most simple of transformations. For example, given a measure M, the measure $M' = \frac{M+1}{2}$ would have a smaller DSS score, while still ranking systems in the exact same order.

3.2 Quantifying The Intrinsic Diversity of a Collection

The amount of diversity present in a ranked list is affected by the amount of diversity present in the collection. Therefore, when evaluating systems for diversity, it is necessary to control for the diversity of the collection. For a specific topic and corpus, query difficulty is a measure of how well a reasonable search engine can be expected to perform ad hoc retrieval. In this section, we introduce analogous notions for diversity, **diversity difficulty** (Section 3.2.1) and **Subtopic Miss Rate** (Section 3.2.2), which assesses the amount of diversity present in the collection. Like query difficulty, diversity difficulty is defined with respect to a topic and a corpus, and independent of any ranked list.

3.2.1 The Topic Level: Diversity Difficulty

In this Section we give a very brief overview of the work in the area of query difficulty prediction before describing our related notion of diversity difficulty. For further discussion, we direct the interested reader to Carmel and Yom-Tov [17].

Query Difficulty Prediction

In general, analysis of variance shows that the topics have a larger impact on evaluation than the systems being evaluated [11, 54, 79]. This huge variability in performance drove researchers to try and predict query difficulty. By determining in advance which queries would be more difficult, search engines can choose appropriate retrieval methods on a per-query basis, hopefully decreasing the overall variance in performance. The goal of this section is to situate our notion of diversity difficulty within this broader work. We do so using a taxonomy due to Carmel and Yom-Tov [17]. At the highest level, this schema first divides query difficulty prediction by whether the analysis is performed pre- or post-retrieval.[2] Pre-retrieval prediction is further divided by whether the analysis is statistical [43] or linguistic [53] in nature. Post-retrieval prediction is split into three categories: clarity, score analysis, and robustness. Clarity analyzes the difference between the top-retrieved documents and the collection as a whole [18, 36, 37]. Score analysis measures how much the document scores used by a system to rank documents vary at the top of the list and across the corpus as a whole [71, 96]. Robustness measures the extent to which retrieval is affected by perturbations. These perturbations can be to the query [80, 91, 96], to the document set [80, 95] or to the retrieval systems [6]. Our approach to measuring diversity difficulty at the topic and subtopic level is to consider the output of systems that pick relevant documents at random. Even though we do not use actual IR systems, this falls in the category of post-retrieval robustness as measured by perturbing systems.

Diversity Difficulty

Imagine a collection and a topic with ten subtopics and 1,009 relevant documents. One of these subtopics, subtopic A, is covered by 1,000 different documents. Subtopics B through J are each covered by only one relevant document. It is possible to generate a diverse ranked list that covers all ten subtopics, but it is difficult. A system would need to order those nine documents relevant to subtopics B–J high in the list—the equivalent of finding a handful of "needles" in the "haystack" of 1,000 documents relevant to subtopic A. However, imagine a different collection

[2] These categories are due to He and Ounis [44]

in which there are large numbers of documents relevant to each subtopic, or perhaps there are large numbers of documents relevant to multiple subtopics. In this collection, it would be easy to produce a diverse list. In fact, almost any list with good performance should exhibit diversity. However, this topic exhibits the same maximum amount of diversity in both collections—each of the 10 subtopics can be covered by some ranked list of ten documents. One could also imagine a third collection where subtopics A through I are each covered by many documents, each of which cover many subtopics, but there are no documents whatsoever relevant to subtopic J. In this collection, it would still be easy to create a diverse ranked list, but the maximum diversity is smaller than in the first two collections. One might argue that this simply means that subtopic J should be disregarded, and that this third collection is just as diverse. We will argue that this depends on how the collection was created, and the purpose it is intended to serve.

We consider diversity difficulty to be a function of the two factors previously discussed: (1) the maximum amount of diversity achievable by any ranked list, and (2) the ease with which a system can produce a diverse ranked list. When the maximum amount of diversity achievable by any system is small, the topic has little diversity. When the maximum amount of diversity is large but it is hard to create a diverse list, the topic is somewhat more diverse. Finally, if the maximum amount of diversity is large and a system created at random will come close to achieving it, the topic is diverse.

Given a topic, S-Recall@k [93] is the percentage of subtopics covered by a list at rank k. The S-Recall of a set of documents is the same for any ranked list of those documents. We consider the maximum amount of diversity (denoted d_{max}) for a topic to be the **Maximum S-Recall** for any set of documents in the corpus. Let ξ represent the minimum cutoff k at which d_{max} can be achieved, i.e. the minimum number of documents that cover the maximum number of subtopics. Unfortunately, computing ξ can be shown to be NP-hard by reduction from the set covering problem [19]. In this work, we use a greedy approximation.

Once we know ξ, imagine the random experiment of selecting ξ relevant documents from the corpus and measuring the S-Recall. The expectation of this experiment is analogous to the S-Recall of a system that performs ad-hoc retrieval perfectly, yet does not attempt to diversify its results. We use the **Expected S-Recall**@ξ (also denoted d_{mean}) to measure how easy it is to create a diverse list. Let M be the number of subtopics, R_i be the number of documents relevant to subtopic i, and R_T be the number of documents relevant to at least one subtopic. Then d_{mean} can

be approximated as

$$d_{mean} \approx 1 - \frac{\sum_{i=1}^{M} \left(1 - \frac{R_i}{R_T}\right)^{\xi}}{M}. \tag{3.2}$$

We note that while d_{mean} can be computed directly, we use an approximation that actually models documents sampled with replacement. Therefore this approximation can be poor when there are few relevant documents for a subtopic, e.g. when the subtopic is navigational. We define diversity difficulty, *dd*, as the harmonic mean of d_{max} and d_{mean},

$$dd = \frac{2 d_{max} d_{mean}}{d_{max} + d_{mean}}. \tag{3.3}$$

Since S-Recall is a percentage of subtopics, diversity difficulty ranges between zero and one. It is large for diverse queries where there are many subtopics and an arbitrary ranked list is likely to cover many of them. It is small for queries lacking in diversity where there are either few subtopics, or there are many subtopics but they are unlikely to be covered.[3]

TREC Collections

A commercial web search engine—which, in theory, indexes the entire web—must retrieve relevant documents for every search intent, no matter how rare. In this context, it is important to find those intents that the search engine is unable to satisfy so that the situation can be rectified. TREC collections, however, are more artificial. Designed to evaluate search engines, they consist of a first tier web crawl and topics created by visually inspecting the search logs of a commercial search engine. In this context, there are often uncovered subtopics with no relevant documents. These subtopics may not have represented common user intents, or documents pertaining to them may be missing from the crawl. Therefore, *for TREC collections only*, we restrict our attention to subtopics that are actually covered by relevant documents. However, this changes the meaning of diversity difficulty. Due to the collection we are using, *in these experiments*, the Maximum S-Recall will be 1 for any topic. In this case, topics will be considered diverse, *i.e.* dd is large, if and only if an arbitrary ranked list is likely to cover all subtopics, independent of the number of subtopics.

Measuring the diversity difficulty of the TREC 2010 and 2011 topics, we see that dd does in fact measure the diversity of topics. Table 3.3 shows several topics

[3]Note that our definition of diversity difficulty will actually be a description of diversity "easiness" in that larger values indicate topics on which systems should do well. This is similar to *e.g.* query average average precision [6]. We choose to call this diversity difficulty rather than diversity easiness to emphasize the similarity to query difficulty prediction.

Topic ID	Title	# Rel Docs	Subtopic						dd
			1	2	3	4	5	6	
143	arka-delphia health club	25	25	21	-	-	-	-	0.994
86	bart sf	82	78	62	60	-	-	-	0.977
125	butter and mar-garine	132	110	47	13	-	-	-	0.735
73	neil young	156	69	52	28	19	-	-	0.730
60	bellevue	313	254	47	16	11	4	4	0.481
57	ct jobs	261	261	14	5	2	-	-	0.449

Table 3.3: Examples of subtopic coverage and diversity difficulty in TREC 2010 and 2011 topics.

	Min	Max	Mean
2010	0.449	0.994	0.727
2011	0.643	0.977	0.809

Table 3.4: TREC 2010 and 2011 diversity difficulty statistics.

and the number of relevant documents for each subtopic. Topics 143 ("arkadelphia health club") and 86 ("bart sf"), have a non-negligible and roughly equal number of relevant documents for each subtopic. These are very diverse topics, and they have very high diversity difficulty scores of almost 1. Topics 125 ("butter and margarine") and 73 ("neil young") each cover all subtopics with many documents, but some subtopics are covered by many more documents than others. They have some diversity, which is reflected in their diversity difficulty scores of about 0.75. Topics 60 ("bellevue") and 57 ("ct jobs") both have dominant subtopics that are far more covered than the others, as well as subtopics that are barely covered. These topics have little diversity, and low diversity difficulty scores that are less than 0.5.

Figure 3.1 shows a histogram of the diversity difficulty of the topics in the combined TREC 2010 and 2011 collection. Table 3.4 shows the minimum, maximum,

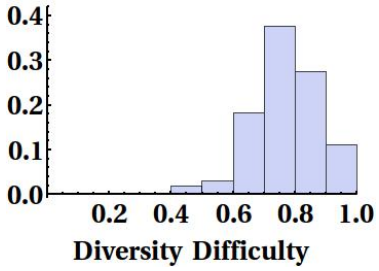


Figure 3.1: Histogram of diversity difficulties of the topics in the combined TREC 2010 and 2011 collection. The larger the value, the easier it is to create a diverse ranked list for that topic.

and mean diversity difficulty values for each year. Using diversity difficulty, we can see that the TREC 2010 and 2011 collections were diverse, with 2011 being somewhat more so.

3.2.2 The Subtopic Level: Subtopic Miss Rate

Because it is necessary to average system evaluations over topics to control for natural variations within a collection, diversity difficulty tells us which topics are naturally more diverse than others. However, for individual topics, there is variation among subtopics as well. In this section, we present **subtopic miss rate**, which, for each topic, measures the relative prevalence of documents relevant to each subtopic.

For a given topic, consider drawing relevant documents at random. Subtopics containing large numbers of relevant documents will be covered early and easily—these are the "easy" subtopics, likely to be covered by any system with reasonable ad hoc performance. However, subtopics with few relevant documents are "harder" and will likely be covered early and well by only high-quality diversity systems.

We define the **subtopic miss rate** of subtopic i at rank k as the probability of drawing k relevant documents at random and failing to cover that subtopic, normalized with respect to all of the subtopics for that topic. This forms a distribution for each topic, with probabilities corresponding to each subtopic's relative difficulty.

It is not strictly necessary to normalize these probabilities into a distribution. We do so to emphasize the relative importance of each subtopic to evaluation, rather than any universal notion of difficulty. The drawback to this approach is that the normalized probabilities are not comparable across topics.

Let M be the number of subtopics, R_i represent the number of documents rel-

evant to subtopic i, and R_T represent the total number of documents relevant to at least one subtopic. The subtopic miss rate, *smr*, of subtopic i at rank k can be approximated as

$$smr_i^k \approx \frac{\left(1 - \frac{R_i}{R_T}\right)^k}{\sum_{j=1}^{M} \left(1 - \frac{R_j}{R_T}\right)^k}. \tag{3.4}$$

While smr_i^k can be computed directly, to simplify computation, we approximate, asserting that documents are sampled with replacement. Again, this is a poor assumption for subtopics with a small number of relevant documents. If no rank is specified, we define the smr of a subtopic as the smr at rank ξ,

$$smr_i = smr_i^{\xi}, \tag{3.5}$$

where ξ is the minimum rank at which all subtopics can be covered.

If a subtopic is covered by all relevant documents, then every reasonable system should be able to cover it. The miss rate of this subtopic is zero at any rank; the subtopic is of no interest and should be ignored. However, if a subtopic is covered by only a relatively small number of relevant documents while most other subtopics are covered heavily, then the miss rate could approach one. This implies that this subtopic will be very useful in differentiating between the best systems.

Table 3.5 shows the subtopic miss rate of several TREC topics. Topic 60 has a small diversity difficulty score, meaning that it is a topic with little inherent diversity. Subtopic one has a very small subtopic miss rate—it is very unlikely to be missed by any ranked list. The remaining five subtopics all have very similar subtopic miss rates. They are all equally unlikely to be covered. This implies that the first subtopic is not useful for evaluation, and that systems should be measured by how well they cover the other five subtopics. Topic 73 has an intermediate diversity difficulty score, and is therefore a topic that is neither particularly diverse nor lacking in diversity. The subtopic miss rates are less similar, showing that several subtopics are highly likely to be covered and several subtopics are less likely to be covered. Each subtopic should contribute differently to evaluation. Topic 86 has a high diversity difficulty score; it is a diverse topic. This topic has a single "dominant" subtopic with a very high miss rate. All systems should be expected to satisfy the more common user intents, but only the best systems are likely to cover this rare one.

Topic ID	Title	*dd*	ξ	Subtopic	Rank			
					ξ	5	10	20
60	"bellevue"	0.481	3	1	0.002	0.000	0.000	0.000
				2	0.143	0.113	0.061	0.016
				3	0.199	0.196	0.182	0.144
				4	0.209	0.213	0.215	0.202
				5	0.224	0.239	0.271	0.319
				6	0.224	0.239	0.271	0.319
73	"neil young"	0.730	2	1	0.141	0.050	0.007	0.000
				2	0.202	0.122	0.040	0.003
				3	0.306	0.344	0.321	0.204
				4	0.351	0.484	0.632	0.793
86	"bart sf"	0.977	1	1	0.087	0.00	0.000	0.000
				2	0.435	0.383	0.278	0.129
				3	0.478	0.617	0.722	0.871

Table 3.5: Example subtopic miss rates of TREC topics. All subtopics tend to have similar rates for non-diverse topics (small diversity difficulty scores), whereas diverse topics (high diversity difficulty scores) tend to have "dominant" subtopics. Recall that ξ is the minimum rank at which all coverable subtopics can be covered by any ranked list.

3.3 Summary

A search engine's diversity is necessarily conflated with its ability to perform ad-hoc retrieval and the diversity of the collection. In this chapter, we introduced a meta-evaluation measure of diversity sensitivity that controls for ad hoc performance. To show that measures prefer more diverse systems, we restricted our attention to artificial ranked lists with perfect combined precision. According to discriminative power, no measure was able to distinguish between these lists. This led us to introduce document selection sensitivity, the coefficient of variation of an evaluation measure over these artificial ranked lists.

To assess collection difficulty, we developed measures at the topic and subtopic level. At the topic level, diversity difficulty blends the maximum possible number of subtopics covered by any ranked list with the number of subtopics covered by the expected ranked list. At the subtopic level, subtopic miss rate measures the probability of selecting documents at random and failing to cover subtopics. Our hypothesis is that these collection-oriented features, while opaque to the user, will be better able to differentiate between systems, thereby leading to a better overall user experience.

Chapter 4

Utilizing Meta-Evaluation Metrics to Increase the Sensitivity of Diversity Evaluation to Diversity

In this chapter, we introduce our new diversity evaluation metric and describe how it incorporates the intrinsic diversity of the collection (Section 4.1). In Section 4.2, we demonstrate that our measures retain their discriminative power. In Section 4.3, we will demonstrate that, by incorporating meta-evaluation, our measures prefer different systems than existing measures. However, just because our measures prefer different systems, it does not mean that they prefer systems that are more diverse. We use document selection sensitivity to argue that our measures do prefer more diverse systems in Section 4.4.

4.1 α#-IA Measures

Consider a hypothetical user whose search needs are satisfied by any document relevant to their intent. If the probability of each subtopic is uniform, then S-Recall represents the percentage of such users that would be satisfied by a particular ranked list. Intent aware measures can be thought of as extending this idea to non-trivial user models and non-uniform subtopic probabilities; in this framework, intent aware measures represent the expected satisfaction of a user over all possible intents. This is an attractive model of diversity, but there is no explicit novelty component: systems will be rewarded for finding multiple documents relevant to a subtopic rather than being penalized. Cascade measures do model novelty, but they do not have this feature of explicitly averaging over intents—in a sense, they "macro-average" subtopics, whereas in this work we wish to "micro-average"

them. Merging cascade measures with intent-aware measures creates a new family of intent aware cascade measures *e.g.* α-nDCG-IA. This family computes gains in the style of cascade measures using Equation 2.12, but separately for each subtopic, with each normalized against a ranked list ideal *for that subtopic*. These separate evaluations can then be merged in the style of the intent aware measures. Unfortunately, this re-inherits the problem of rewarding systems for ignoring minor intents. Therefore, our final family has a #-measure component as well. Intent-Aware cascade #-measures are defined as a linear combination of S-Recall and an intent aware cascade measure. For example,

$$\alpha\#\text{-nDCG-IA@}k = \lambda \times \text{S-Recall@}k + (1-\lambda)\sum_{i=1}^{M} w_i \times \alpha\text{-nDCG}_i\text{@}k. \tag{4.1}$$

Our goal is to develop evaluation measures that explicitly take into account the diversity present in the collection. Our hypothesis is that all systems will perform well on the easier topics for which any ranked list is likely to be diverse, and the easier, more represented subtopics that any ranked list is likely to cover. If a system performs well with regard to these topics and subtopics, it does not provide us with much information. Therefore, we wish to place more emphasis on the more difficult topics and less prevalent subtopics, as only high quality systems should be able to perform well on these.

We focus on the difficult topics and subtopics in two ways. The first, inspired by GMAP [59], is by using the geometric mean. This has the effect of amplifying the impact of topics and subtopics for which a system performed poorly. By assumption, these must have been the more difficult topics and subtopics. The second is to explicitly account for the difficulty using diversity difficulty and subtopics miss rate. Since *dd* is a number between zero and one, with zero representing a topic with no diversity, we weight each topic by one minus its diversity difficulty. *smr* can be used directly.

Experimentally, we investigate three methodologies for averaging evaluations over topics described in Table 4.1 and four methodologies for averaging over subtopics described in Table 4.2.

4.2 Discriminative Power

In this section, we show that incorporating the intrinsic diversity of the collection does not sacrifice the sensitivity of evaluation to changes in ranked lists. One of the primary measures of sensitivity appearing in the IR literature [32, 62–64] is dis-

1. **Avg**: the arithmetic average over the topics.
2. **Geom**: the geometric mean over the topics.
3. **DD**: *dd*-weighted average.

Table 4.1: Topic averaging methodologies used in α#-IA measures.

1. **Cascade**: we do not average over subtopics. As in cascade and D# measures, ranked lists are normalized by a single ideal ranked list.
2. **Micro**: each subtopic is weighted by its intent probability. Since the TREC 2010 and 2011 collections assumed intent probabilities are uniform, in our experiments, this is equivalent to the arithmetic mean.
3. **Geom**: the geometric mean.
4. **SMR**: each subtopic is weighted by its subtopic miss rate.

Table 4.2: Subtopic averaging methodologies used in α#-IA measures.

criminative power [60] (Section 2.3.3). Discriminative power measures sensitivity by conducting statistical significance testing on pairs of systems. Given the same set of queries, two different systems will produce different ranked lists. Ideally, measures should produce different sets of evaluations. The discriminative power of a measure is defined as the percentage of system pairs that are significantly different.

In this section, we compare the discriminative power of α#-IA measures with that of existing measures. There are, essentially, four aspects of α#-IA measures that can be varied: our choice of discount function (Table 2.1), our tuning of the α and λ parameters used to model a user's tolerance for redundancy and the weight given to S-Recall, respectively, our choice of topic and subtopic normalization (Tables 4.1 and 4.2), and the rank at which our evaluation is calculated. Unfortunately, it is not immediately obvious how to measure the effect of topic averaging on discriminative power. We leave this for future work. In this section, we focus on subtopic normalization. In all experiments, α and λ vary over the set $\{0, 0.1, 0.2, \ldots, 1\}$. Following Clarke *et al.* [32], when using RBP, β is set to 0.8.

Table 4.3 shows the maximum discriminative power at rank 20 of each α#-IA measure observed as α and λ are varied, as well as the maximum observed value of the D# measures as λ is varied. From this table we observe that no measures have substantially more discriminatory power than any other when parameters are appropriately tuned. We note that the α#-IA measures have more discriminatory power than D# measures, though not substantially.

Figure 4.1 shows the discriminative power of each evaluation measure at rank 20 with DCG discounting for all values of α and λ. These results were found to be typical for all three discount functions. (See Appendix A.1 for other choices of rank and discount function.) We can compare the α#-IA measures to existing measures (with the exception of D# measures) by carefully considering these plots. For any subtopic average, setting $\lambda = 1$ (the far-right side in 3D plots) shows S-Recall. Using the cascade average and setting $\lambda = 0$ (the near-left side in 3D plots) shows α-nDCG. Using the micro average and setting $\lambda = \alpha = 0$ (the leftmost corner) shows nDCG-IA. Since the maximum for each year is achieved by cascade averaging, and not on the near-left or far-right side (i.e. it is achieved with $0 < \lambda < 1$), we can conclude that the α#-IA measures do have somewhat higher discriminatory power than existing measures.

From Figure 4.1, we observe that setting $\alpha = 0.3$ and $\lambda = 0.5$ seem to be reasonable choices to use in further investigation. Figures 4.2 and 4.3 show all four subtopic averages at ranks 5,10, and 20 as one of the parameters is fixed while

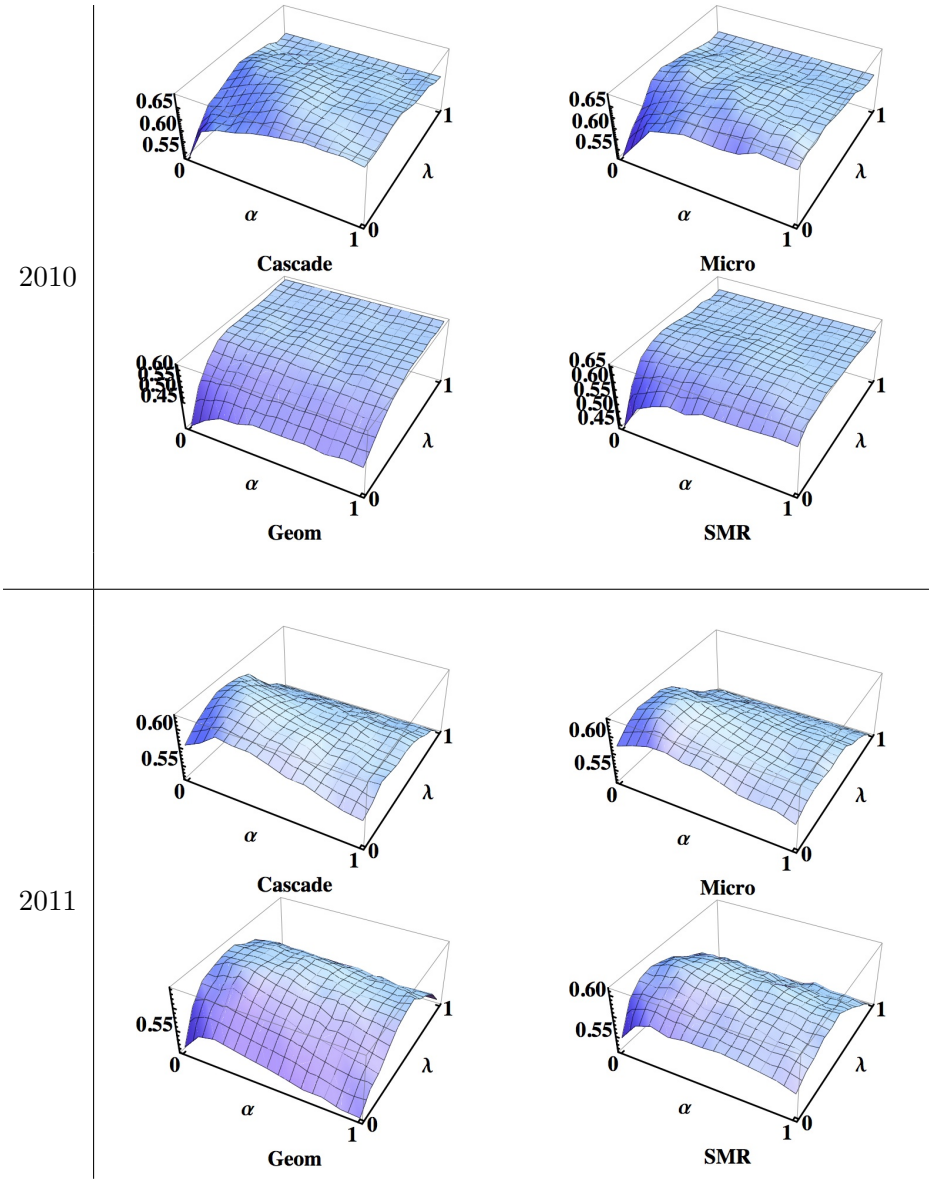


Figure 4.1: Discriminative power at rank 20 using DCG discounting as a function of α and λ. (For other choices of rank and discount function see Appendix A.1.)

the other is allowed to vary using DCG discounting. (See Appendix A.1 for other choices of discount function.) From these we conclude that while the subtopic averages that emphasize the difficult subtopics—the geometric average (geom) and the subtopic miss rate-weighted average (smr)—have lower discriminative power overall, they are comparable when α and λ are appropriately tuned.

The results of this section as a whole tell us that α#-IA measures are slightly more sensitive than existing measures, as assessed by discriminative power.

		2010	2011
Discount	Subtopic	Max	Max
ERR	Cascade	0.677	0.606
	Micro	0.673	0.610
	Geom	0.627	0.593
	SMR	0.659	0.601
D#-ERR		0.667	0.607
DCG	Cascade	0.675	0.617
	Micro	0.673	0.623
	Geom	0.617	0.595
	SMR	0.651	0.607
D#-nDCG		0.643	0.608
RBP	Cascade	0.677	0.607
	Micro	0.677	0.621
	Geom	0.617	0.595
	SMR	0.657	0.600
D#-RBP		0.653	0.595

Table 4.3: Maximum discriminative power observed on actual runs at rank 20. All choices of subtopic average and discount function have comparable maxima.

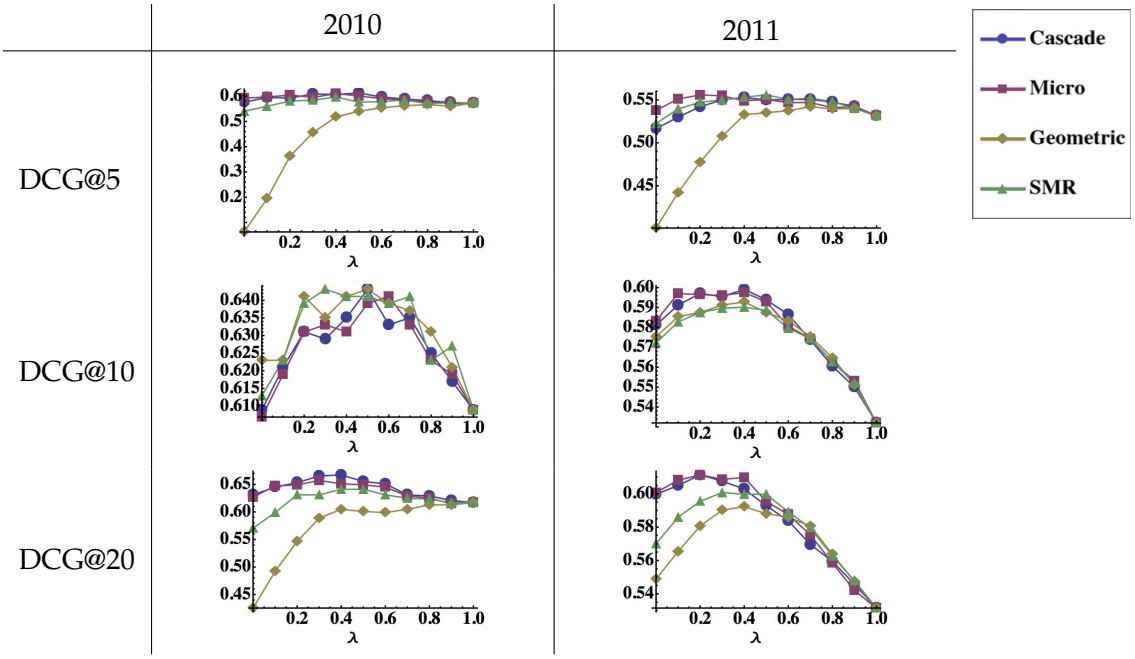


Figure 4.2: Discriminative power of as a function of λ with DCG discounting. α is fixed at 0.3. While choice of subtopic average clearly impacts discriminative power, all maxima are comparable.

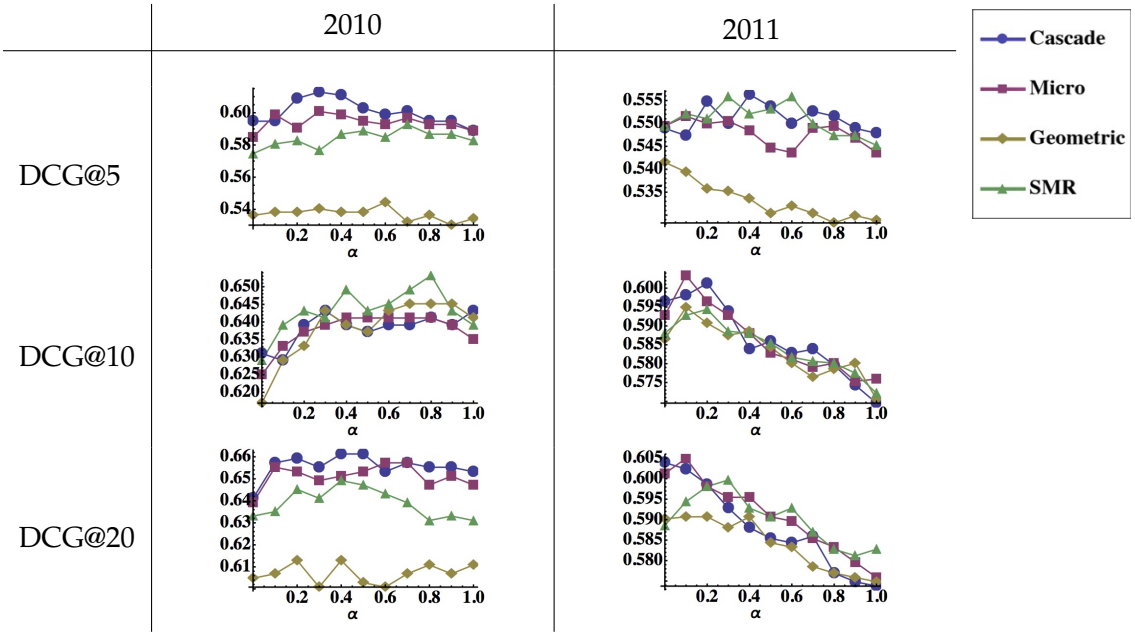


Figure 4.3: Discriminative power as a function of α with DCG discounting. λ is fixed at 0.5. While choice of subtopic average clearly impacts discriminative power, all maxima are comparable.

4.3 Impact on Evaluation

In this section, we explore the extent to which α#-IA measures evaluate systems differently than existing measures. Given a collection, an evaluation measure induces an ordering on the submitted runs. We use Kendall's τ [47] to assess the degree of correlation between the ranking of systems by different measures. By evaluating all systems submitted to TREC 2010 and 2011, we can compare the relative system rankings as computed by ERR-IA and D#-nDCG, the primary measures by which systems are evaluated at TREC and NTCIR respectively, and α#-IA measures to see how correlated they are. We can also measure the impact of the topic averaging methodologies of Section 4.1 by using them with ERR-IA and D#-nDCG. In our evaluations, we use the default parameters of TREC and NTCIR, and set $\alpha = 0.3$ and $\lambda = 0.5$ in α#-IA measures, which were shown to produce metrics with high discriminatory power in Section 4.2. This can tell us whether two measures evaluate systems similarly. However, if two measures are found to be different, it cannot tell us which of the two is better. This question will be addressed in Section 4.4.

Table 4.4 shows the baseline TREC and NTCIR measures compared with each other and several α#-IA measures. Each cell in the table shows the Kendall's τ value in 2010 and 2011, separated by a slash. With τ values ranging from roughly 0.7 to 0.9, we can see that the ERR-IA, D#-nDCG, and α#-IA measures with geometrice topic averaging and arithmetic subtopic averaging (Geom-Micro), and arithmetic topic averaging and subtopic miss rate-weighted subtopic averaging (AVG-SMR) all rank systems in highly correlated orders. However, when comparing any of these measures to α#-IA with diversity difficulty topic averaging and geometric subtopic averaging (DD-Geom), we get highly uncorrelated rankings with τ values approximately between 0.15 and 0.2. This tells us that DD-Geom evaluates systems very differently from other measures.

Tables 4.5 and 4.6 show the impact of topic averaging on the gold standard measures. With τ values ranging from 0.74 to 0.89, we can see that ordering systems by arithmetic (avg) and geometric (geom) averaging produce similar lists. However, averaging topics by their diversity difficulty (DD) produces orderings that rank systems very differently. In fact, in 2010, the τ values of 0.06 and 0.08 show that the results using ERR-IA with DD averaging are almost completely uncorrelated with the results using ERR-IA with arithmetic and geometric averaging.

Table 4.7 shows the impact of subtopic averaging on α#-IA measures. We observe that, independent of topic average, cascade normalization (casc) and geometric subtopic averaging (geom) are quite similar. This matches our intuition of α-nDCG (recall that arithmetic average and cascading subtopic average is a lin-

	ERR-IA	D#-nDCG	DD-Geom	Geom-Micro	Avg-SMR
ERR-IA	-	0.82 / 0.71	0.15 / 0.23	0.72 / 0.68	0.80 / 0.73
D#-nDCG		-	0.19 / 0.13	0.74 / 0.86	0.89 / 0.86
DD-Geom			-	0.21 / 0.16	0.20 / 0.17
Geom-Micro				-	0.79 / 0.86
Avg-SMR					-

Table 4.4: TREC and NTCIR gold standard vs a small sample of α#-IA measures. Kendall's τ 2010 / 2011.

ERR-IA	avg	geom	DD
avg	-	0.89 / 0.81	0.06 / 0.25
geom		-	0.08 / 0.17
DD			-

Table 4.5: Impact of topic averaging on ERR-IA. Kendall's τ 2010 / 2011.

ear combination of S-Recall and α-nDCG), namely that penalizing redundancy increases the impact of difficult subtopics.

We can also see that, with a minimum τ value of 0.72, if we use arithmetic topic averaging (avg, top table), the choice of subtopic averaging does not have a large impact. The impact is somewhat larger with geometric topic averaging (geom, middle table), with a minimum of 0.6. When we use diversity difficulty (DD, bottom table) topic weighting, however, the difference becomes more dramatic, with a minimum of 0.4. However, we observe that subtopic miss rate weighting (smr) is highly similar to the arithmetic average of subtopics (micro).

That diversity difficulty averaging is so different from the other averages supports our hypothesis that evaluation should use information about the collection to emphasize difficult topics. We have shown that doing so causes evaluation metrics

D#-nDCG	avg	geom	DD
avg	-	0.74 / 0.88	0.50 / 0.29
geom		-	0.56 / 0.34
DD			-

Table 4.6: Impact of topic averaging on D#-nDCG. Kendall's τ 2010 / 2011.

avg	casc	micro	geom	smr
casc	-	0.87 / 0.77	0.94 / 0.87	0.81 / 0.73
micro		-	0.82 / 0.74	0.90 / 0.94
geom			-	0.80 / 0.72
smr				-

geom	casc	micro	geom	smr
casc	-	0.72 / 0.66	1.00 / 0.92	0.72 / 0.65
micro		-	0.73 / 0.62	0.99 / 0.97
geom			-	0.72 / 0.60
smr				-

DD	casc	micro	geom	smr
casc	-	0.44 / 0.63	0.99 / 1.00	0.40 / 0.60
micro		-	0.45 / 0.63	0.90 / 0.94
geom			-	0.41 / 0.60
smr				-

Table 4.7: Impact of subtopic averaging on α#-IA measures. Kendall's τ 2010 / 2011.

to prefer different systems (although we have not yet shown that it causes metrics to prefer more diverse systems). However, Table 4.7 also shows that subtopic miss rate-weighted averaging (smr) is very similar to the arithmetic average of subtopics (micro), suggesting that it would be better to emphasize subtopics on which systems performed poorly, rather than subtopics that we expect to be difficult. We believe that our hypothesis is valid, but our approximation of smr is not. We discuss this further in Section 4.4 and the conclusion of this chapter (Section 4.5).

4.4 Sensitivity Experiments

For each topic, 1,000 ranked lists were created by ranking the relevant documents at uniformly random. These ranked lists were used to compute the document selection sensitivity of each measure on each topic. To increase the impact of the topic averaging scheme being used, the reported results are averaged over all 100 topics in TREC 2010 and TREC 2011 combined. As before, α—which models a user's tolerance for redundancy—and λ—which controls the mixture with S-Recall—vary over the set $\{0, 0.1, 0.2, \ldots, 1\}$. Table 4.8 shows that, unlike discriminatory power, document selection sensitivity can be affected by choice of topic and subtopic averaging. At rank 20, document selection sensitivity ranges from a low of 0.05 to

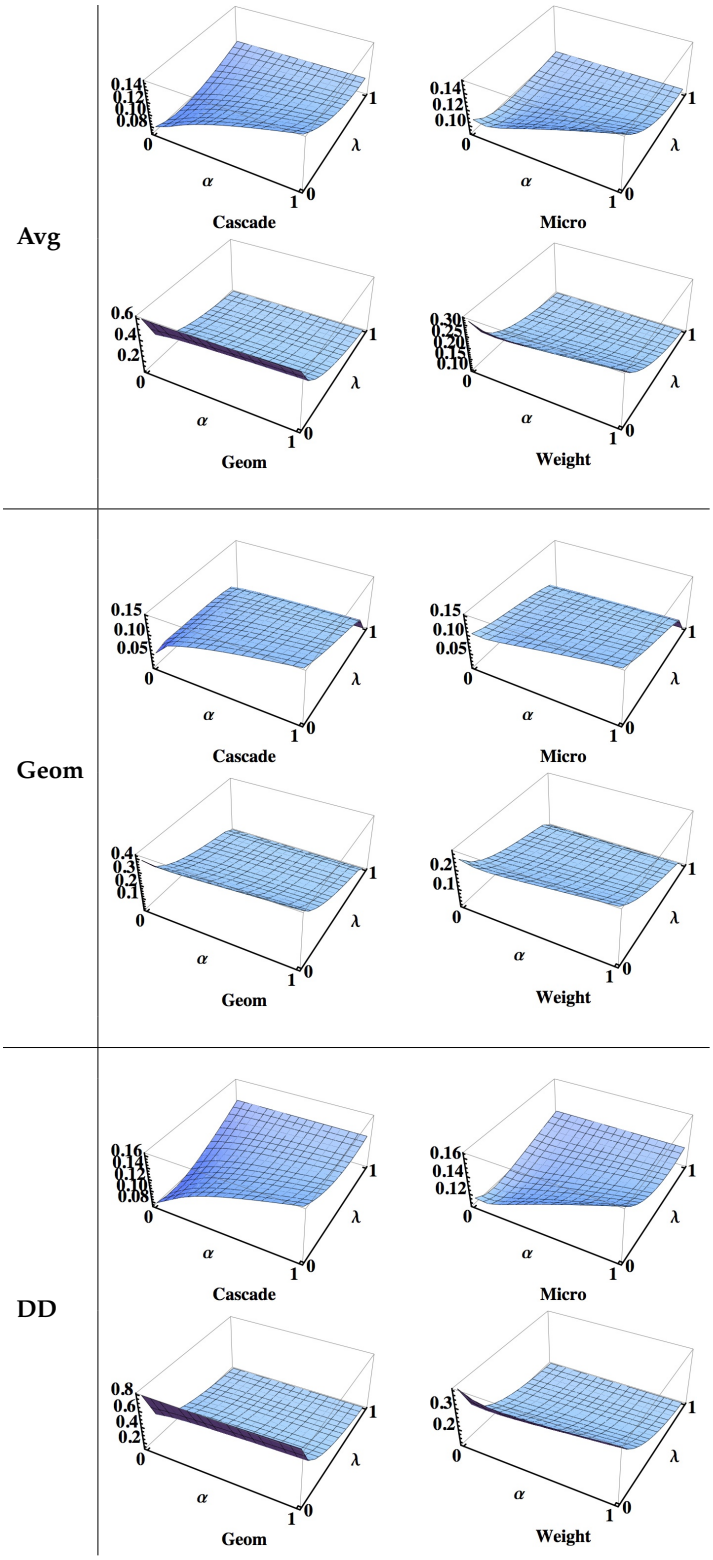


Figure 4.4: Document selection sensitivity at rank 20 as a function of α and λ using DCG discounting. (For other ranks and discount functions see Appendix A.2.) Choice of topic and subtopic averaging can have a substantial impact.

a high of 0.8. This can also be seen in Figure 4.4, which shows the selection sensitivity using DCG discounting at rank $k = 20$. (For other ranks and discount functions see Appendix A.2.) Geometric (geom) and arithmetic topic averaging are quite similar. Diversity difficulty topic weighting (DD) shows marked increases in selection sensitivity, as does geometric subtopic weighting (geom). Subtopic miss rate weighting (smr) has higher selection sensitivity than subtopic intent-weighted (micro) and cascade normalization.

Figures 4.5 and 4.6 show the selection sensitivity of the topic averages at ranks $k = 5$, 10, and 20, each with one of the parameters fixed using DCG discounting. (For other discount functions see Appendix A.2.) From these figures, as well as Table 4.8, we can see that selection sensitivity clearly goes down as the cutoff increases. This makes sense intuitively. Imagine that there are 20 relevant documents, 5 documents relevant to all subtopics and 15 documents each relevant to a single subtopics. At $k = 20$, you will get at least some gain from every relevant document. At $k = 5$, you may see the 5 documents relevant to all subtopics, or you may see none of them. Seeing all of them versus none of them should have more variance than seeing all of them in different orders. Consulting Table 4.8, we can see that α#-IA measures clearly have higher document selection sensitivity than D# measures. We can compare the α#-IA measures against the other existing measures by carefully considering Figure 4.4. Again, for any subtopic average, setting $\lambda = 1$ (the far-right side in 3D plots) shows S-Recall. Using the cascade average and setting $\lambda = 0$ (the near-left side in 3D plots) shows α-nDCG. When the subtopic intent distribution is uniform, then using subtopic intent-weighted averaging (micro) and setting $\lambda = \alpha = 0$ (the leftmost corner) shows nDCG-IA. Since the maximum is achieved by geometric subtopic averaging (geom), and not on the far-right side where $\lambda = 1$, we can conclude that the α#-IA measures can have significantly higher document selection sensitivity than existing measures.

According to document selection sensitivity, one should use diversity difficulty topic averaging and geometric subtopic averaging. That one should use diversity difficulty topic averaging supports our hypothesis that evaluation should take a collection-oriented view, emphasizing topics that are difficult, rather than a user-oriented view, emphasizing topics with poor results. However, as with Section 4.3, geometric subtopic averaging outperforms subtopic miss rate-weighted averaging. This would seem to directly contradict our hypothesis; by emphasizing small values, geometric subtopic averaging takes the user-oriented view, emphasizing subtopics on which the user is expected to be left unsatisfied. Instead, we believe that this is likely due to our particular implementation of subtopic miss rate,

			ERR	DCG	RBP
Topic	Subtopic	Rank	Max	Max	Max
Avg	Cascade	5	0.229	0.218	0.218
		10	0.210	0.174	0.176
		20	0.204	0.152	0.170
	Micro	5	0.229	0.218	0.218
		10	0.211	0.174	0.174
		20	0.204	0.154	0.154
	Geom	5	1.558	1.556	1.556
		10	1.012	0.978	0.978
		20	0.683	0.608	0.608
	SMR	5	0.598	0.529	0.516
		10	0.518	0.404	0.422
		20	0.480	0.327	0.391
	D#	5	0.218	0.218	0.218
		10	0.152	0.152	0.152
		20	0.116	0.096	0.111
Geom	Cascade	5	0.224	0.204	0.196
		10	0.205	0.168	0.167
		20	0.199	0.149	0.162
	Micro	5	0.224	0.204	0.204
		10	0.206	0.169	0.169
		20	0.199	0.151	0.151
	Geom	5	1.176	1.120	1.120
		10	0.761	0.667	0.667
		20	0.540	0.420	0.420
	SMR	5	0.504	0.436	0.423
		10	0.448	0.344	0.344
		20	0.425	0.292	0.323
	D#	5	0.176	0.175	0.175
		10	0.109	0.095	0.097
		20	0.081	0.057	0.077
DD	Cascade	5	0.260	0.260	0.260
		10	0.220	0.194	0.201
		20	0.211	0.163	0.193
	Micro	5	0.260	0.260	0.260
		10	0.220	0.194	0.194
		20	0.211	0.165	0.165
	Geom	5	2.015	2.029	2.029
		10	1.319	1.300	1.300
		20	0.871	0.801	0.801
	SMR	5	0.719	0.642	0.629
		10	0.612	0.492	0.518
		20	0.559	0.383	0.478
	D#	5	0.260	0.260	0.260
		10	0.194	0.194	0.194
		20	0.129	0.129	0.129

Table 4.8: Maximum observed document selection sensitivity. Smaller cutoffs have higher sensitivity.

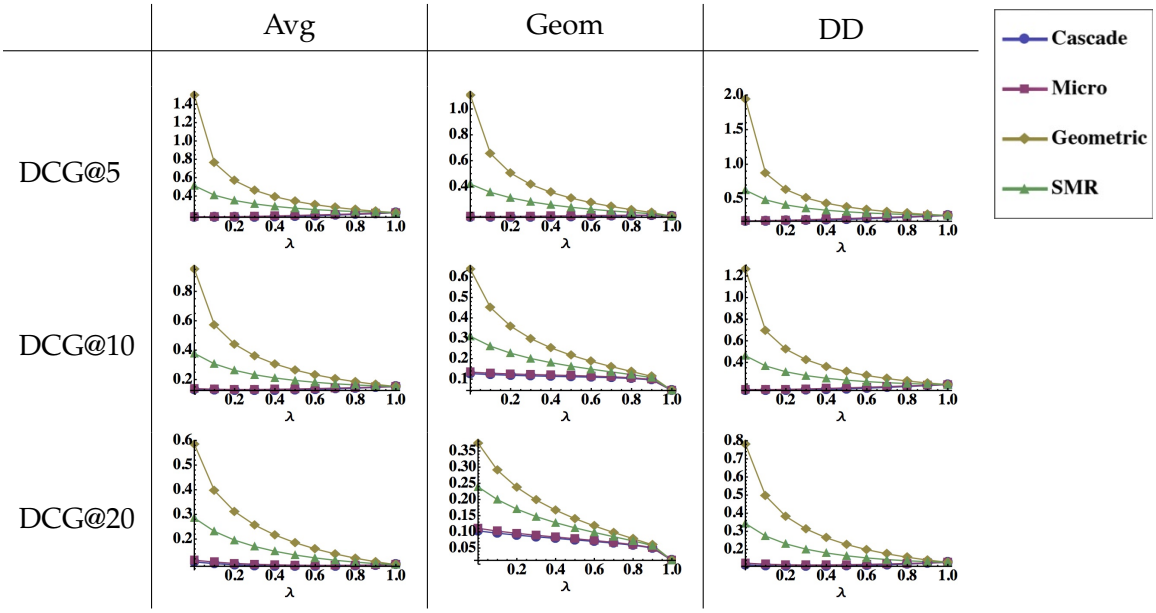


Figure 4.5: Document Selection Sensitivity as a function of λ with DCG discounting. (For other discount functions see Appendix A.2.) α is fixed at 0.3.

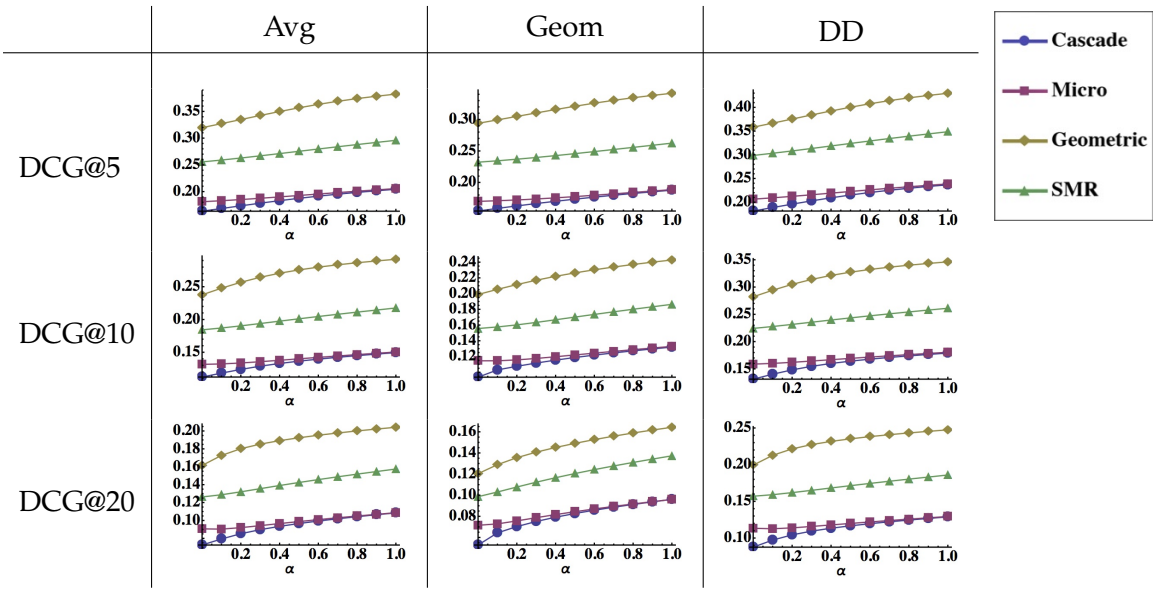


Figure 4.6: Document Selection Sensitivity as a function of α with DCG discounting. (For other discount functions see Appendix A.2.) λ is fixed at 0.5.

both the approximation of sampling with replacement, and by choosing to measure subtopic miss rate at ξ, which is often as high as rank one or two. We believe that geometric subtopic averaging, by emphasizing the subtopics where systems performed poorly, as would be expected of subtopics with large subtopic miss rates, was actually a better approximation of subtopic miss rate than our computed approximation. We will revisit this in future work. We do observe that smr, even as approximated here, still outperforms micro and cascade normalization with respect to document selection sensitivity.

4.5 Summary

In this chapter, we attempted to isolate diversity from confounding factors so that we can begin to understand it. To this end, we introduce a new family of measures that explicitly accounts for collection diversity. We showed that α#-IA measures, which combine the best features of existing evaluation measures and emphasize difficult topics and subtopics, sometimes rank systems in quite different orders than existing measures, yet have slightly more discriminative power.

That our measures prefer different systems does not indicate that they prefer more diverse systems. However, according to document selection sensitivity, α#-IA measures that explicitly account for collection diversity were far more sensitive to differences in these lists than existing measures, suggesting that these measures may prefer more diverse systems. However, while averaging subtopics by their difficulty also led to higher document selection sensitivity, it was still less than geometric averaging. This is likely due to limitations of our implementation of difficulty at the subtopic level. We believe that these results support our hypothesis that taking a collection-oriented view of evaluation leads to systems that are preferable to the user.

Chapter 5

An Information-Theoretic Framework for Unifying Evaluation and Meta-Evaluation

In Chapters 3 and 4, we demonstrated the utility of developing meta-evaluation measures and incorporating them into the measurement of system quality. In this chapter, we introduce a powerful new probabilistic framework that, by allowing us to view evaluation using the tools of information theory, provides immediate access to a large number of powerful evaluation and meta-evaluation tools allowing for a deeper understanding of the performance of search engines. It is our hope that by creating a single, unifying framework that can be used to perform evaluation and can be easily manipulated to create meta-evaluations, future researchers will be able to incorporate meta-evaluations into evaluation measures as we did in Chapter 4.

Our framework for evaluation is based on the same observation that underlies BPref, that relevance judgments can also be interpreted as a preference between documents having different relevance grades. However, we take this one step further by interpreting relevance judgments as a retrieval system. Therefore, evaluation can be considered as a rank correlation between systems and relevance judgments. To this end, we develop a probabilistic framework for rank correlation based on the expectation of random variables, which we demonstrate can also be used to compute existing evaluation metrics. However, the true value of our framework lies in its potential for creating new, information-theoretic meta-evaluation tools that can be, hopefully, be more easily unified with evaluation.

In Section 5.1, we motivate and define our new framework and explore it's rela-

tionship with traditional evaluation and rank correlation measures. In Section 5.2, we demonstrate a practical application of re-interpreting these kinds of analyses information-theoretically, namely that two ranked lists can be compared conditionally with respect to a third. In Section 5.3, we define a new evaluation measure, **Relevance Information Correlation**, which is defined within our framework. In Section 5.4, we demonstrate another practical application of re-interpreting these kinds of analyses information-theoretically, namely that multiple ranked lists, and, hopefully, in the future, multiple QRELs, can be considered jointly.

5.1 A Probabilistic Interpretation of Rank Correlation

In this section, we define our probabilistic framework for evaluation and meta-evaluation. We begin by deriving our framework from traditional rank correlation in Section 5.1.1. In Section 5.1.2, we demonstrate how to compute traditional evaluation measures in our probabilistic framework. In Section 5.1.3, we prove that our information-theoretic interpretation of our framework is equivalent to the traditional rank correlation measures from which it was derived.

5.1.1 Derivation from Traditional Rank Correlation

Mathematically, one can view a search system as providing a *total ordering* of the documents ranked and a *partial ordering* of the entire collection, where all ranked documents are preferred to unranked documents but the relative preference among the unranked documents is unknown. Similarly, one can view the relevance assessments as providing a partial ordering of the entire collection: in the case of binary relevance assessments, for example, all judged relevant documents are preferred to all judged non-relevant and unjudged documents, but the relative preferences among the relevant documents and among the non-relevant and unjudged documents is unknown. Thus, mathematically, one can view retrieval evaluation as comparing the partial ordering of the collection *induced by the search system* with the partial ordering of the collection *induced by the relevance assessments*.

To formalize and instantiate a framework for comparing such partial orderings, consider the simplest case where we have two total orderings of objects, i.e., where the entire "collection" of objects is fully ranked in both "orderings." While such a situation does not typically arise in search system evaluation (since not all *documents* are ranked by the retrieval system nor are they fully ranked by relevance assessments), it does often arise when comparing the *rankings of systems* induced by two (or more) evaluation metrics; here Kendall's τ is often the metric used to compare these (total order) rankings. In what follows, we define a *probabilistic framework* within which to compare two total orderings, and we show how traditional metrics

(such as Kendall's τ) are easily cast within this framework.

Consider two total orderings of n objects. There are $\binom{n}{2}$ (unordered) pairs of such objects. A pair is said to be *concordant* if the two orderings agree on the relative rankings of the objects and *discordant* if the two orderings disagree. Let c and d be the number of concordant and discordant pairs, respectively. Then Kendall's τ is defined as follows:

$$\tau = \frac{c-d}{c+d}. \tag{5.1}$$

If we let C and D denote the *fraction* of concordant and discordant pairs then Kendall's τ is defined as

$$\tau = C - D. \tag{5.2}$$

Note that $c + d \neq \binom{n}{2}$ if there are ties.[1]

We define our probabilistic framework in terms of three things: (1) a sample space of objects, (2) a distribution over this sample space, and (3) random variables over this sample space. Let our sample space Ω be all possible $2 \cdot \binom{n}{2}$ *ordered* pairs of distinct objects, and consider a *uniform distribution* over this sample space. For a given ranking R, define a random variable $X_R : \Omega \to \{-1, +1\}$ that outputs $+1$ for any ordered pair concordant with R and -1 for any ordered pair discordant with R.

$$X_R\left[(d_i, d_j)\right] = \begin{cases} +1 & \text{if } d_i \text{ appears before } d_j \text{ in } R. \\ -1 & \text{otherwise.} \end{cases} \tag{5.3}$$

We thus have a well-defined *random experiment*: draw an ordered pair of objects at random and output $+1$ if that ordered pair agrees with R's ranking and -1 otherwise. Since all ordered pairs of objects are considered uniformly, the *expected value* $E[X_R]$ of this random variable is zero.

Given a second ranked list S, one can similarly define an associated random variable X_S. Now consider the random experiment of *multiplying* the two random variables: the product $X_R \cdot X_S$ will be $+1$ precisely when the pair is *concordant*—i.e. both lists agree that the ordering of the objects is correct ($+1$) or incorrect (-1), and the product will be -1 when the pair is *discordant*—i.e. the lists disagree. In this probabilistic framework, Kendall's τ is the expected value of the product of these

[1]Kendall defined two means by which τ can account for ties, depending on the desired behavior. Imagine comparing two ranked lists, one of which is almost completely composed of ties. τ_A, defined above, approaches $+/-1$. τ_B includes the number of ties in the denominator, and therefore approaches 0. We believe that the former approach is appropriate in this context. Since QRELs are almost exclusively composed of ties (recall that all pairs of unjudged documents in the corpus are considered to be tied), using the latter would mean that effect of the relatively rare meaningful comparisons would be negligible.

random variables:

$$\tau = E[X_R \cdot X_S]. \tag{5.4}$$

The real power of this framework is in the definition of these random variables and the ability to manipulate the probability distriubtion. This gives us the ability: (1) to generalize such that we can comparing partial orderings as they arise in system evaluation, and (2) to measure the correlation of these random variables using information-theoretic techniques.

5.1.2 Traditional Evaluation Measures in Our Probabilistic Framework

In this section, we demonstrate how to compute AP and nDCG in our probabilistic framework. To compute average precision,[2] recall the observation by Yilmaz and Aslam [87] that average precision can be formulated as the expectation of the random experiment described in Table 2.2. To compute this expectation in our framework, let $\Omega = \{(d_i, d_j)\}$ be the set of all ordered pairs of documents. With respect to a ranked list R, define the random variable $X_R \colon \Omega \to \{0, 1\}$ as

$$X_R\left[(d_i, d_j)\right] = \begin{cases} 1 & \text{if } d_j \text{ appears before } d_i \text{ in } R. \\ 0 & \text{otherwise.} \end{cases} \tag{5.5}$$

Define a QREL variable $Q \colon \Omega \to \{0, 1\}$ as

$$Q\left[(d_i, d_j)\right] = \begin{cases} 1 & \text{if } d_j \text{ is relevant.} \\ 0 & \text{otherwise.} \end{cases} \tag{5.6}$$

To compute AP, we must define the probability distribution in terms of the ranked list. Let

$P_R = U_{I(\{(d_i,d_j)|d_i \text{ is relevant, } d_j \text{ appears before } d_i\})}$, i.e. the uniform distribution over all such pairs of documents, be the probability distribution associated with the list R, then

$$\text{AP} = E_{P_R}[X_R \cdot Q]. \tag{5.7}$$

To compute nDCG@k, let $\Omega = \{d_i\}$ be the set of all documents. With respect to a ranked list R, define the random variable $X_R \colon \Omega \to \{0, 1\}$ as

$$X_R(d_i) = \frac{1}{\lg(r_i + 1)} \tag{5.8}$$

[2]Our formulation can be extended to GAP. We restrict our discussion to binary relevance for the sake of clarity.

where r_i is the rank of document d_i in R. Define a QREL variable $Q\colon \Omega \to \{0,1\}$ as

$$Q(d_i) = k(2^{g_i} - 1) \tag{5.9}$$

where g_i is the relevance grade of document d_i. If we Let $P_R = U_{I(r_i \leq k)}$ be the probability distribution associated with the list R, then

$$\text{DCG@}k = E_{P_R}[X_R \cdot Q]. \tag{5.10}$$

To compute nDCG, we simply compute the DCG of the ideal list as above, and normalize.

5.1.3 Correspondance with Traditional Rank Correlation

In Section 5.1.1, we defined Kendall's τ as the expected product of random variables. The following theorem allows us to restate Kendall's τ equivalently as the mutual information between those random variables.

Theorem 5.1. $I(X_R; X_S) = \frac{1+\tau}{2}\log(1+\tau) + \frac{1-\tau}{2}\log(1-\tau)$.

Proof. Denote X_R and X_S as X and Y. Consider the following joint probability distribution table.

		Y	
		-1	$+1$
X	-1	a	b
	$+1$	c	d

Observe that: $a + b + c + d = 1$; $C = a + d$, $D = b + c$. Therefore $\tau = a + d - b - c$; and since document pairs appear in both orders, $a = d$ and $b = c$.

The joint probability distribution can be rewritten as follows.

		Y	
		-1	$+1$
X	-1	$\frac{C}{2}$	$\frac{D}{2}$
	$+1$	$\frac{D}{2}$	$\frac{C}{2}$

Observe that, since $C+D = 1$, the marginal probability $P(X) = P(Y) = \left(\frac{C}{2} + \frac{D}{2}, \frac{C}{2} + \frac{D}{2}\right) = \left(\frac{1}{2}, \frac{1}{2}\right)$.

Recall that

$$\begin{aligned} I(X;Y) &= KL(P(X,Y)||P(X)P(Y)) \\ &= \sum_{x,y} p(x,y) \lg \frac{p(x,y)}{p(x)p(y)} \\ &= \sum_{x,y} p(x,y) \lg p(x,y) + \sum_{x,y} p(x,y) \lg \frac{1}{p(x)p(y)}. \end{aligned} \tag{5.11}$$

Since $P(X,Y) = \left(\frac{C}{2}, \frac{D}{2}, \frac{C}{2}, \frac{D}{2}\right)$ and $P(X)P(Y) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$,

$$\begin{aligned} I(X,Y) &= 2 \cdot \frac{C}{2} \lg \frac{C}{2} + 2 \cdot \frac{D}{2} \lg \frac{D}{2} + 2 \cdot \frac{C}{2} \lg 4 + 2 \cdot \frac{D}{2} \lg 4 \\ &= C \lg \frac{C}{2} + D \lg \frac{D}{2} + 2C + 2D \\ &= C \lg C - C + D \lg D - D + 2C + 2D \\ &= C \lg C + D \lg D + 1 \\ &= C \lg C + (1-C) \lg(1-C) + 1 \end{aligned} \tag{5.12}$$

Since $C + D = 1$ and $\tau = C - D$, we have that $\tau = 2C - 1$, $C = \frac{1+\tau}{2}$ and $D = 1 - C = \frac{1-\tau}{2}$.

In terms of C, if H_2 represents the entropy of a Bernoulli random variable,[3]

$$\begin{aligned} I(X;Y) &= -H_2(C) + 1 \\ &= -H_2\left(\frac{1+\tau}{2}\right) + 1 \\ &= \frac{1+\tau}{2} \lg \frac{1+\tau}{2} + \frac{1-\tau}{2} \lg \frac{1-\tau}{2} + 1 \\ &= \frac{1+\tau}{2} \lg(1+\tau) - \frac{1+\tau}{2} + \frac{1-\tau}{2} \lg(1-\tau) \\ &\quad -\frac{1-\tau}{2} + 1 \\ &= \frac{1+\tau}{2} \lg(1+\tau) + \frac{1-\tau}{2} \lg(1-\tau) \end{aligned} \tag{5.13}$$

□

Corollary 5.1. *For two ranked lists R and S, $I(X_R; X_S) = 1 - H_2(K)$ where $K = \frac{1-\tau}{2}$ is the normalized Kendall's τ distance between R and S.*

[3] $H_2(p) = -p \lg p - (1-p) \lg(1-p)$. Note that $H_2(p) = H_2(1-p)$.

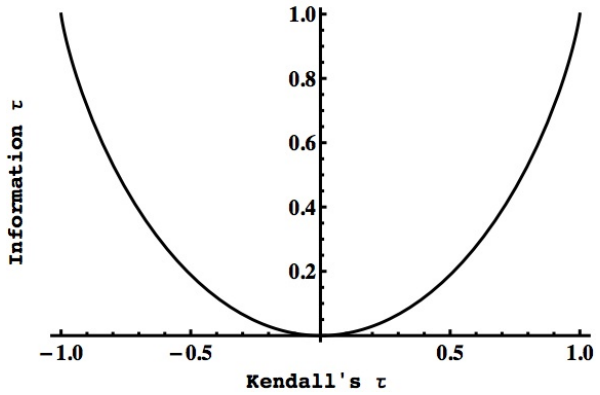


Figure 5.1: Information τ as a function of Kendall's τ. Note that the function is bijective for $\tau \geq 0$.

Unlike Kendall's τ, the mutual information between ranked lists ranges from 0 on lists that are completely uncorrelated to 1 on lists that are either perfectly correlated or perfectly anti-correlated.

5.2 Meta-Evaluation Application #1: Conditional Rank Correlation

In Section 5.1.1, we defined Kendall's τ as the expected product of random variables. In Section 5.1.3, we proved Theorem 5.1, allowing us to restate Kendall's τ equivalently as the mutual information between the random variables. If we restrict our attention to pairs of lists that are not anti-correlated, then the relationship is bijective (see Figure 5.1). Given this fact, we define a variant of Kendall's τ, *information* τ:

$$\tau_I(R, S) = I(X_R; X_S) \tag{5.14}$$

where X_R is the ranked list random variable defined in Equation 5.3 observed with respect to the uniform probability distribution over all pairs of distinct objects. By reframing Kendall's τ equivalently in terms of mutual information, we immediately gain access to a large number of powerful theoretical tools. For example, we can define a conditional information τ between two lists given a third. For lists R and S given T,

$$\tau_I(R, S \mid T) = I(X_R; X_S \mid X_T). \tag{5.15}$$

Kendall's τ can tell you whether two sets of rankings are similar, but it cannot tell you why. Information τ can be used as a meta-evaluation tool to find the underlying cause of correlation between measures. We demonstrate the use of information τ as a meta-evaluation tool by using it to analyze measures of the *diversity*

of information retrieval systems. As we discussed in Chapter 3, diversity measures conflate several factors including: a diversity model that rewards novelty and penalizes redundancy, and a measure of ad hoc performance that rewards systems for retrieving highly relevant documents. We wish to know not only whether two diversity measures are correlated, but also the similarity between their component diversity models. Using Kendall's τ, we can observe whether the rankings of systems by each measure are correlated. But even if they are correlated, this could still be for one of two reasons: either both the diversity and the performance components evaluate systems similarly; or else one of the components is similar, and its effect on evaluation is dominant. However, if the measures are correlated when conditioned on their underlying performance components, then this must be due to similarities in their models of diversity.

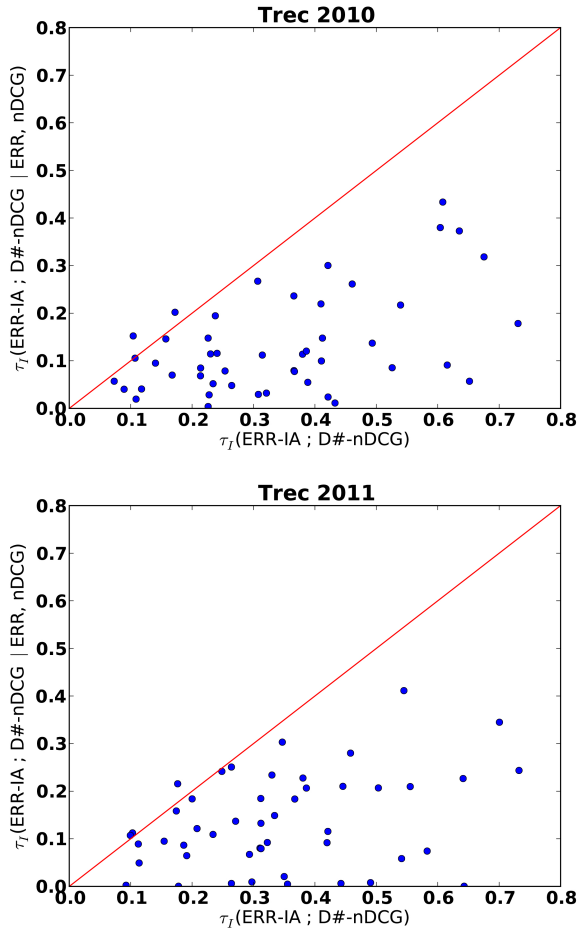


Figure 5.2: Per-query information τ (conditional rank correlation) between the TREC and NTCIR gold standard diversity measures conditioned on their underlying performance measures.

	TREC 2010	TREC 2011
τ_I(ERR-IA ; α-nDCG)	0.8929	0.8375
τ_I(ERR-IA ; α-nDCG | nDCG)	0.4860	0.4434
τ_I(ERR-IA ; α-nDCG | ERR)	0.2499	0.3263
τ_I(ERR-IA ; α-nDCG | nDCG, ERR)	0.2451	0.2805
τ_I(ERR-IA ; D#-nDCG)	0.6390	0.5545
τ_I(ERR-IA ; D#-nDCG | nDCG)	0.3026	0.1728
τ_I(ERR-IA ; D#-nDCG | ERR)	0.1222	0.1442
τ_I(ERR-IA ; D#-nDCG | nDCG, ERR)	0.1239	0.1003

Table 5.1: TREC 2010 and 2011 information τ (conditional rank correlation) between diversity measures conditioned on ad hoc performance measures.

We measured this effect on the the TREC 2010 and 2011 Web collections [29,30]. Note that the performance measures are evaluated using graded relevance, while the diversity measures use binary judgments for each subtopic. All evaluations are performed at rank 20. Figure 5.2 shows the rank correlation between ERR-IA and D#-nDCG, the primary measures reported by TREC and NTCIR [76], when conditioned on their underlying performance models. Each query is computed separately, with each datapoint in the figure corresponding to a different query. Table 5.1 shows the results of conditioning additional pairs of diversity measures (now averaged over queries in the usual way) on their performance models. The results in Figure 5.2 are typical of all choices of measure on a per-query basis. For additional choices, see Appendix A.3.

Our results confirm that while diversity measures are very highly correlated, most of this correlation disappears when one conditions on the underlying performance model. This indicates that most of the correlation is due to the similarity between the performance components and not the diversity components. For example, in TREC 2010, ERR-IA and α-nDCG have an information τ of almost 0.9. However, when conditioned on ERR, the similarity falls to only 0.25. This means that while these two measures are mostly ranking systems for the same reason, that reason is simply ERR. However, of the 0.9 bits that are the same, 0.25 are due to some factor other than ERR. This other factor must presumably be the similarity in their diversity models.

5.3 Evaluation within our Framework

In Section 5.1, we demonstrated a probabilistic framework for evaluation based on the correlation between a system and the incomplete ranking generated by a set of relevance judgments. In this section we define an information-theoretic evaluation measure, **relevance information correlation**. We define the measure in Section 5.3.1. In Section 5.3.2, we demonstrate that our measure is consistent with existing measures.

5.3.1 Relevance Information Correlation

To use our probabilistic framework, we must define a sample space, a probability distribution, and random variables. Let the sample space, $\Omega = \{(d_i, d_j)\}$, be the set of all ordered pairs of judged documents. This means that we are ignoring unjudged documents, rather than considering them non-relevant. This is equivalent to computing an evaluation measure on the *condensed list* [61] created by removing all non-judged documents from the list. We define the probability distribution in terms of the QREL to ensure that all ranked lists will be evaluated using the same random experiment. Initially, let $P = U|_{I(g_i \neq g_j)}$, where g_i represents the relevance grade of document d_i, be the uniform probability distribution over all pairs of documents whose relevance grades are not equal.[4]

We define a QREL variable Q over ordered pairs of documents as

$$Q\left[(d_i, d_j)\right] = \begin{cases} 1 & \text{if } g_i > g_j \\ 0 & \text{otherwise.} \end{cases} \tag{5.16}$$

Note that this definition can be applied to both graded and binary relevance judgments.

We now turn our attention to defining a ranked list random variable over ordered pairs of documents (d_i, d_j). If both document d_i and d_j appear in the ranked list, than our output can simply indicate whether d_i was ranked above d_j. If document d_i appears in the ranked list and d_j does not, then we will consider d_i as having been ranked above d_j, and vice versa. If neither d_i nor d_j is ranked, we will output a null value. If we were to instead restrict our attention only to judged document pairs where at least one document is ranked, then a ranked list consisting of a single relevant document followed by some number of non-relevant documents would have perfect mutual information with the QREL since all of the ranked rele-

[4] This distribution is sufficient for performing traditional, recall-oriented evaluation. We will introduce a different probability distribution later which will be used for precision-oriented evaluation at arbitrary ranks.

vant documents appear before all of the ranked non-relevant documents. However, this system must be penalized for preferring all of the ranked non-relevant documents to all of the unranked relevant documents. If we instead use a null value, our example ranked list would almost always output null. This behavior would be independent of the QREL, meaning the two variables will have almost no mutual information. In effect, the null value creates a recall component for our evaluation measure; no system can have a large mutual information with the QREL unless it retrieves most of the relevant documents.

Another problem we must consider is that mutual information is maximized when two variables are completely correlated or completely *anti*-correlated. Consider an example ranked list consisting of a few non-relevant documents followed by several relevant documents and then many more non-relevant documents. Since this example ranked list will disagree with the QREL on almost all document pairs, its random variable will have a very high mutual information with the QREL variable. The system is effectively being rewarded for finding the subset of non-relevant documents that happen to be present in the QREL. To address this, we truncate the list at the last retrieved relevant document prior to evaluation.

Let r_i represent the rank of document d_i in the truncated list S. Then the ranked list variable R_S is defined as

$$R_S\left[(d_i, d_j)\right] = \begin{cases} 1 & \text{if } r_i < r_j \\ 0 & \text{if neither } d_i \text{ nor } d_j \text{were retrieved} \\ -1 & \text{otherwise.} \end{cases} \tag{5.17}$$

We define our new measure, **Relevance Information Correlation**, as the mutual information between the QREL variable Q and the ranked list variable R:

$$RIC(System) = I(R_{System}; Q) \tag{5.18}$$

RIC is computed separately for each query, and then averaged, as with mean average precision. In order to compute RIC we must estimate the joint probability distribution of document preferences over Q and R. This could be done in various ways. In this work, we use the maximum likelihood estimate computed separately for each query. We also note that RIC has no explicit rank component, and would therefore seem to treat all relevant documents equally independent of the rank at which they were observed. However, there is an *implicit* rank component in that a relevant document that is not retrieved early in the list must be incorrectly ranked below many non-relevant documents. This argument is similar in spirit to BPref.

Precision-Oriented Evaluation

The discussion above is inherently recall-oriented. Our precision-oriented version, RIC@k, differs from RIC in two ways. First, we normalize with respect to the maximum possible RIC@k of an ideal ranked list, as with nDCG. Second, we alter the probability distribution so as to give more weight to documents with higher relevance grades. To do so, we begin by observing that evaluation metrics can be viewed as inducing probability distributions over ranks. For example, Carterette [20] derives the probability of stopping at a rank k from to $nDCG$ as

$$P_{DCG}(k) = \frac{1}{\log_2(k+1)} - \frac{1}{\log_2(k+2)}. \tag{5.19}$$

Imagine a QREL with R_{gmax} documents relevant at the highest grade. According to the QREL these documents are equally likely to appear at ranks one through R_{gmax}, but have zero probability of appearing anywhere else. Therefore, in any ideal ranked list, the probability associated with one of these documents will be $P_{DCG}(k)$ for some k with $1 \leq k \leq R_{gmax}$. We define the probability of a *document* as the average probability of the ranks at which the document can appear in an ideal list. If R_g is the number of documents that are relevant at grade g, then for a document d such that $rel(d) = g$, the minimum rank for this document in an ideal list

$$k_{min} = \sum_{i=g+1}^{gmax} R_i, \tag{5.20}$$

i.e. after all of the documents with higher relevance grades, and the maximum rank is

$$k_{max} = k_{min} + R_g. \tag{5.21}$$

Then the probability associated with the document is

$$\begin{aligned} P(d) &= \alpha \frac{\sum_{i=k_{min}}^{k_{max}} \frac{1}{\log_2(i+1)} - \frac{1}{\log_2(i+2)}}{R_g} \\ &= \alpha \frac{\frac{1}{\log_2(k_{min}+1)} - \frac{1}{\log_2(k_{max}+2)}}{R_g}, \end{aligned} \tag{5.22}$$

where α is a normalizing constant. Note that the probability of non-relevant documents is *non-zero*, and that this definition can also be used for binary relevance.

RIC requires us to define a probability distribution over document pairs, whereas Equation 5.22 defines a probability for documents. To create the appropriate distri-

bution, we assume that each document in the pair is chosen independently,

$$P(d_i, d_j) = \beta P(d_i) P(d_j) \tag{5.23}$$

where β is a normalizing constant that ensures that $P(d_i, d_j)$ forms a distribution.

We define $RIC@k$ by normalizing by the ideal ranked list, as in nDCG, and computing mutual information with respect to the probability distribution defined in 5.23.

$$RIC@k(S) = \frac{I(R_S; Q)}{I(R_{ideal}; Q)} \tag{5.24}$$

5.3.2 Correlation with Existing Measures

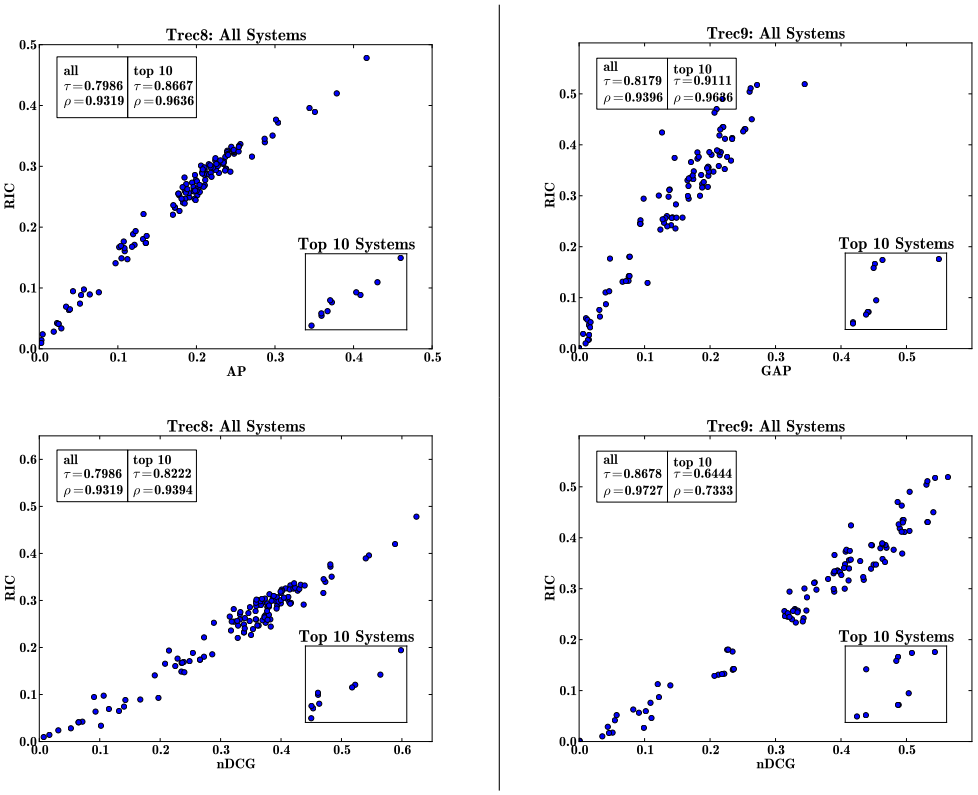


Figure 5.3: Correlation between RIC and AP (top) and nDCG (bottom). TREC 8 (left) uses binary relevance judgments. TREC 9 (right) uses graded relevance judgments.

Our measure is quite novel in its formulation, and makes many non-standard assumptions about information retrieval evaluation. Therefore it is necessary to validate experimentally that our measure prefers the same retrieval systems as existing measures. Note that for two evaluation measures to be considered compatible, it is sufficient that they rank systems in the same relative order; it is not

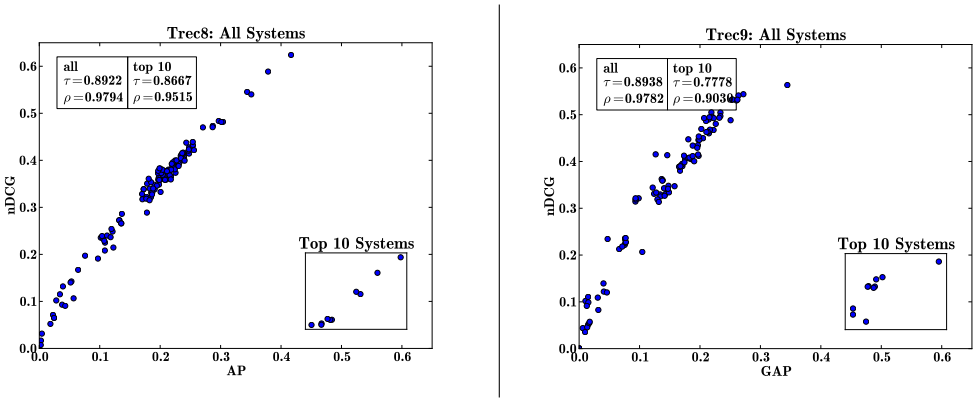


Figure 5.4: Correlation between (G)AP and nDCG. TREC 8 (left) uses binary relevance judgments. TREC 9 (right) uses graded relevance judgments.

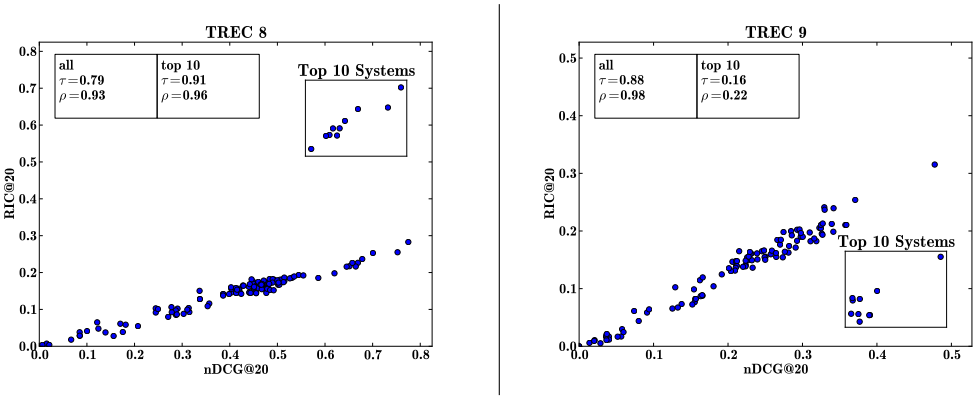


Figure 5.5: Correlation between RIC and nDCG at rank k=20. TREC 8 (left) uses binary relevance judgments. TREC 9 (right) uses graded relevance judgments.

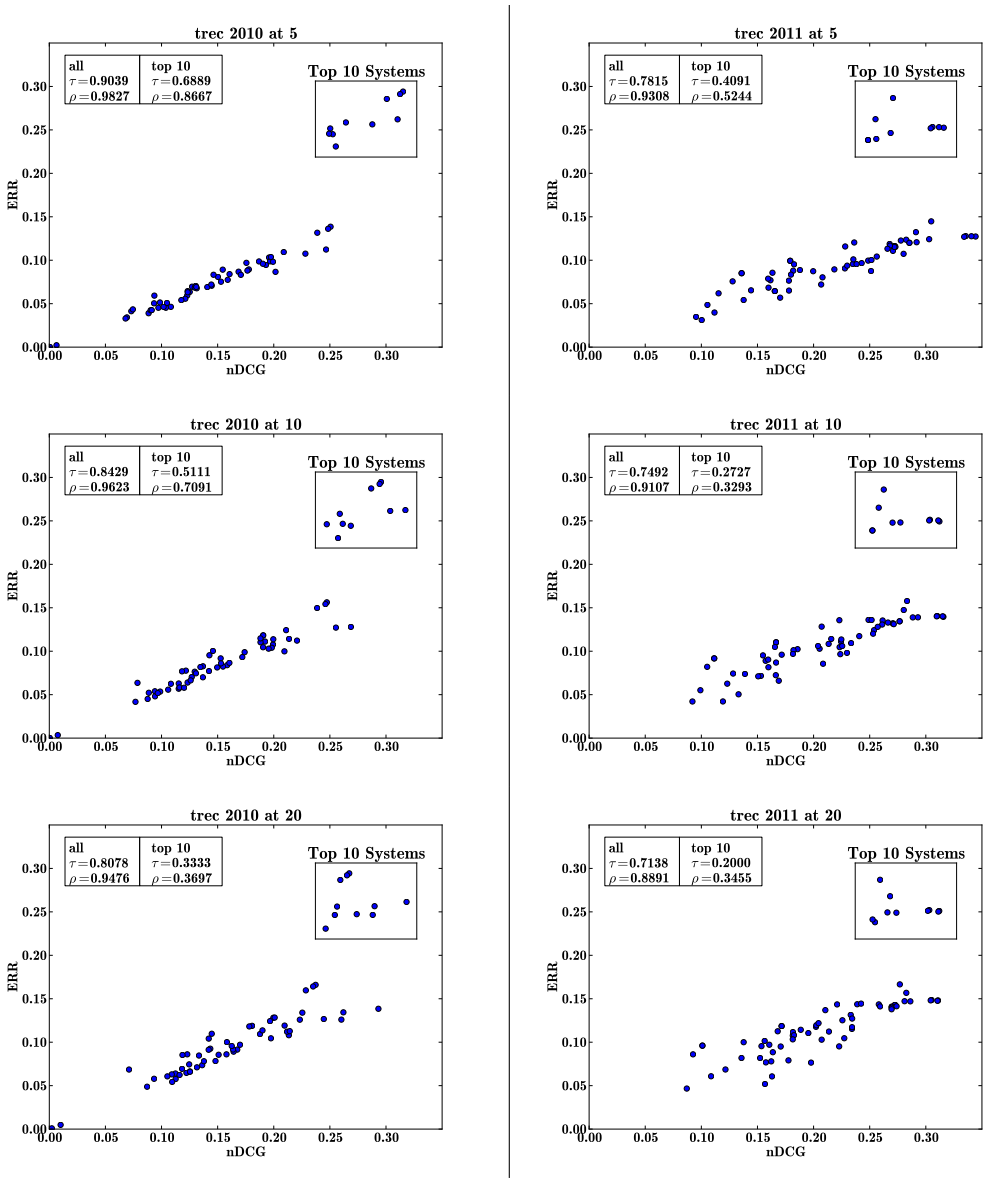


Figure 5.6: Correlation between nDCG and ERR at ranks 5, 10, and 20 on Trecs 2010 (left) and 2011 (right)

necessary that they always assign systems similar absolute scores. For example, a system's nDCG is often higher than its average precision.

To show that the recall-oriented version of RIC is consistent with AP and nDCG, we computed the RIC, AP, and nDCG[5] of all systems submitted to TRECs 8 and 9 [83, 84] (see Figure 5.3). TREC 8 uses binary relevance judgments. TREC 9 uses graded relevance judgments, requiring the use of graded average precision. Inset

[5]nDCG is computed at rank 1000.

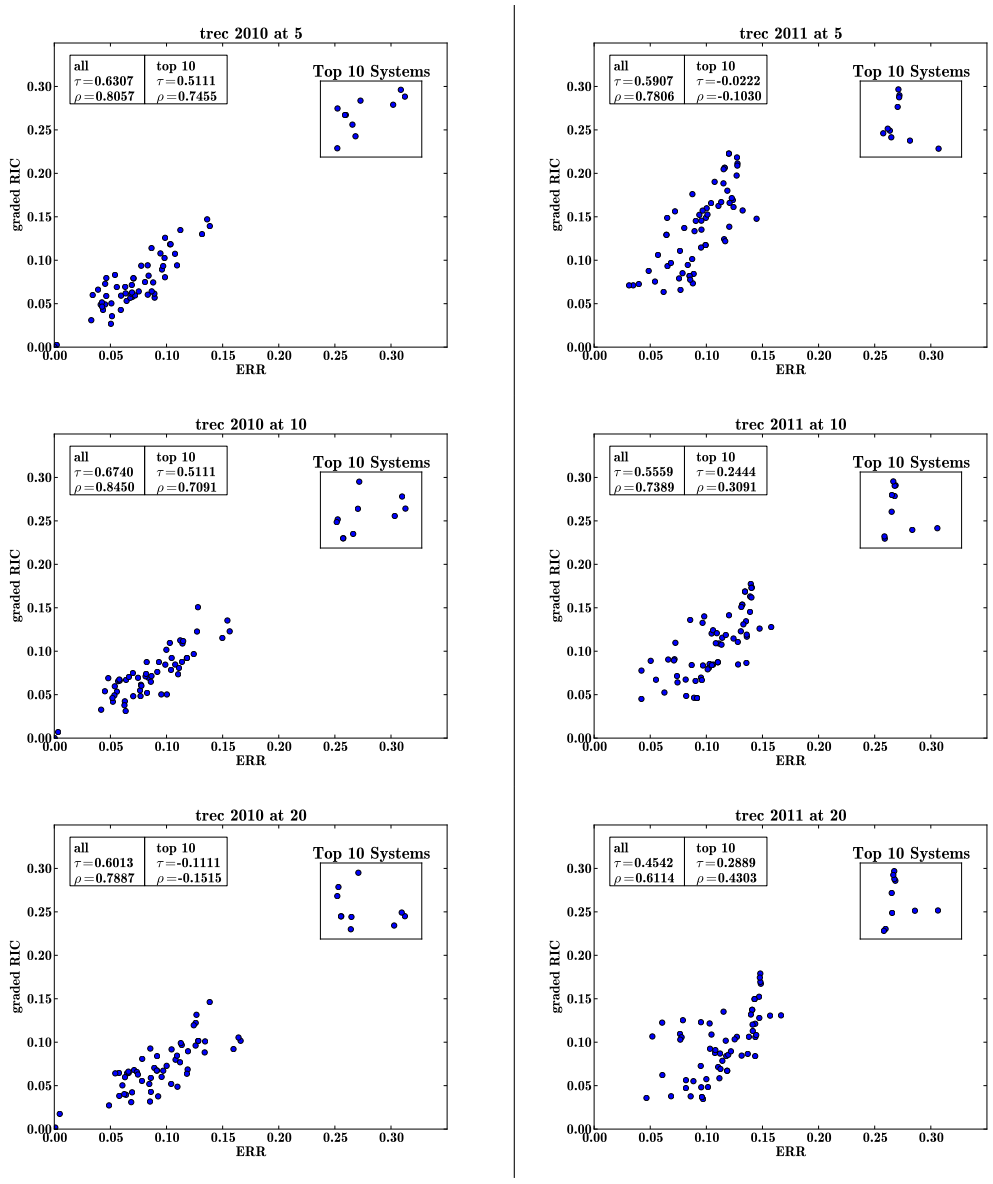


Figure 5.7: Correlation between RIC and ERR at ranks 5, 10, and 20 on Trecs 2010 (left) and 2011 (right)

into each plot is the output of the measures on the top ten systems as computed by the measure we are comparing RIC against. For each experiment, we report the Kendall's τ and Spearman's ρ [78] rank correlations for all systems, and for the top ten systems. With Kendall's τ values of at least 0.799 on all systems and 0.644 on top ten systems, the ranking of systems by RIC is still highly correlated with those of both AP and nDCG, although RIC is not as highly correlated with either AP or nDCG as AP and nDCG are with each other (see Figure 5.4).

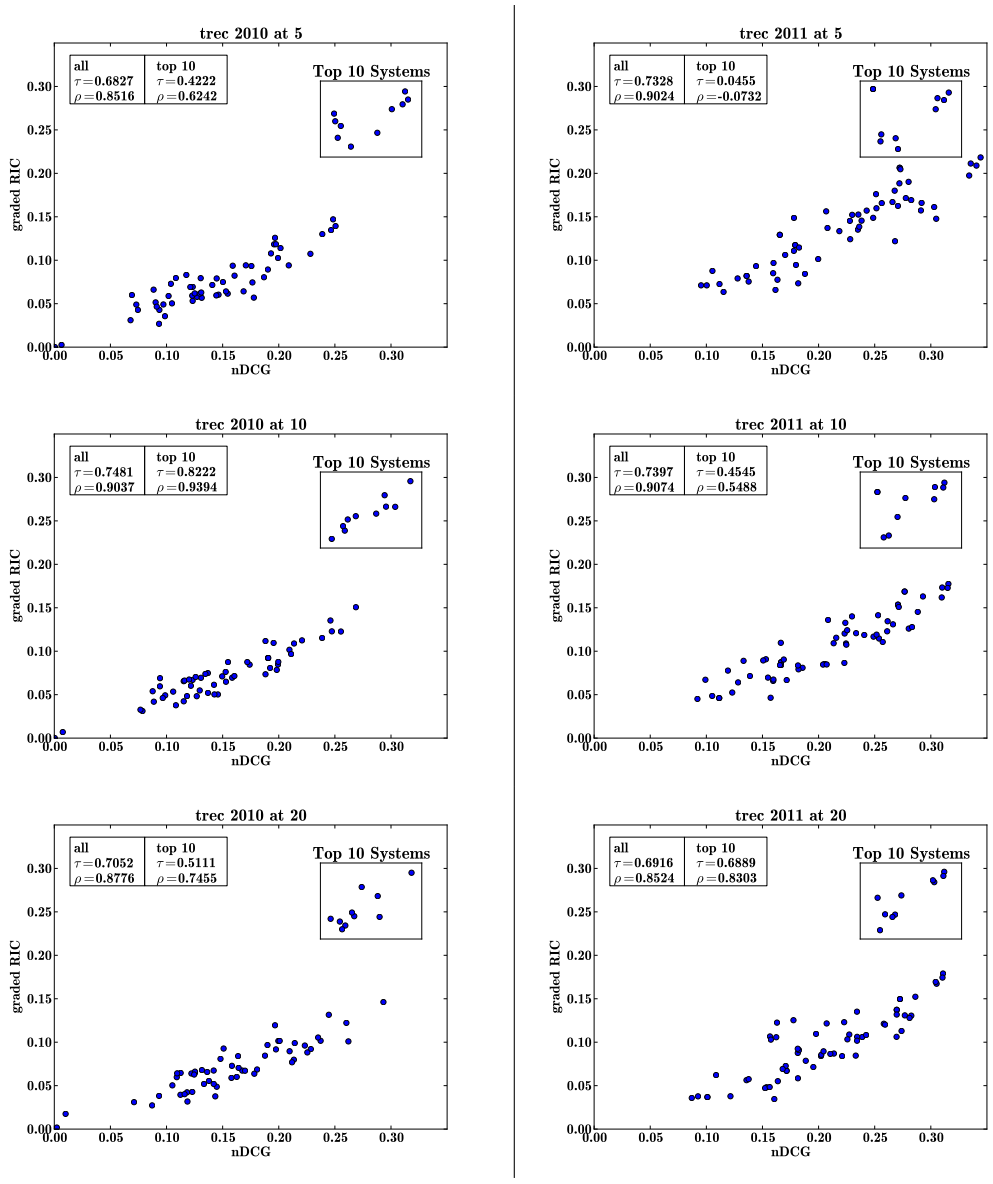


Figure 5.8: Correlation between RIC and nDCG at ranks 5, 10, and 20 on Trecs 2010 (left) and 2011 (right)

To show that the precision-oriented version, RIC@k, is consistent with existing measures, we computed the RIC and nDCG of all systems submitted to TRECs 8 and 9 at rank 20 (see Figure 5.5), as well as the RIC, nDCG and ERR of the systems submitted the the TREC 2010 ad hoc task and to TREC 2011 at ranks 5, 10, and 20 (see Figures 5.6, 5.7, and 5.8.) Considering TRECs 8 and 9, we see that, with a minimum τ of 0.79, RIC@20 is consistent with nDCG@20 overall. However, we do observe that, while RIC@20 and nDCG@20 agree about the best two systems,

and that the next 8 systems are better than the other systems, the two do not order systems 3 through 10 consistently, causing the τ of the top ten systems to fall all the way to 0.16.

Unfortunately, the rank correlation between measures is much lower on TRECs 2010 and 2011. The correlation between our measure and existing measures can be even smaller, with a minimum Kendall's τ of 0.45, which is halfway between perfect correlation and random noise. However, it is less clear on what is a "correct" ordering for TRECs 2010 and 2011 than on TRECs 8 and 9, as evidenced by the fact that the correlation between ERR and nDCG, two measures trusted by the community, can have a correlation as low as a Kendall's τ of 0.71. However the accuracy of our measure on these collections is still questionable. In future work, we believe it will be possible to improve upon this result by better estimating the underlying random variables, for example by employing smoothing or sampling techniques. However, we demonstrate in other sections that this level of accuracy is sufficient for myriad, novel, useful applications of our framework.

	TREC 8	TREC 9
(G)AP	0.716	0.648
nDCG	0.713	0.757
RIC	0.719	0.744

Table 5.2: Discriminative power of (graded) AP and nDCG vs. RIC for Recall-Oriented Experiments.

To further validate our measure, we also compute the *discriminative power* [60] of the various measures. Our results are displayed in Tables 5.2 and 5.3. As measured

	TREC 8	TREC 9
nDCG@20	0.665	0.643
RIC@20	0.656	0.451

	TREC 2010	TREC 2011
ERR@20	0.453	0.451
nDCG@20	0.529	0.516
RIC@20	0.114	0.275

Table 5.3: Discriminative power of nDCG vs. RIC for Precision-Oriented Experiments.

by discriminatory power, we see that, in a recall-oriented setting, RIC is at least as sensitive, if not more so, than AP and nDCG. Unfortunately, RIC is not very discriminative in a precision-oriented context.

5.4 Meta-Evaluation Application #2: Upper Bound on Metasearch

In Section 5.3, we defined an evaluation measure in terms of mutual information. One advantage of this approach is that collections of systems can be evaluated directly by considering the output of their random variables jointly, without their needing to be combined.The relevance information correlation for a collection of systems, denoted S_1 through S_n, can be defined as

$$RIC(S_1, \ldots, S_n) = I(R_{S_1}, \ldots, R_{S_n}; Q) \tag{5.25}$$

In this section, we will show that this produces a natural upper bound on metasearch performance that is consistent with other upper bounds appearing in the literature.

We compare our upper bound against those of Montague [51]. Montague describes metasearch algorithms as sorting functions whose comparators, as well as the documents to be sorted, are defined in terms of collections of input systems. By also using the QREL as input, these algorithms can estimate upper bounds on metasearch performance. These bounds range from the ideal performance that cannot possibly be exceeded by any metasearch algorithm, to descriptions of reasonable metasearch behavior that should be similar to the performance of any quality metasearch algorithm.

Montague defines the following upper bounds on metasearch:

1. Naive: Documents are sorted by comparison of relevance judgments, *i.e.* the naive upper bound is created by returning all relevant documents returned by any system in the collection above any non-relevant document. Relevant documents not retrieved by any system are not ranked.

2. Pareto: If document A is ranked above document B by all systems, then document A is considered "greater" than document B. Otherwise, the documents are sorted by comparison of relevance judgments.

3. Majoritarian: If document A is ranked above document B by at least half of the systems, then document A is considered "greater" than document B. Otherwise, the documents are sorted by comparison of relevance judgments.

We will compare our direct joint evaluation with these upper bounds, and several metasearch algorithms commonly used as baselines in the IR literature: the

	τ	ρ	RMSE
anz	0.221 / 0.147 / 0.221	0.330 / 0.213 / 0.327	0.481 / 0.533 / 0.516
condorcet	0.519 / 0.393 / 0.540	0.689 / 0.549 / 0.737	0.362 / 0.403 / 0.398
mnz	0.587 / 0.395 / 0.543	0.764 / 0.551 / 0.733	0.351 / 0.385 / 0.382
majoritarian	0.552 / 0.436 / 0.541	0.735 / 0.605 / 0.731	0.340 / 0.370 / 0.367
pareto	0.657 / 0.311 / 0.423	0.836 / 0.445 / 0.590	0.044 / 0.079 / 0.079
naive	0.788 / 0.579 / 0.661	0.931 / 0.762 / 0.849	0.039 / 0.093 / 0.096

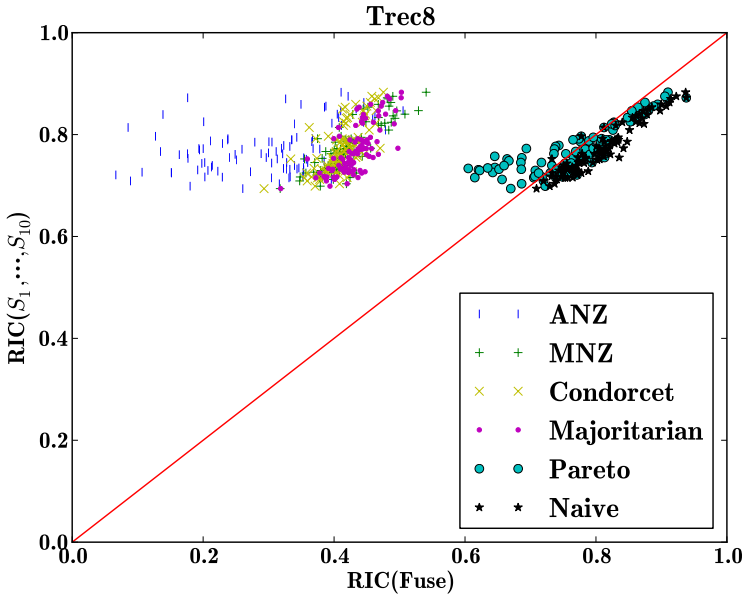
Table 5.4: Correlation between joint distribution and metasearch algorithms. Results are for Trec 8 / Trec 9 with binary relevance judgments / Trec 9 with graded relevance judgments.

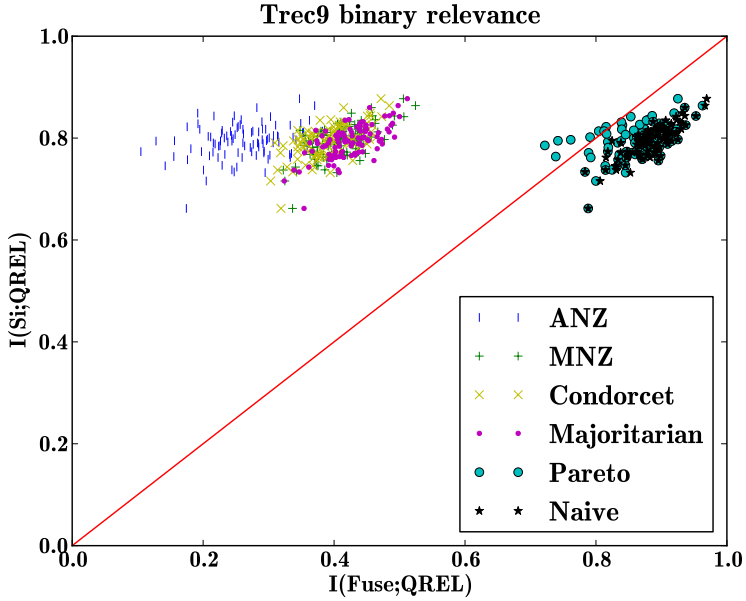
CondorcetFuse metasearch algorithm [52] and the Comb family of metasearch algorithms [70]. We examined the direct evaluation and metasearch performance of collections of ten randomly selected systems. Experiments were performed on TREC 8 and TREC 9 with both binary and graded relevance judgments.

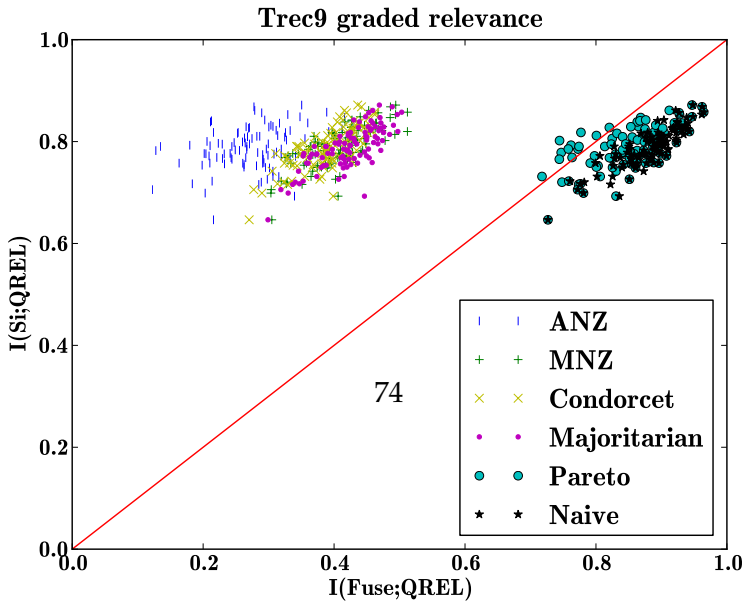
Figure 5.9 shows the RIC of the system output by a metasearch algorithm plotted against the joint RIC of the input systems, and Table 5.4 shows various measures of their correlation. Montague found that combANZ is inferior to CondorcetFuse and combMNZ, CondorcetFuse and combMNZ perform comparably to the Majoritarian bound, and the Naive bound is not appreciably better than the Pareto bound. If direct evaluation and the Naive bound are both reasonable estimates of the actual upper bound, then these results should be confirmed by Figure 5.9 and Table 5.4, as indeed they are. Note that there is almost no correlation between the joint evalution and the weakest metasearch algorithm, combANZ: combANZ does not approximate the upper bound on metasearch. The correlation improves as the quality of the metasearch algorithm improves, and it does so in a manner consistent with Montague. The correlations between the joint evaluation and the output of combMNZ, CondorcetFuse, and the Majoritarian bound are similar; while they are still biased as estimators, the correlation is beginning to approach monotonicity. Finally, with a root mean square error of 0.039 on TREC 8, the joint evaluation estimation of the upper bound is essentially identical to that of the Naive upper bound. If the Naive upper bound is a reasonable estimate of the upper bound on metasearch performance, then so is the joint evaluation of the input systems.

5.5 Summary

In this chapter, we developed a probabilistic framework for the analysis of information retrieval systems based on the correlation between a ranked list and the preferences induced by relevance judgments. Our framework is based on the choice of sample space, probability distribution, and random variables. Using this framework allows us access to information-theoretic tools which can be used to better understand information retrieval systems. For example, by considering rank correlation information-theoretically, which we call *information* τ, we can measure the correlation between two lists conditioned on a third. We demonstrated the value of this by showing that the correlation between diversity measures is primarily due to the underlying impact of performance. We also demonstrated that by appropriately defining our random variables we develop a new evaluation measure, *relevance information correlation*. By appropriately defining the probability distribution, this measure can be be used for precision- and recall-oriented experiments. Since this measure is computed information-theoretically, we show how it can be used to evaluate a collection of systems simultaneously, which provides a natural upper bound on metasearch performance.


Trec8
RIC($S_1, \cdots, S_{10}$)
RIC(Fuse)
ANZ
MNZ
Condorcet
Majoritarian
Pareto
Naive


Trec9 binary relevance
I(Si;QREL)
I(Fuse;QREL)
ANZ
MNZ
Condorcet
Majoritarian
Pareto
Naive


Trec9 graded relevance
I(Si;QREL)
I(Fuse;QREL)
ANZ
MNZ
Condorcet
Majoritarian
Pareto
Naive

Chapter 6

Information Difference

Imagine that you are attempting to improve an existing ranker. On what basis do you decide whether or not your changes are beneficial? One typically evaluates both systems on a number of queries, and measure the difference in average performance. If one system outperforms the other, whether you have made an improvement is clear. But what happens when the systems perform similarly? It could be that your new system is essentially unchanged from your old system, but it is also possible that the two systems chose highly different document sets that just happened to have very similar evaluation scores. In the latter case, it may be possible to create a new, better system based on a combination of the two existing systems.

We propose a measure of the magnitude of the difference between systems in their ranking of documents for which we have relevance information, rather than the magnitude of the difference between their performance. We denote this quantity as the **information difference** between systems. We define and experimentally validate information difference in Section 6.1. In Section 6.2, we demonstrate the utility of information difference as a meta-evaluation tool by comparing the relative impact of retrieval models and parameter tuning. In Section 6.3, we show another practical application of information difference, namely that it can be used to find the best systems to merge for metasearch.

6.1 Definition

In this section, we define our notion of information difference, and then demonstrate empirically that it can be used to determine whether two systems with similar performance have similar *behavior*, *i.e.* that they rank documents consistently.

Our definition of information difference is inspired by the Boolean algebra sym-

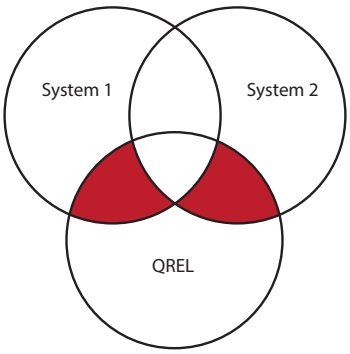


Figure 6.1: Information difference corresponds to the symmetric difference between the intersections of the systems with the QREL in information space (red portion of the Venn diagram).

metric difference operator as applied to information space (see Figure 6.1).

$$id(S_1, S_2) = I(R_{S_1}; Q \mid R_{S_2}) + I(R_{S_2}; Q \mid R_{S_1}) \tag{6.1}$$

with Q and R defined as in Equations 5.16 and 5.17 respectively, and using a uniform distribution over all pairs of documents with different relevance grades. We also define $id@k$ by using the probability distribution described in Equation 5.23.

As a preliminary validation of information difference, we analyzed pairs of systems submitted to TREC 8, selected at random (see Figure 6.2). The x-axis shows the magnitude of the change in AP, and the y-axis shows the information difference. The two are roughly correlated. This corresponds with our intuition that, in general, systems that rank documents similarly should be expected to have similar performance.

To demonstrate the utility of information difference, we demonstrate that it can be used to detect whether systems are similar. As a proxy for similarity, we sort a collection of systems by performance, and separate them into twenty equal-sized bins. By definition, each bin contains systems with small differences in performance. We will consider two systems within the same bin to be "similar" if they were submitted by the same research group. It is reasonable to assume that the majority of these systems were different instantiations of the same underlying technology, although there will be many instances where this is not the case at all. Within each bin, we compare each pair of systems to determine which pairs are similar according to our proxy. For example, Table 6.1 shows the pairs of systems submitted to TREC 8 that had the smallest information difference. When the system pairs were sorted by their information difference, the first 27 pairs meet our proxy for similarity. This is not the case when we sort by $|\Delta$ AP$|$.

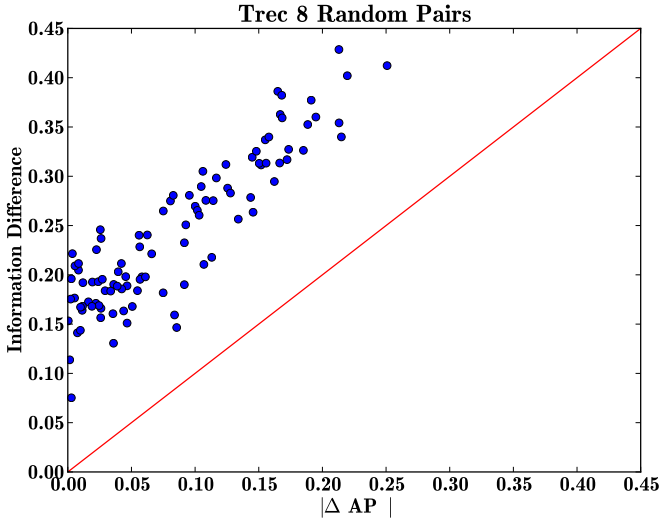


Figure 6.2: Scatter plot of information difference and the magnitude of change in AP of random pairs of TREC 8 systems.

Rank	System 1	System 2	id	$\|\Delta$ AP$\|$
1	**blue**UB99T	**blue**UB99SW	0.010	0.005
2	**blue**unc8al32	**blue**unc8al42	0.012	0.002
3	**blue**fub99tt	**blue**fub99tf	0.017	0.000
4	**blue**nttd8al	**blue**nttd8alx	0.023	0.002
5	**blue**ibmg99a	**blue**ibmg99b	0.027	0.012
		⋮		
28	**blue**isa25t	**red**cirtrc82	0.084	0.004
29	**blue**CL99SD	**blue**CL99SDopt2	0.086	0.000
30	**blue**ok8amxc	**blue**ok8alx	0.086	0.006
31	**blue**tno8d4	**red**MITSLStd	0.088	0.016
32	**blue**uwmt8a2	**blue**uwmt8a1	0.089	0.002

Table 6.1: The systems from TREC 8 were binned by average precision. Information difference and Δ AP were computed for all system pairs within each bin. Sorting by information difference, the first 27 pairs match our proxy for similarity.

To determine the quality of information difference as a similarity classifier, we compare it's ROC curve to those of the following baselines:

1. Mutual Information: $I(R_{S_1}; R_{S_2})$—Computing information difference requires relevance judgments. This comparison allows us to measure how much these judgments increase our ability to classify systems. Recall that, in the base case where both systems totally order the same set of documents, this is equivalent to Kendall's τ. Note that since we are not using relevance judgments,

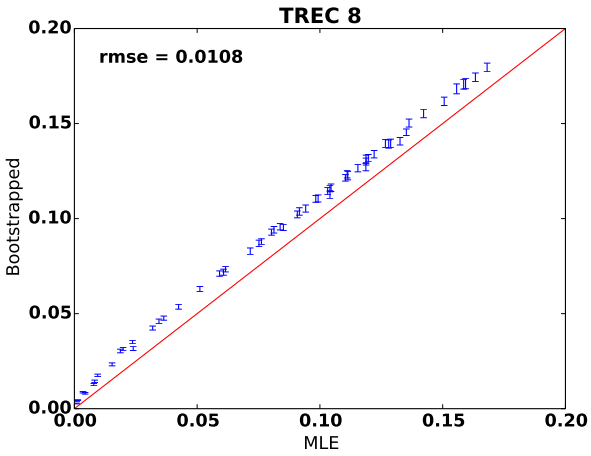


Figure 6.3: The maximum likelihood estimate versus a bootstrapped estimate of the mutual information between 60 pairs of systems submitted to TREC 8. Systems are truncated at rank 100. The bootstrapped estimate is computed over 100 samples, each of which consists of 100 pairs of documents chosen at (uniformly) random.

we cannot truncate our ranked lists at the last relevant document. We found that, in recall-oriented experiments, this could produce a quite large number of additional document pairs, greatly increasing runtime. Fortunately, we found simple bootstrap estimates to be highly accurate (see Figure 6.3).

2. Jaccard Coefficient: $\frac{S_1 \cap S_2}{S_1 \cup S_2}$—This is a set-based measure, rather than a list-based measure, *i.e.* it determines whether two systems ranked the same set of documents, independent of order. This comparison allows us to determine whether document *order* is necessary to classify systems, or whether document *selection* alone is sufficient.

We performed our experiments on TRECs 8 & 9 at ranks 20 and 1000, as well as the systems submitted to the TREC 2010 ad hoc and diversity tasks, and TREC 2011. For each collection, we compute the ROC curves of information difference, mutual information and Jaccard coefficient when used to classify systems as similar, as well as the average Jaccard coefficient between all pairs of systems that were compared. Figure 6.4 shows all computed AUC scores as a function of average Jaccard coefficient. Figure 6.5 shows the ROC curves of each classifier on each collection. We observe that when the average Jaccard coefficient is high, similarity detection becomes trivial. This is plausible. If the average Jaccard coefficient is high, then the majority of documents chosen by the majority of the systems are the same. Any deviation from this set of common documents can be used to classify systems accurately, *i.e.* if two systems each chose the *same* rare document, then they

are very highly likely to be similar. However, when the average Jaccard coefficient is lower, the problem becomes more difficult. For example, on TREC 8 at rank 20, with an average Jaccard coefficient of 0.687, the Jaccard coefficient classifier has an AUC of 0.82, and our mutual information classifier has an AUC of 0.8. However, when we take both ranking information and relevance judgments into account, our information difference classifier is able to achieve an AUC of 0.963. From this we conclude that information difference is a better similarity classifier than our baseline models of mutual information and Jaccard coefficient.

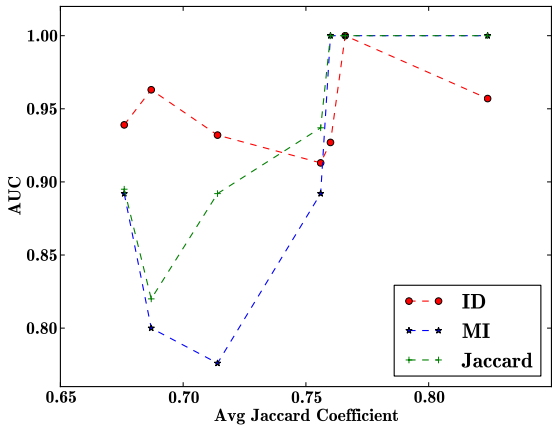


Figure 6.4: AUC as a function of average Jaccard coefficient. When the average Jaccard coefficient is high, similar systems are easy to detect. When the average Jaccard coefficient is smaller, only information difference is able to detect similar systems.

6.2 Application #1: Quantifying the Impact of Parameter Tuning and Retrieval Model Selection

At the heart of a search engine is a retrieval model, a function that takes a document and a query and returns a number which the search engine uses to rank documents. Different retrieval models have can have many theoretical differences, but they also tend to have a large number of theoretical similarities as well. It would be interesting to know which has more impact, the theoretical similarities or the differences. Do different models tend to behave similarly? The usual way to approach this question would be to observe their performance. In practice, well-tuned, well-implemented retrieval models tend to have similar performance. Does this imply that different models are similar? In this section, we use information difference to show that different retrieval models really are simpler by demonstrating that the choice of retrieval model has smaller impact on the *behavior* (rather

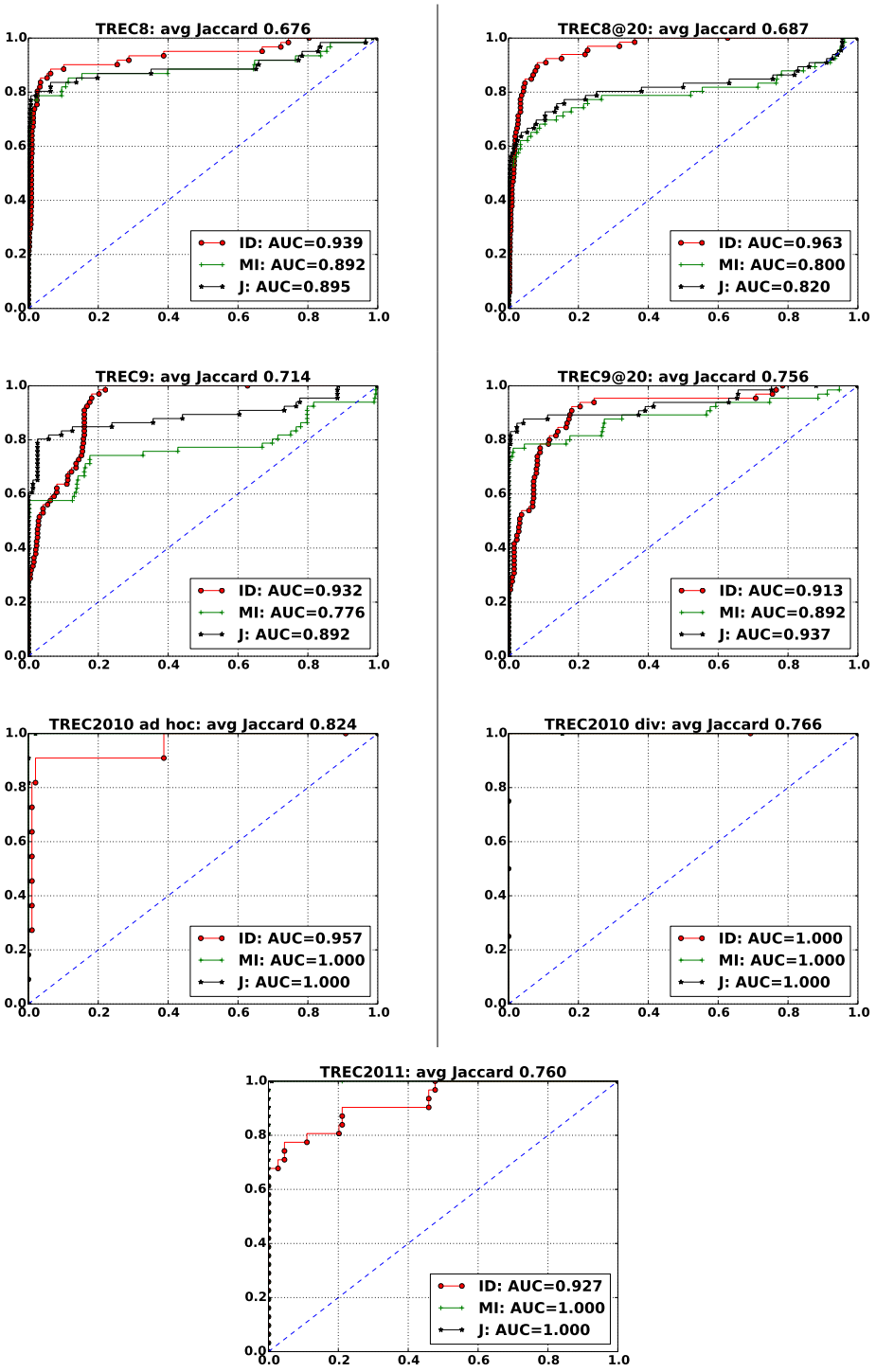


Figure 6.5: ROC curves of information difference, mutual information, Kendall's τ, and Jaccard coefficient as similarity classifiers. When the average Jaccard coefficient is high, similar systems are easy to detect. When the average Jaccard coefficient is smaller, only information difference is able to detect similar systems.

than the *performance*) of a search engine than does implementation details such as parameter tuning. We discuss the retrieval models used in our experiments, and their similarities, in Section 6.2.1. In Section 6.2.2 we present our experiments.

6.2.1 Retrieval Models

i	a term
Q	a query
D	a document
f_i	the frequency of the term i in the document
qf_i	the frequency of the term i in the query
cf_i	the frequency of the term i in the collection
N	the number of documents in the collection
n_i	the number of documents in the collection that contain the term i
dl	the length of document D
$avdl$	the average length of the documents in the collection
C	the number of word occurrences in the collection
C_i	the number of times the term i appears in the collection
$\overset{rank}{=}$	rank equivalent, *i.e.* two functions are *rank equivalent* if they induce the same partial ordering on their domain

Table 6.2: Notation used in retrieval models.

In this section, we briefly describe the retrieval models we are analyzing and their theoretical similarities. The first model we discuss is BM25, which we will describe, fairly accurately, as an empirically-derived $tf.idf$ vector space-like model. $tf.idf$ models treat documents and queries as elements of a vector space whose basis is formed by the collection's vocabulary. A document or query is represented as a vector whose components are comprised of weights computed by multiplying the term frequency (tf)—the contribution of the term based on the frequency of the term in the document, by the inverse document frequency (idf)—the contribution of the term based on the scarcity of the term in the collection. The document's score is computed by taking something like an inner product between the query vector and the document vector. Using notation described in Table 6.2, the "inner product" computed by BM25 is:

$$BM25(d,q) = \sum_{t \in q} \log \frac{1}{\frac{n_i - 0.5}{N - n_i + 0.5}} \cdot \frac{(k_1+1)f_i}{K + f_i} \cdot \frac{(k_2+1)qf_i}{k_2 + qf_i} \tag{6.2}$$

where

$$K = k_1 \left((1-b) + b \frac{dl}{avdl} \right). \tag{6.3}$$

Common choices for the free parameters are to set k_1=1.2, $b = 0.75$, and k_2=100.

Language models (LM) are generative models of text widely used throughout natural language processing. The most simple models are unigram or bag-of-word models, which are simply probability distributions over the collection's vocabulary, where terms are assumed to be independent. In this model, we build a language model for each document and then rank documents by *query likelihood*, how likely the query is given the document.

$$P(Q \mid D) = \prod_{i=1}^{n} P(q_i \mid D) \tag{6.4}$$

The difficulty lies in estimating $P(q_i \mid D)$. If we simply use the maximum likelihood estimate $P(q_i \mid D) \approx \frac{f_i}{dl}$ then the probability of any document that does not contain all of the query terms would be zero. A more fundamental issue is the problem of data sparsity. It is difficult to properly estimate probabilities from as few examples as can be gleaned from a single document. Therefore, we smooth our probability estimates by supplementing the evidence provided by the document with the evidence provided by the collection as a whole. One common model used for smoothing is known as Jelinek-Mercer smoothing. In Jelinek-Mercer smoothing, we interpolate linearly between the prevalence of the term in the document and the prevalence of the term in the collection as a whole:

$$P(q_i \mid D) \approx (1-\lambda) \frac{f_i}{dl} + \lambda \frac{C_i}{C} \tag{6.5}$$

where λ controls the weight given to the evidence provided by the document as opposed to the collection. In practice, to increase accuracy, we usually compute the log-likelihood of the query. Therefore, applying smoothing to Equation 6.4 and taking the log of each side, we get:

$$\log P(Q \mid D) = \sum_{i=1}^{n} \log \left((1-\lambda) \frac{f_i}{dl} + \lambda \frac{C_i}{C} \right). \tag{6.6}$$

The smoothing model that we investigate in Section 6.2.2 is known as Dirichlet smoothing. In Dirichlet smoothing, we allow the weight assigned to the document

to be a function of the documents length:

$$\lambda = \frac{\mu}{dl + \mu}. \tag{6.7}$$

The ranking formula for a language model with Dirichlet smoothing is

$$\log P(Q \mid D) = \sum_{i=1}^{n} \log \frac{f_i + \mu \frac{C_i}{C}}{dl + \mu}. \tag{6.8}$$

A common range for μ is between 1000 and 2000.

While the intuition behind *tf.idf* models such as BM25 and language models is quite different, their performance tends to be quite similar in practice. The following argument, due to Croft *et al.* [35], shows that in fact language models with Jelinek-Mercer smoothing are quite similar to *tf.idf* models.

$$\begin{aligned}
\log P(Q \mid D) &= \sum_{i=1}^{n} \log(1-\lambda)\frac{f_i}{dl} + \lambda\frac{C_i}{C} \\
&= \sum_{i:\ f_i>0} \log(1-\lambda)\frac{f_i}{dl} + \lambda\frac{C_i}{C} + \sum_{i:\ f_i=0} \log \lambda\frac{C_i}{C} \\
&= \sum_{i:\ f_i>0} \log \frac{(1-\lambda)\frac{f_i}{dl} + \lambda\frac{C_i}{C}}{\lambda\frac{C_i}{C}} + \sum_{i=1}^{n} \log \lambda\frac{C_i}{C} \\
&\overset{rank}{=} \sum_{i:\ f_i>0} \log \frac{(1-\lambda)\frac{f_i}{dl}}{\lambda\frac{C_i}{C}} + 1
\end{aligned} \tag{6.9}$$

where in the third line we add $\sum_{i:\ f_i>0} \lambda\frac{C_i}{C}$ to the second term and subtract it from the first. This demonstrates that the similarity between a document and a query is some function that is similar to an "inner product" between term frequency and inverse document frequency vectors, as in a *tf.idf* models.

Our final class of retrieval models is the divergence from randomness (DFR) model, which we present here very briefly. We direct the interested reader to Amati [2] for more details. DFR models attempt to recognize those documents in which query terms appear disproportionately from what would be expected given the collection as a whole. A DFR is specified by the choice of 1) a term frequency normalization factor, 2) the "first normalization" which functions similarly to smoothing

in language models, and 3) a model of randomness. In general, a DFR is defined as

$$DFR(D,Q) = \sum_{i \in Q} \frac{qf_i}{\max(qf)} \cdot w(i,D) \tag{6.10}$$

where $w(i,D)$ represents the extent to which the term i differs from what it should be "randomly" in document D. In all of our DFR instantiations, we use the following term frequency normalization:

$$tfn = f_i \cdot \log\left(1 + c \cdot \frac{avdl}{dl}\right) \tag{6.11}$$

where c is a free parameter commonly set to 1.

In this work we focus on two DFR models. The first is In$_e$B2, which we use for recall-oriented evaluation. In$_e$B2 uses an Inverse Expected Document Frequency model of randomness and a ratio of two Bernoulli's processes as the first normalization.

$$In_eB2 : w(i,D) = \frac{C_i + 1}{n_i \cdot (tfn + 1)} \left(tfn \cdot \log \frac{N+1}{n_e + 0.5}\right) \tag{6.12}$$

where

$$n_e = N(1 - (1 - n_i/N)^{C_i}). \tag{6.13}$$

The second model is PL2, which we use for evaluation at rank 20. PL2 uses a Poisson approximation as a model of randomness and a Laplace model for the first normalization.

$$PL2 : w(i,D) = \frac{1}{tfn+1}\left(tfn \cdot \log \frac{tfn}{\lambda} + (\lambda tfn) \cdot \log e + 0.5 \cdot \log(2\pi \cdot tfn)\right) \tag{6.14}$$

where λ is the variance and mean of a Poisson distribution. It is given by $\frac{C_i}{N}$. Note that C_i is much smaller than N.

While these models are also theoretically dissimilar from those already discussed, it is possible to derive BM25, along with common choices for parameters, from a particular instantiation of the DFR model. As the derivation is not particularly enlightening, we again direct the interested reader to Amati [2] for details.

6.2.2 Experiments

To maximize the effect of retrieval models, we utilize highly simple search engines, *i.e.* without query expansion, pseudo-relevance feedback, etc. We build these models using a standard, state-of-the-art search engine, in this case the Terrier search engine [55]. We will analyze these systems as they are run over TRECs 8 and 9,

since they are relatively well-judged, and therefore more reusable than TRECs 2010 and 2011. We compare the previously discussed models,

1. a query-prediction language model (LM) with Dirichlet smoothing,
2. BM25, and
3. Divergence from Randomness (DFR) models,

across a range of 21 different, evenly spaced, "reasonable" parameter values (see Figure 6.6), and the "best" of these observed parameter values which achieves the maximum performance (see Table 6.3).[1]

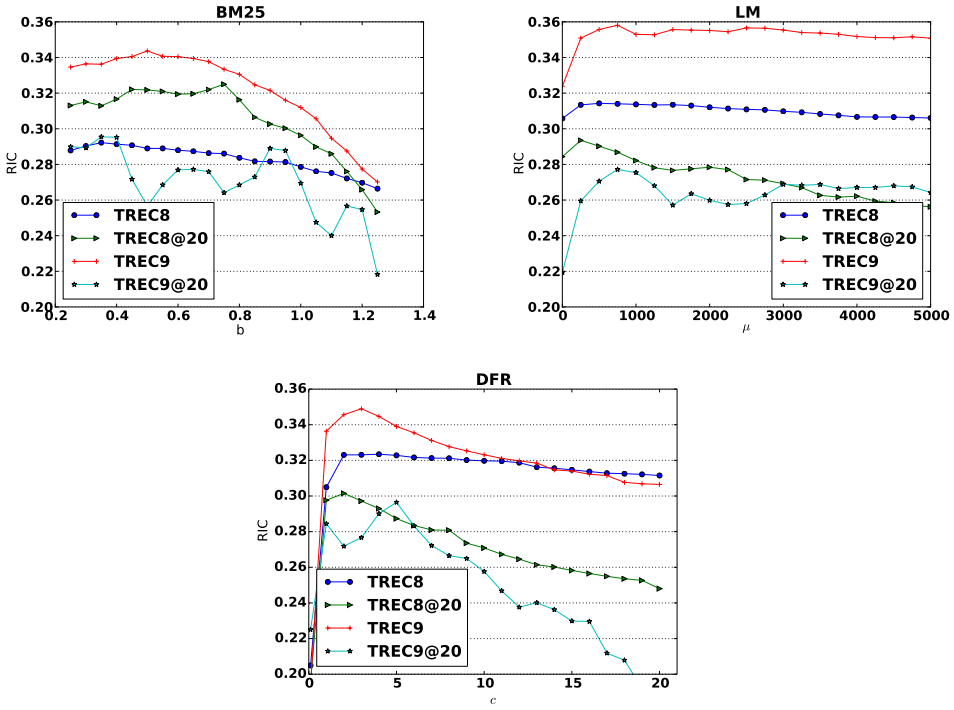


Figure 6.6: Performance as a function of retrieval model parameters—For BM25 we tune the parameter b. For LM we tune the parameter μ. For DFR we tune the parameter c.

As we can see from Table 6.3, the three models perform relatively consistently with one another. We also observe (Figure 6.6) that, with the exception of BM25 and DFR at rank 20, each model has reasonably consistent performance with itself as parameters are tuned. Therefore, at this point, using only performance delta as

[1]Our goal is to compare search engines during the *evaluation phase*, when relevance assessments have already been used. Therefore we employ the *best* parameters for *these* queries, rather than *optimal* parameters applicable to *future* queries.

RIC	TREC8	TREC8@20	TREC9	TREC9@20
BM25	0.292	0.325	0.344	0.295
LM	0.314	0.294	0.358	0.277
DFR	0.323	0.302	0.349	0.296

Table 6.3: Best observed performance of standard retrieval models.

our guide, it seems that all models are consistent, independent of how well tuned they are.

ID		TREC8	TREC8@20	TREC9	TREC9@20
LM	DFR	0.076	0.110	0.088	0.122
LM	BM25	0.068	0.149	0.092	0.127
BM25	DFR	0.077	0.091	0.108	0.056

Table 6.4: Information difference between standard retrieval models with "best" parameters.

Using information difference we can now measure the similarity of these models are in terms of their *behavior*, rather than their *performance*. Recalling that a small information difference implies a high degree of similarity, consider Table 6.4, which shows the information difference between the best performing retrieval models. As a point of reference, using an information difference threshold of 0.1 would have achieved a roughly 92% average accuracy on the classification task described in Section 6.1. Therefore, it is quite likely that information difference would have failed in this case and considered these retrieval models to be the same system.

Now we consider the difference between instantiations of a single model. We instantiate each model with all 21 parameter values and compute the information difference between each pair of instantiations. Figure 6.7 shows cumulative histograms of the information difference between all $\binom{21}{2}$ pairs of these model instantiations. These histograms show that there is far more difference in behavior within a model across parameterizations than their is across models with the best parameterization. For example, consider the largest information difference between models, which is between our language model and BM25 on TREC 8 at rank 20. The information difference of 0.149 is smaller than roughly 40% of the pairs of BM25 models. The smallest information difference is between BM25 and DFR on TREC 9 at rank 20. The information difference of 0.056 is smaller than all but roughly 45% of the pairs of LM models. Our information difference classifier is once again likely to fail and consider these systems as different, even though they are merely

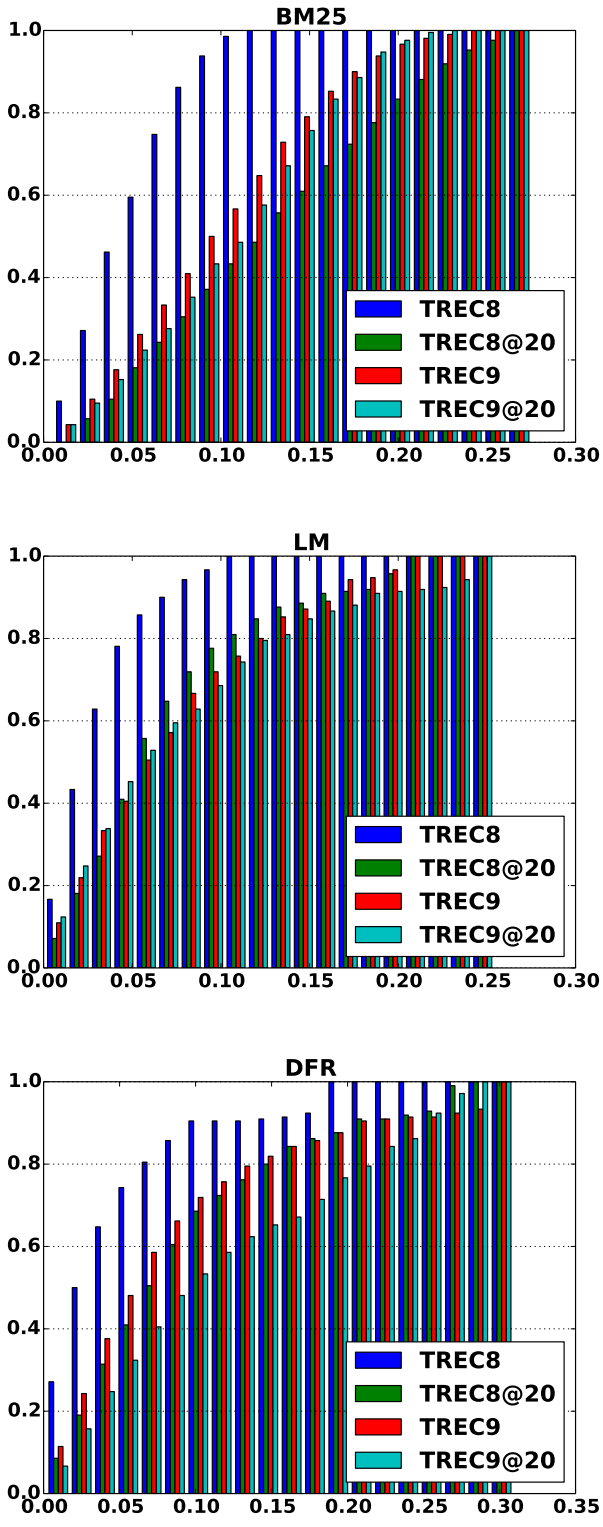


Figure 6.7: Cumulative histograms of information difference between parameterizations of a standard retrieval model.

different instantiations of the same retrieval model. Since our non-performance, behaviorally based classifier would have considered different retrieval models to be the same system and different instantiations of the retrieval models to be different, we conclude that parameter tuning actually has a larger effect on the *behavior* of search engines than does the underlying retrieval model.

6.3 Application #2: Selecting Systems for Metasearch

When performing metasearch, the common wisdom is that one should fuse the best systems that are the most dissimilar from one another. However, this is a difficult assumption to test: how does one measure dissimilarity? Even if one could measure dissimilarity, how would one determine which are the best systems? If you had access to relevance judgments, there would be no need to perform metasearch, and without relevance judgments how would you determine which systems are best?

We do know that the choice of systems has a profound impact on metasearch performance. For example, it is known [81] that the best results are achieved by fusing a small number of high quality systems. Consider Figure 6.8, which shows the average precision of metasystems created using CombMNZ. New systems are added in order, from best to worst as measured by AP, and at random. We can see that when added in best to worst order the initial improvement is quite large for the first five systems, after which performance steadily decreases to well below the performance of the best system pre-fusion. When added in a random order, performance is never as high as that of the best pre-fusion systems.

In this section, we demonstrate that, by using our information-theoretic framework to measure the similarity between systems, we can intelligently select systems to fuse. We do so without utilizing relevance judgments, achieving performance improvements over fusing all systems, and significant performance improvements over choosing at random. We describe our selection methodology in Section 6.3.1 and present our experimental results in Section 6.3.2.

6.3.1 Methodology

In order to facilitate the description of our methodology, we begin by assuming that we have access to relevance judgments. In Section 6.3.2, we will show that in this case our methodology does not outperform simply fusing the best systems. However, determining which systems are best is equivalent to evaluation, and this cannot be done without acquiring costly relevance judgments.

Given relevance judgments, measuring system quality and system similarity can be done using standard evaluation tools and our framework, specifically infor-

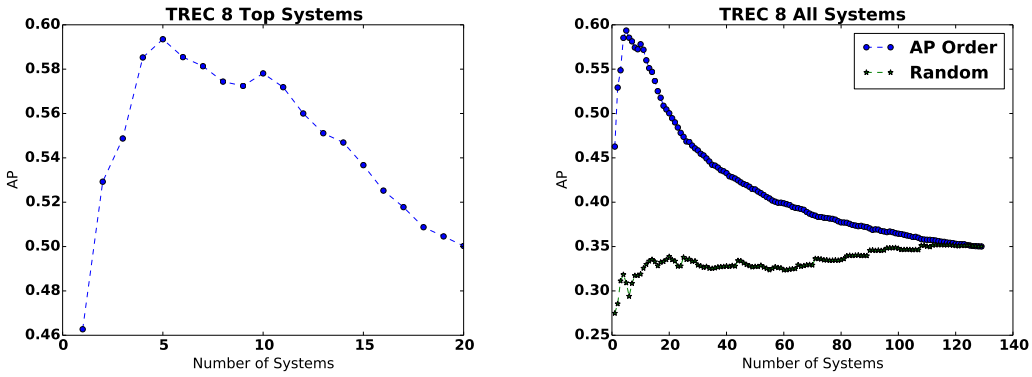


Figure 6.8: Performance of CombMNZ fusion algorithm as systems are added in AP-sort order. The plot on the left focuses on the 20 best systems. The plot on the right shows all submitted systems added in both AP-sort order, and in a randomly selected arbitrary order. Notice that, in AP-order, performance initially increases sharply as the best systems are fused, and then suffers as additional systems are added. When added in a random order, performance starts out weak and gradually increases until reaching a steady state.

mation difference. The question is how to merge the two. We follow the framework utilized by Zuccon *et al.* [98] to diversify systems. Zuccon *et al.* interpreted the diversification problem within the framework of *Facility Location Analysis* (FLA) from Operations Research [42]. In FLA, one is given a set of customer "locations" D and is tasked with finding the (in our case) subset $S \subset D$ of k "facilities" that optimizes some objective incorporating the "cost" associated with each facility and the "distance" between the facilities and their customers.

Interpreted in terms of FLA, the Maximal Marginal Relevance (MMR) method [16] becomes a greedily approximate solution to the *Obnoxious Facility Dispersion* (OFD) problem. Imagine that one wishes to determine where to locate k nuclear-waste storage facilities. One wishes to find the k sites that will have the minimum cost to operate yet will also be the furthest from the general population. This metaphor is rather tortured for us, since our facilities and our customers are drawn from the same set of documents, but the intuition holds. The MMR method dictates that we select the documents that are most relevant (minimal operational cost) and least similar (maximal distance from the general population). This can be formalized into the following heuristic: given a set of previously retrieved documents S and a

candidate document d,

$$h(d, S) = \lambda r(d) + (1 - \lambda) \min_{d' \in S} w(d, d')$$

or

$$h(d, S) = \lambda r(d) - (1 - \lambda) \max_{d' \in S} s(d, d') \tag{6.15}$$

where r is some notion of document quality, s is some notion of document similarity, and w is some notion of distance between documents. The similarity forumlation of Equation 6.15 is exactly the ranking formula used in MMR. Given this objective function, Algorithm 6.1 performs a greedy best-first search to find an approximate solution.

Algorithm 6.1 Greedy Best-First Search for Obnoxious Facility Dispersion

```
1: procedure OBNOXIOUSFACILITYDISPERSION(D, k, r, h)
2:     d_1 = argmax_{d∈D} r(d)
3:     S ← d_1
4:     for i = 2, ..., k do
5:         d* = argmax_{d∈D\s} h(d, S)
6:         S ← S ∪ {d*}
7:     return S
```

The primary contribution of Zuccon *et al.* is the observation that diversification is better modeled in terms of the *Desirable Facility Placement* (DFP) problem. Imagine that you wish to decided where to place k hospitals. You wish to find the k locations that:

1. minimize the total cost of maintaining those facilities, and
2. minimize the distances from the customer locations to their closest facilities.

This is easiest to describe in terms of minimizing cost and distance. However, we find it easiest to formulate our heuristic in terms of maximizing benefits and similarities instead.

$$f(S) = \lambda \sum_{d \in S} r(d) - (1 - \lambda) \sum_{d \in D \setminus S} \left(\min_{d' \in S} w(d, d') \right)$$

or

$$f(S) = \lambda \sum_{d \in S} r(d) + (1 - \lambda) \sum_{d \in D \setminus S} \left(\max_{d' \in S} s(d, d') \right) \tag{6.16}$$

where r, s, and w are again notions of document quality, similarity, and distance. Observe that here our "customers" are the non-retrieved documents, in contrast to

OFD, where the "customers" were the previously retrieved documents.

Finding the set of documents S that maximizes our heuristic f is NP-Complete (consider a reduction from *e.g.* the set cover problem). Algorithm 6.2 performs a greedy local search for an approximate solution. Note that, unlike the algorithm

Algorithm 6.2 Greedy Local Search for Desirable Facility Placement

```
1:  procedure DesirableFacilityPlacement(D, k, r, f)
2:      for i = 1, ..., k do
3:          d* = argmax_{d∈D\S} r(d)
4:          S ← S ∪ {d*}
5:      repeat
6:          for d ∈ S do
7:              for d' ∈ {D\S} do
8:                  S' ← (S\{d}) ∪ {d'}
9:                  if f(S') > f(S) then
10:                     S ← S'
11:     until S does not change
12:     return S
```

for OFD, that if we search for S_1 of size k and S_2 of size $k+1$ that $|S_1 \cap S_2| \leq k$, *i.e.* it is possible that S_1 and S_2 will contain an entirely different set of documents.

Given the presence of relevance assessments, it is clear how to use this framework to choose systems for fusion. If we interpret our locations and customers as systems, rather than documents, we may use the evaluation measure of our choice for r, information difference or mutual information for w and s, respectively, and apply the algorithms above to merge the two. The question of how can we make use of this framework in practice, where we have no relevance information, is harder to answer. While we can compute the distance between systems by using their mutual information rather than their information difference, the difficulty lies in determining which systems are "best."

Determining which systems are best is equivalent to performing evaluation. Previous work [9, 75] has shown that in the absence of relevance judgments, it is possible to determine which systems are *worst* by sorting them by their consensus with the majority of other systems. However, we wish to determine which systems are best. For inspiration, we turn to very early research into the metasearch problem. For example, works such as Yuwono and Lee [92] and Lu *et al.* [49] incorporated models of system quality into their fusion algorithms. As a proxy for system quality, these algorithms explicitly computed similarity scores between the query and the top documents retrieved by the system. In essence, these early algorithms evaluated the systems being used in metasearch with regards to a privileged

system treated as a gold standard. We adopt this practice for our purposes, using either systems created specifically for this purpose, or by averaging multiple submitted systems, held out appropriately, chosen at random.

6.3.2 Experiments

In this section, we present our results employing the framework described in the previous section. We begin by analyzing our results when we have access to relevance judgments. In this hypothetical situation, we are analyzing the heuristic itself: is it best to choose the highest quality, most dissimilar systems? To answer this question, we choose systems to fuse using the Desirable Facility Placement (dfp) strategy (Algorithm 6.2), the Obnoxious Facility Dispersion (ofd) strategy (Algorithm 6.1), and a simple best-first (best) strategy. All metasystems are created using the CombMNZ metasearch algorithm. Figure 6.9 shows the results of this experiment as we increase the number of systems to fuse. λ, which varies from the equivalent of a clustering algorithm when $\lambda = 0$, to a best-first strategy when $\lambda = 1$, is fixed at 0.5. These results show that while there may be some minimal improvement using the MMR-like ofd strategy, the key to metasearch is simply in determining which systems are best. Figures 6.10, 6.11, and 6.12 show the results of tuning λ when choosing 3, 5, and 10 systems. The strategies' performance increases dramatically as λ approaches 1, when these strategies become equivalent to the best-first approach.

These results demonstrate that the best way to approach metasearch is to fuse the best systems. However, how does one determine which systems are best if one does not have access to relevance judgments? It is possible to generate some prior belief about system quality by treating a specific "control" system as some kind of gold standard by which to judge the others. This will certainly be better than random, but will not be accurate enough. Going back to the works by Soboroff *et al.* [75] and Aslam and Savell [9], we know that the better (though not necessarily the best) systems tend to be very similar. In these experiments we hypothesize that by combining these two imperfect sources of information using our FLA algorithms, we can improve upon the performance of simply fusing all systems, picking systems at random, or selecting the "best" systems according to our control-system prior.

To generate our prior belief in system quality, we must pick a system to act as a gold standard. For TRECs 8 and 9, we use the BM25 systems we generated in Section 6.2 with default parameters. For TRECs 2010 and 2011, we picked a system at random and evaluated our metasearch strategies over the remaining systems,

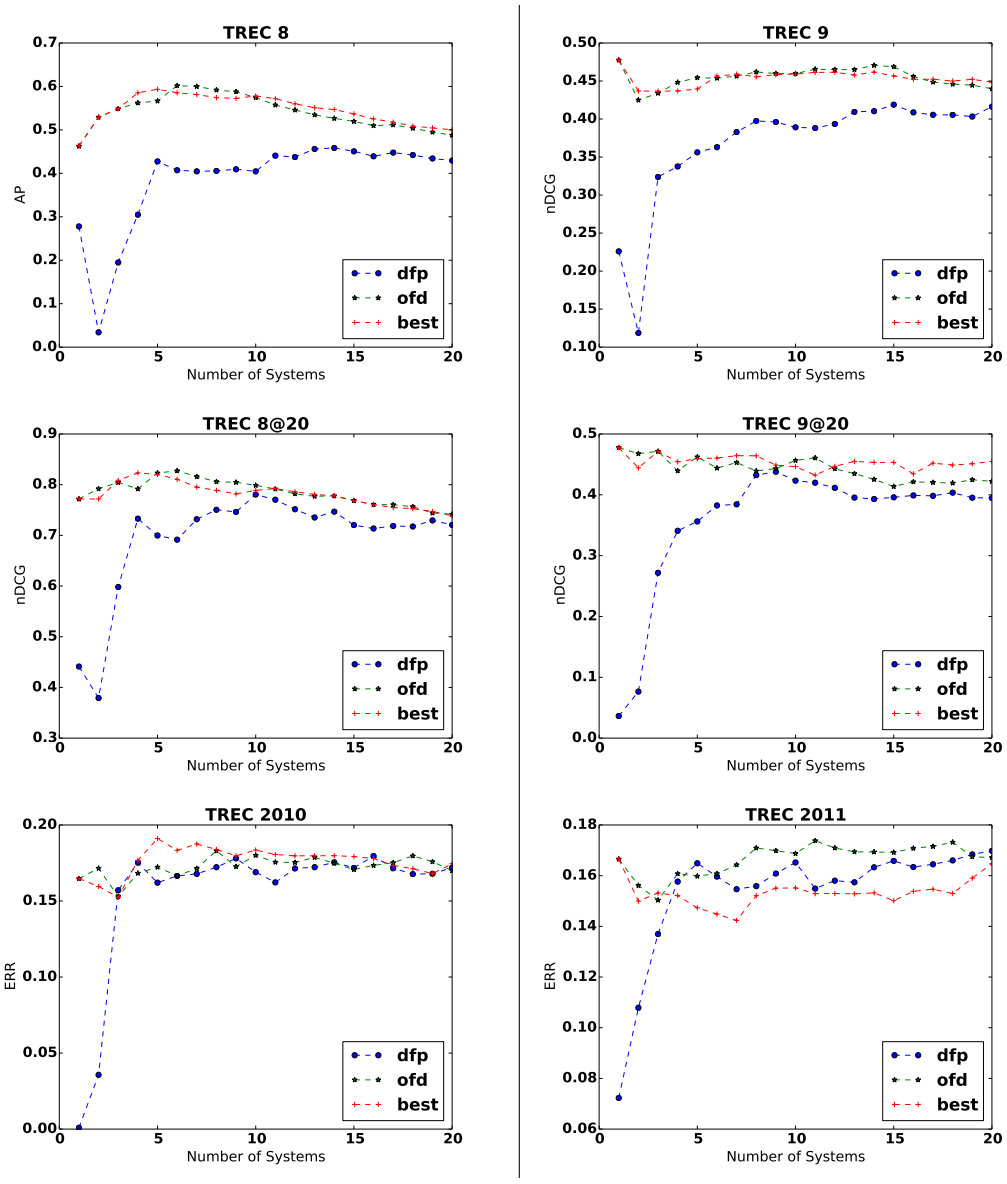


Figure 6.9: CombMNZ metasystems created using FLA algorithms, given relevance judgments, with $\lambda = 0.5$. There is little to no improvement over simply using the best systems.

with the "control" system removed from the collection. This process was repeated 100 times, and 95% confidence intervals are reported. In both cases, systems were "evaluated" by summing document scores from the control system over the set of documents retrieved by both systems. The scores were then normalized to be between 0 and 1 before being used in the computation of our heuristic functions. The similarity scores in our heuristic functions are computed using mutual information as described in Section 6.1. We also report the performance of fusing all

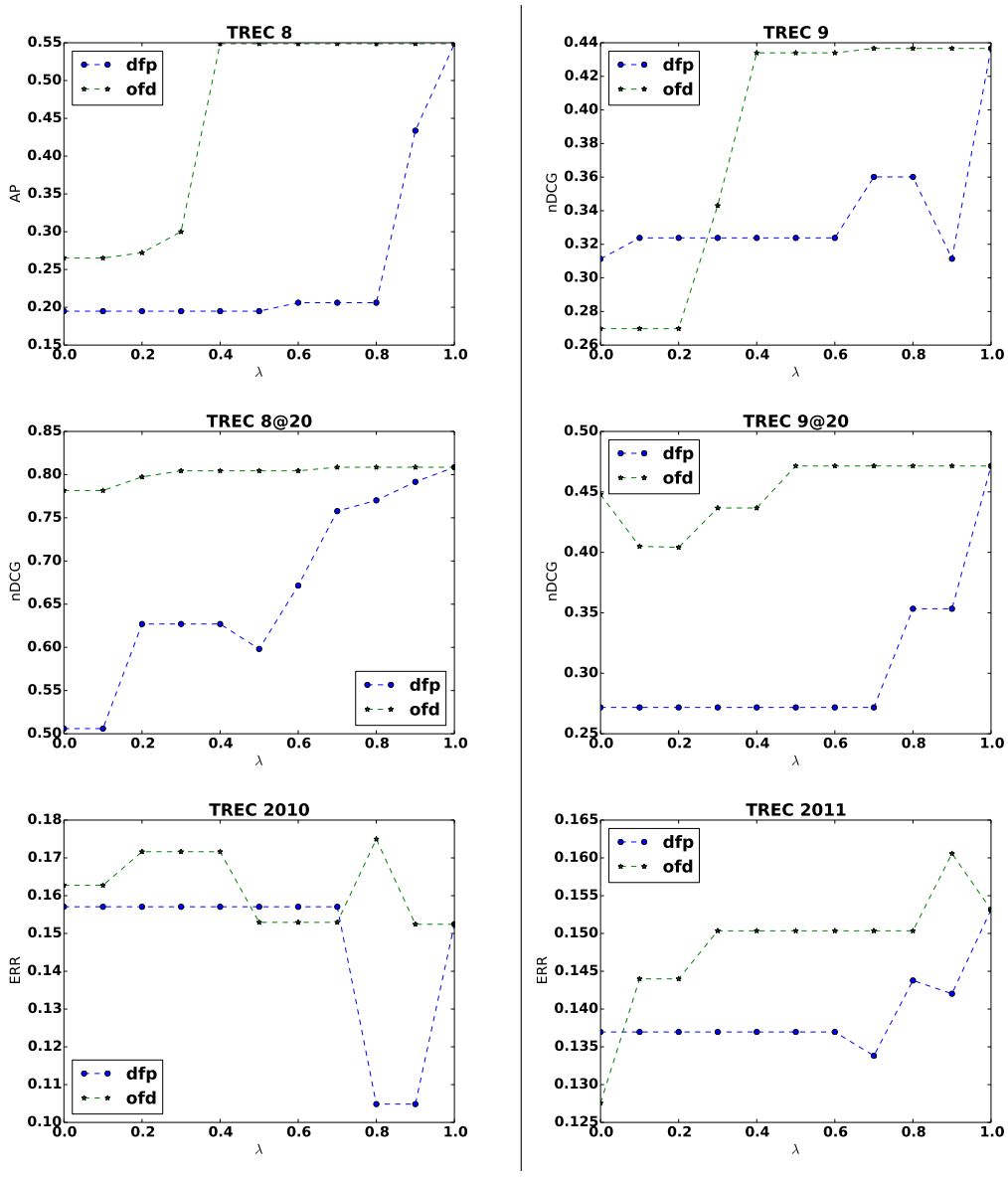


Figure 6.10: CombMNZ metasystems created from 3 input systems using FLA algorithms, given relevance judgments, as λ is varied. There is little to no improvement over simply using the best systems.

systems, and confidence intervals of the performance over sets of systems chosen at random, sampled 100 times. Figure 6.13 shows the results of this experiment as we increase the number of systems to fuse when fixing λ at 0.5. Figures 6.14, 6.15, and 6.16 show the results of tuning λ when choosing 3, 5, and 10 systems. In all cases, the dfs strategy shows some amount of improvement over simply fusing all systems, and significant improvement over all other strategies and simple random

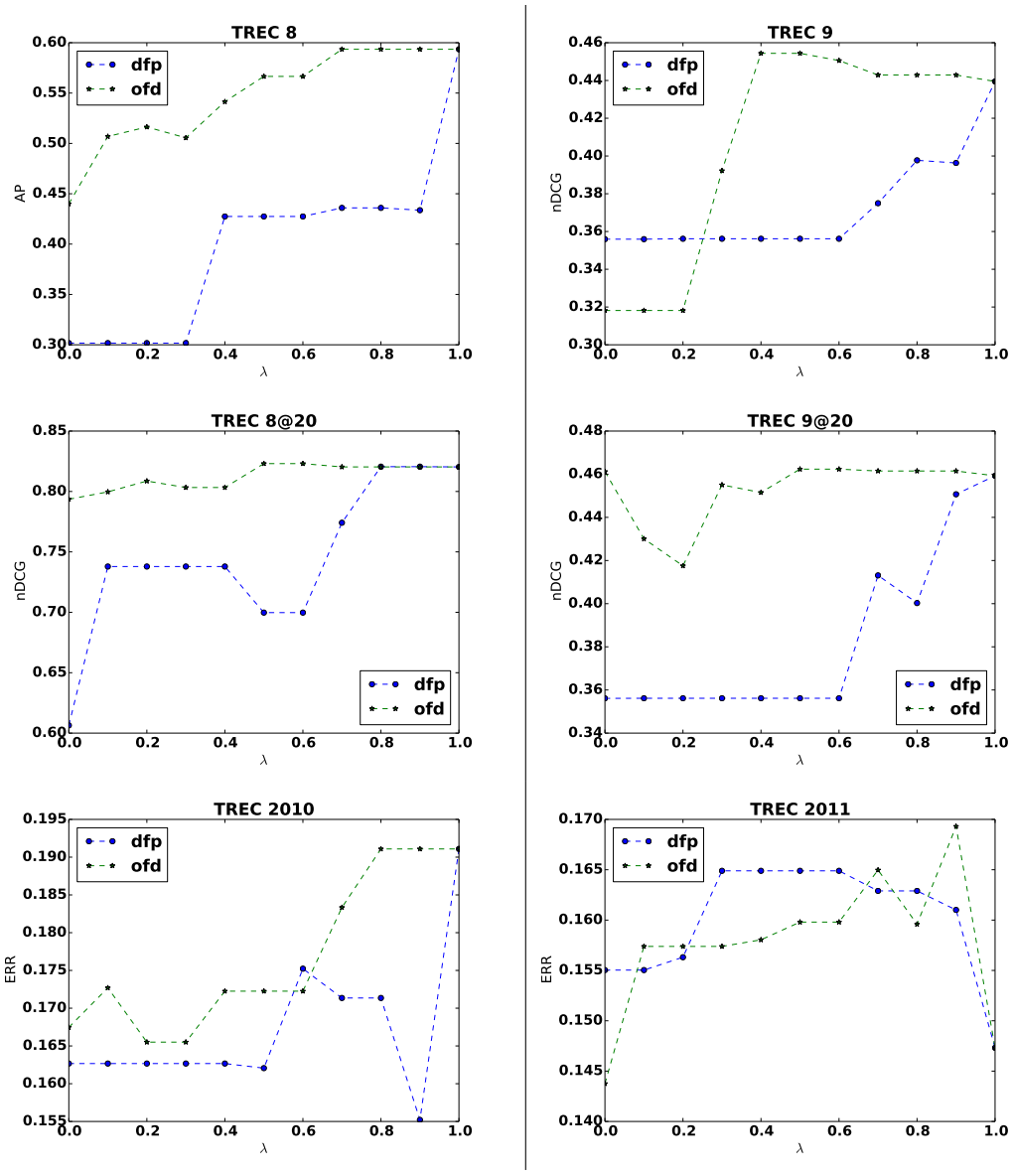


Figure 6.11: CombMNZ metasystems created from 5 input systems using FLA algorithms, given relevance judgments, as λ is varied. There is little to no improvement over simply using the best systems.

chance.

From this we can conclude that when performing metasearch, contrary to common wisdom, the key is to fuse the best systems; similarity does not play a significant role. However, this is not feasible in practice. Using our framework to compute similarity within the context of facilities location analysis, one can isolate the most *representative* systems. While further study is necessary, these preliminary

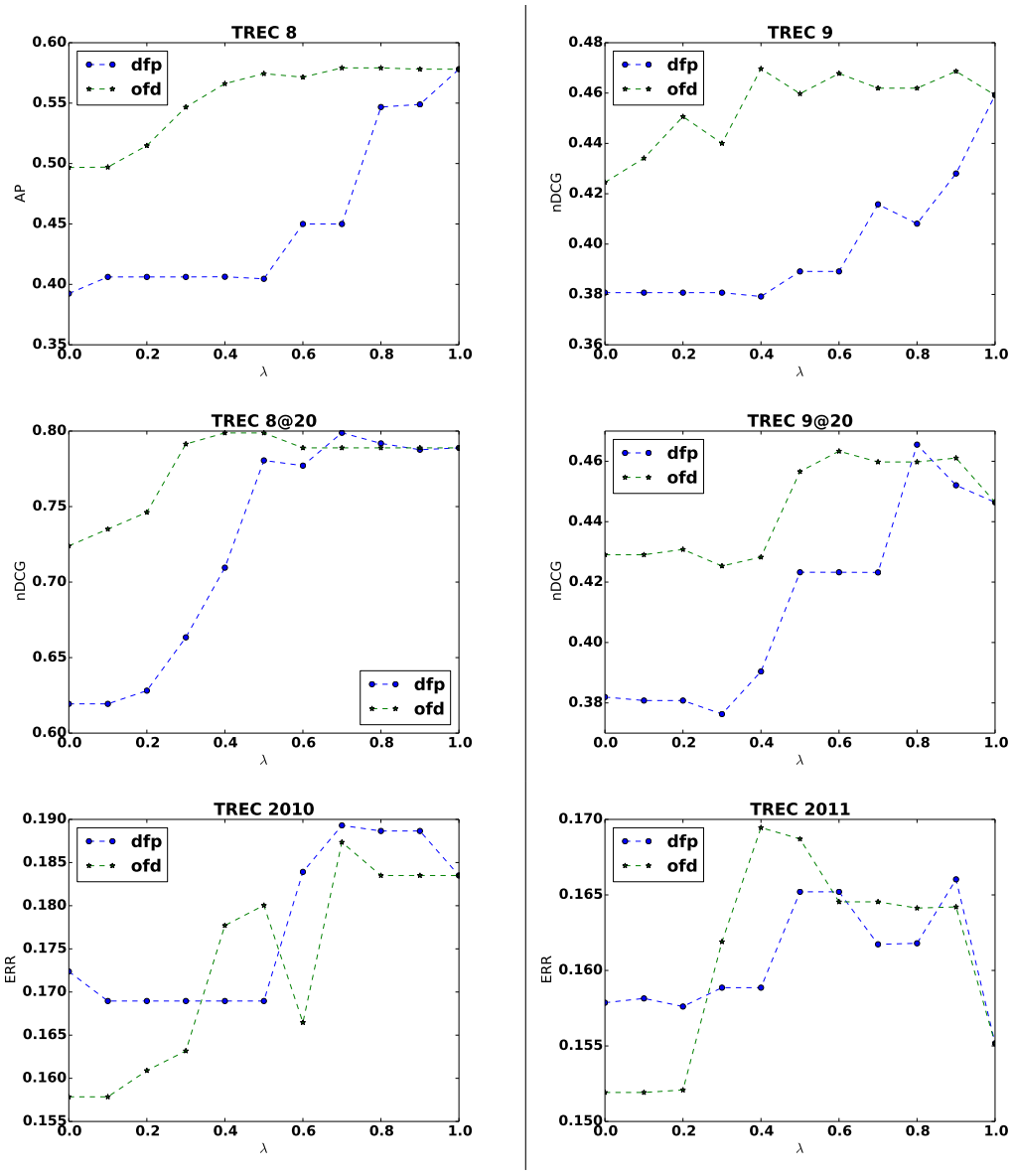


Figure 6.12: CombMNZ metasystems created from 10 input systems using FLA algorithms, given relevance judgments, as λ is varied. There is little to no improvement over simply using the best systems.

results demonstrate that this may improve upon the typical strategy of simply fusing as many systems as possible.

6.4 Summary

Search engines are usually compared in terms of their *performance*. However, it is possible for search engines to have quite different *behavior* in terms of which

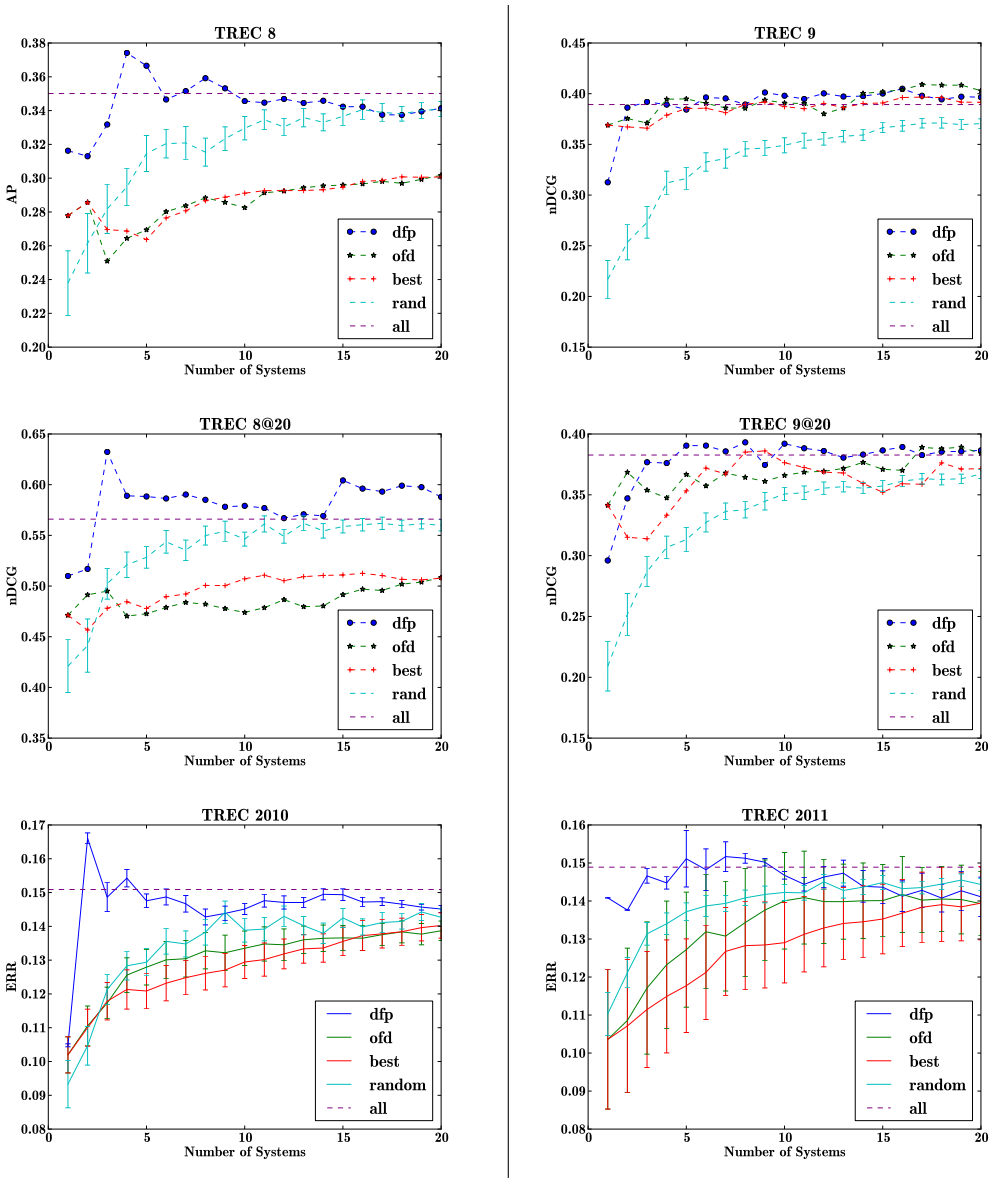


Figure 6.13: CombMNZ metasystems created using FLA algorithms, without relevance judgments, with $\lambda = 0.5$. There is some improvement over simply fusing all systems.

documents they rank, and in which order, and still have similar performance. In this chapter we developed *information difference*, a new tool for comparing search engines in terms of their behavior by comparing the order in which the systems rank judged documents. We showed that while it is trivial to determine whether systems are similar when most engines retrieve the same set of documents, it is necessary to utilize both document order and relevance judgments when there are many uniquely retrieved documents.

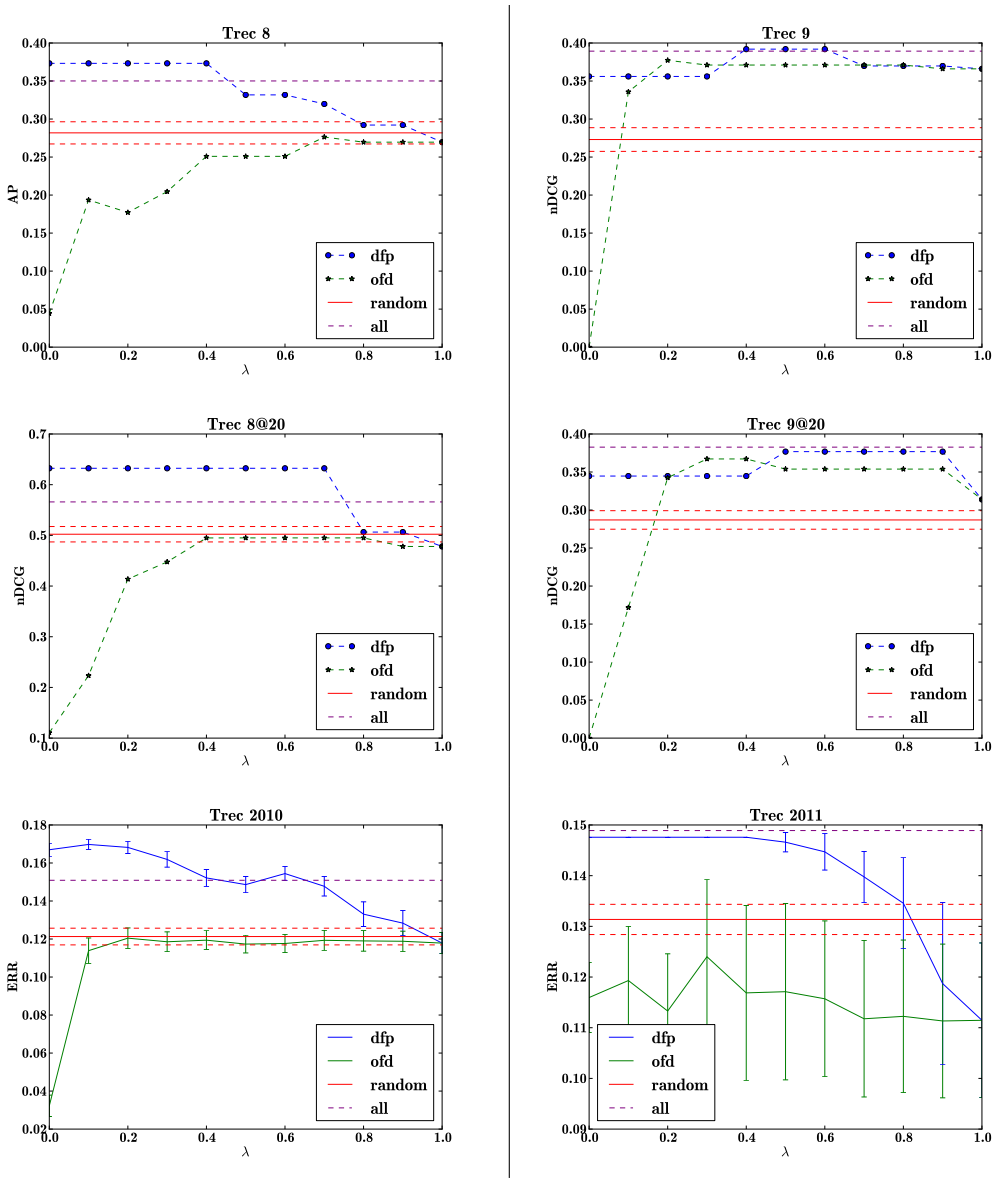


Figure 6.14: CombMNZ metasystems created from 3 input systems using FLA algorithms, without relevance judgments, as λ is varied. There is some improvement over simply fusing all systems.

Also in this chapter, we developed two novel applications of this technology. We used information difference to compare retrieval models that appear quite different, yet have deep theoretical connections and often have similar performance in practice. We found that these models could actually be more similar to one another than different instantiations of the same model. We also showed that information difference could be used to find subsets of systems that are representative of the

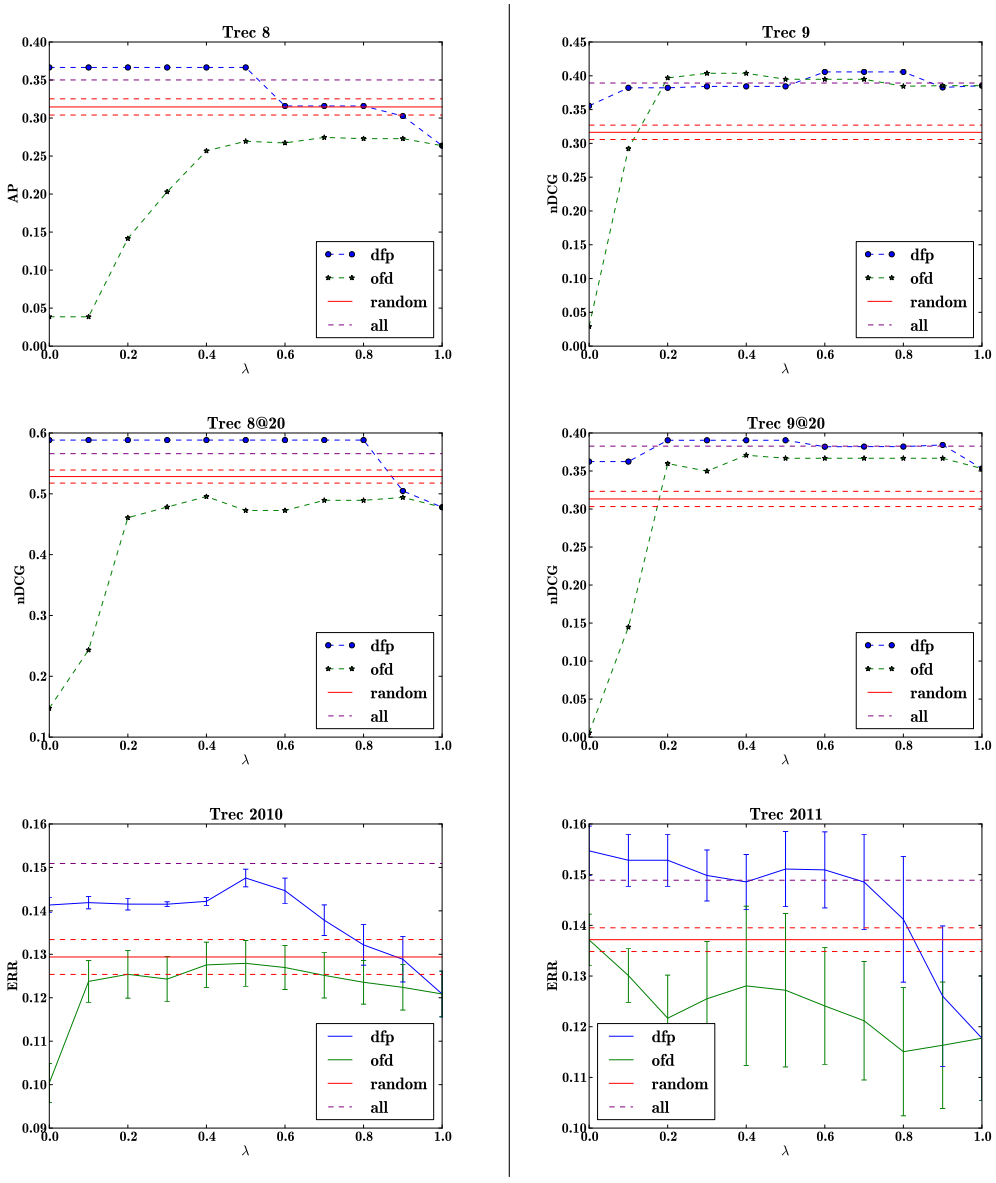


Figure 6.15: CombMNZ metasystems created from 5 input systems using FLA algorithms, without relevance judgments, as λ is varied. There is some improvement over simply fusing all systems.

whole collection, and that this can be leveraged for metasearch to significantly improve over selecting systems to fuse at random.

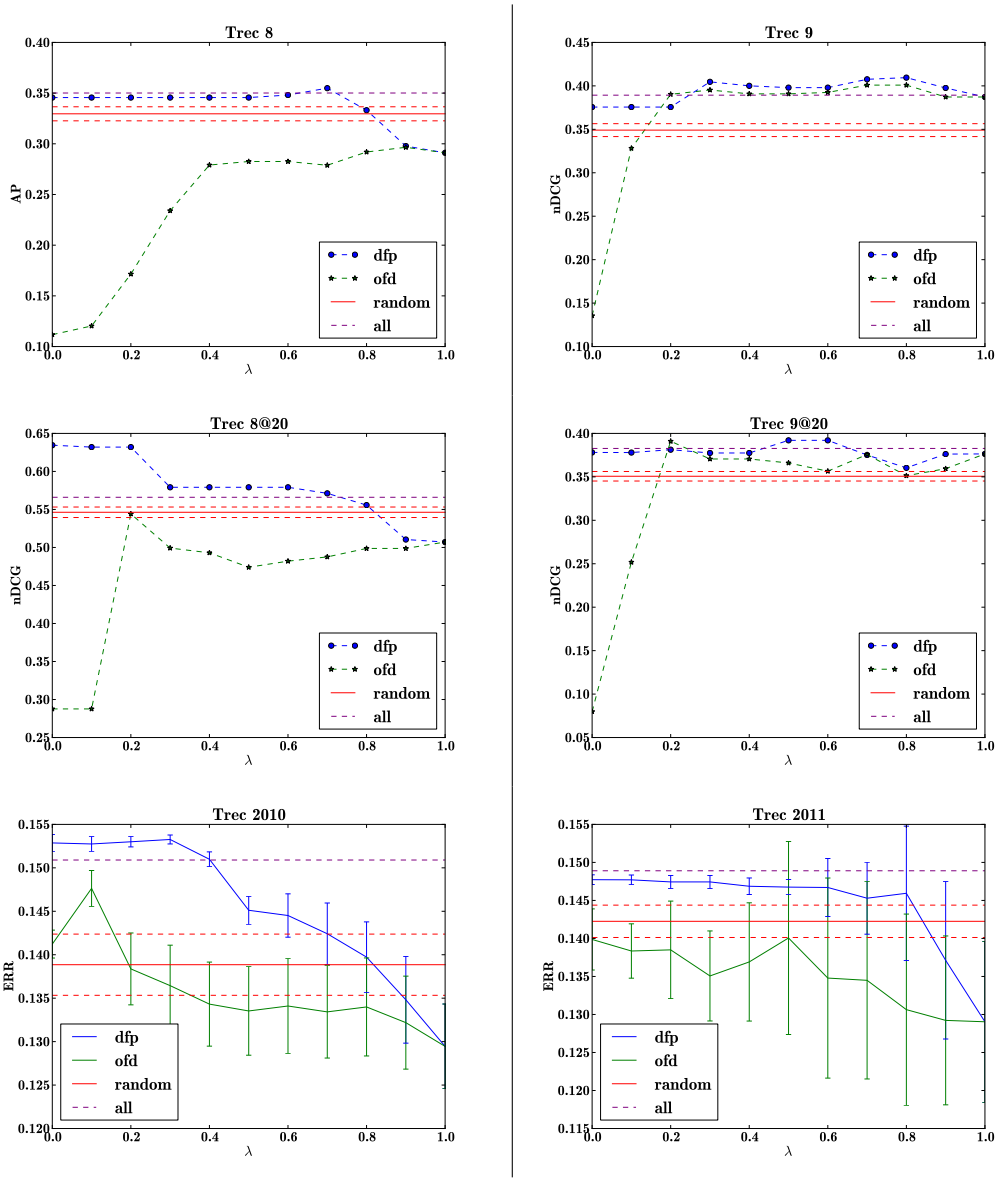


Figure 6.16: CombMNZ metasystems created from 10 input systems using FLA algorithms, without relevance judgments, as λ is varied. There is some improvement over simply fusing all systems.

Chapter 7

Conclusion

The goal of this work is two-fold: to 1) emphasize the need for increased meta-evaluation in information retrieval research, and to 2) provide a framework with which this can hopefully be achieved. We describe the utility of meta-evaluation research in terms of diversity. A search engine's diversity is necessarily conflated with its ability to perform ad hoc retrieval and the diversity of the collection. In this work, we attempted to isolate diversity from those other factors so that we can begin to understand it. We 1) introduced a meta-evaluation measure of sensitivity that controls for ad hoc performance, and 2) introduced a new family of measures that explicitly account for the collection diversity and. Our hypothesis is that these collection-oriented features, while opaque to the user, are better able to differentiate between systems, thereby leading to a better overall user experience. To assess collection difficulty, we developed measures at the topic and subtopic level. At the topic level, diversity difficulty blends the maximum possible number of subtopics covered by any ranked list with the number of subtopics covered by the expected ranked list. At the subtopic level, subtopic miss rate measures the probability of selecting documents at random and failing to cover subtopics. We showed that α#-IA measures, which combine the best features of existing evaluation measures and emphasize difficult topics and subtopics, sometimes rank systems in quite different orders than existing measures, yet have slightly more discriminative power.

That our measures prefer different systems does not indicate that they prefer more diverse systems. To show that our new measures preferred more diverse systems than existing measures, we restricted our attention to artificial ranked lists with perfect combined precision to show that our measures were less influenced by ad-hoc performance than existing measures. According to discriminative power, no measure was able to distinguish between these lists. This led us to introduce

document selection sensitivity, the coefficient of variation of an evaluation measure over these artificial ranked lists. According to this measure, α#-IA measures that explicitly account for collection diversity were far more sensitive to differences in these lists than existing measures, suggesting that these measures may prefer more diverse systems. However, while averaging subtopics by their difficulty also led to higher document selection sensitivity, it was still less than geometric averaging. This is likely due to limitations of our implementation of difficulty at the subtopic level.

We believe that these results support our hypothesis that taking a collection-oriented view of evaluation leads to systems that are preferable to the user. We contrast this with the user-oriented view of Sakai's intuitiveness measure [62–64]. We look forward to comparing these two approaches, in terms of correlation with each other and with the preference of actual users.

We note that our framework accepts any definition of difficulty at the collection level. In future work, we will explore alternate definitions of, and uses for, diversity difficulty at the topic and subtopic levels. We also wish to explore the correlation with diversity difficulty and ad hoc query difficulty. Is one predictive of the other?

There is also the question of incorporating relevance grades and intent probabilities into document selection sensitivity. We briefly suggested one way this can be done, but surely there are other ways. Would incorporating this information produce a more useful meta-evaluation?

Finally, recent work has shown that subtopic taxonomy [12], e.g. whether the subtopic is *navigational* or *informational*, has been shown to lead to better performance of both diversification algorithms [67] and diversity evaluation measures [62], since a user is far less tolerant of redundancy for a navigational query than an informational one. In future work, we intend to show the effect of incorporating subtopic taxonomy into document selection sensitivity and α#-IA measures.

To make it easier for future researchers to combine evaluation and meta-evaluation analyses, we developed a probabilistic framework for the analysis of information retrieval systems based on the correlation between a ranked list and the preferences induced by relevance judgments. Using this framework, we developed powerful information-theoretic tools for better understanding information retrieval systems. We introduced several preliminary uses of our framework: (1) a measure of conditional rank correlation, *information* τ, which is a powerful meta-evaluation tool whose use we demonstrated on understanding novelty and diversity evalution; (2) a new evaluation measure, *relevance information correlation*, which is correlated with traditional evaluation measures and can be used to (3) evaluate a collection

of systems simultaneously, which provides a natural upper bound on metasearch performance.

Additionally, we introduced a measure of the similarity between rankers on judged documents, *information difference*, which allows us to determine whether systems with similar performance are actually different. We used this measure to demonstrate that properly tuning a retrieval system is more important than selecting the right retrieval model. Further, we showed how by using information difference as a distance measure we were able to select the most representative systems for meta-search, significantly out-performing choosing systems at random.

We see great promise for this framework in the future. For example, in Section 2.3.2 we described the issue of rank correlation between rankings of subsets that only partially overlap, as well as the issue of aggregating many rankings of few objects into a single, maximally-consistent ranking. One aspect of comparing rankings that was not considered is that not all documents are equally important, *i.e.* swapping two non-relevant documents is unimportant, whereas swapping a highly relevant document with a slightly relevant document is. As we showed, using information difference, our framework can compare systems directly, conditioned on a QREL, and without the need to create a target list. In future work, we hope to extend this measure to the comparison of ranked lists of *systems* rather than ranked lists of *documents*. Given some notion of the *correct* ordering of systems, *i.e.* some analogue of a QREL, we could compare evaluation measures based on which systems they preferred, using a system similar to information difference. This is in contrast to current meta-evaluation measures such as discriminative power and document selection sensitivity, that compare measures based on how likely they are to have a preference. The difficulty lies in creating this QREL anologue. Further, given this information about the true preference between systems, why would it be necessary to do further evaluation? One potential application would be in the analysis of crowdsourced evaluation, in which we have a gold standard ranking induced by judgments created by trained assessors. Using something akin to information difference, we could compare alternative evaluation measures to one another in an IR-motivated way that is sufficiently top-heavy and that accounts for system quality appropriately.

Another potential application of our framework is to diversity evaluation. Consider the recent work by Chandar and Carterette [23, 24], in which the authors solicit diversity preferences in the form of *document triples*. To collect these preferences, users are presented with an initial relevant document, the *top* document, and then asked which of two additional relevant documents, the *left* and *right* docu-

ments, they would prefer to see next in a ranked list. These documents are denoted D_T, D_L, and D_R, respectively. Denote a triple as $\langle D_L, D_R \mid D_T \rangle$. For a given triple, Let $\succ$ denote a user's preference, e.g. $D_L \succ D_R$ means that the user preferred the left document to the right document. In our framework, we can frame these preference triples as random variables. Given a triple $\langle D_L, D_R \mid D_T \rangle$ where $D_L \succ D_R$, let r_i represent the rank of document d_i in the list S. Then the ranked list variable R_S can be defined along the following lines:

$$R_S(\langle D_L, D_R \mid D_T \rangle \text{ where } D_L \succ D_R) = \begin{cases} 1 & \text{if } r_T < r_L < r_R \\ 0 & \text{if } r_T \not< r_L \text{ or } r_T \not< r_R \\ -1 & \text{if } r_T < r_R \leq r_L \end{cases} \tag{7.1}$$

Using such a random variable, along with an appropriate distribution over triples, we can evaluate systems using these preferences. In those way, our ad hoc evaluation measure can also be used as a diversity measure simply by using a different set of relevance judgments. Further, comparing this user-driven model of diversity to that of existing measures can help determine whether the current diversity evaluation paradigm is indeed correlated with user preferences.

Finally, we believe our framework can be applied to the reusability issue inherent to web-scale evaluation. As we discussed in Section 2.3.1, one of the key challenges in information retrieval as it migrates to the web is the issue of scale. From 1992 to 1999 the various corpora used by TREC contained less than two million documents and were distributed together on 6 CD-ROMs. While it was not possible for a human being to judge each of these documents with regards to each topic in every collection, it was possible to judge a representative sample. The corpus used from 2009 to 2012, which is insignificant compared to the web itself, contains over one billion documents and was distributed on two separate three Terabyte hard drives. Only a relatively insignificant fraction of these documents have ever been read by humans. Our framework can leverage the relatively large numbers of noisy assessments created by crowdworkers, as well as the actual user preferences collected by commercial search engines. Also, search engines themselves, by their very nature, produce estimated relevance assessments; assuming we had a gold standard ranker in which we had a great deal of trust, we could estimate the relevance of any document with regards to any query. Since these different sets of assessments will only partially overlap, and are very likely to contradict one another, currently, in order to leverage all of these assessments to perform a single evaluation, it would be necessary to evaluate with regards to each in turn and av-

erage the results, or else combine them all into a single, unified set of evaluations. In our framework, just as we evaluated multiple systems simultaneously with respect to a single QREL to find upper bounds on metasearch, we can evaluate a single system with respect to multiple QRELs.

$$RIC(S) = I(S; Q_1, \ldots, Q_n) \tag{7.2}$$

The ability to leverage all possible sources of relevance assessments for evaluation will be of great use to IR researchers as well as commercial search engines.

Appendix A

Additional Figures

In this Appendix, we present additional figures that were not included in the main body of the text.

A.1 Discriminative Power of α#-IA Measures

In Section 4.1, we introduced the family of Intent-Aware cascade #-measures. α#-IA measures are defined as a linear combination of S-Recall and an intent aware cascade measure. For example,

$$\alpha\#\text{-nDCG-IA@}k = \lambda \times \text{S-Recall@}k + (1-\lambda)\sum_{i=1}^{M} w_i \times \alpha\text{-nDCG}_i\text{@}k. \tag{A.1}$$

In Section 4.2, we analyzed the sensitivity of α#-IA measures in terms of discriminative power [60] (Section 2.3.3), finding that our measures have slightly higher discriminative power than existing measures.

With respect to discriminative power, there are four aspects of α#-IA measures that can be varied: the choice of discount function (Table 2.1), the α and λ parameters used to model a user's tolerance for redundancy and the weight given to S-Recall, respectively, the choice of subtopic normalization (see Table 4.2), and the rank at which evaluation is performed. In the main body of the text, we focused on DCG discounting at rank 20. In this appendix we present results for other choices of discount function and rank. As before, in all experiments, α and λ vary over the set $\{0, 0.1, 0.2, \ldots, 1\}$. When using RBP, β is set to 0.8.

Figures A.1 through A.8 show the discriminative power of each evaluation measure for all values of α and λ, sorted by discount function and then by rank. We can compare the α#-IA measures to existing measures (with the exception of D# measures) by carefully considering these plots. For any subtopic average, setting

$\lambda = 1$ (the far-right side in 3D plots) shows S-Recall. Using the cascade average and setting $\lambda = 0$ (the near-left side in 3D plots) shows α-nDCG. Using the micro average and setting $\lambda = \alpha = 0$ (the leftmost corner) shows nDCG-IA. As before, since the maximum for each year is achieved by cascade averaging, and not on the near-left or far-right side (i.e. it is achieved with $0 < \lambda < 1$), we can conclude that the α#-IA measures do have somewhat higher discriminatory power than existing measures.

Figures A.9 and A.10 show the impact of λ as α is fixed. Figures A.11 and A.12 show the impact of α as λ is fixed. From these it is clear that while the subtopic averages that emphasize the difficult subtopics—the geometric average (geom) and the subtopic miss rate-weighted average (smr)—have lower discriminative power overall, they are comparable when α and λ are appropriately tuned.

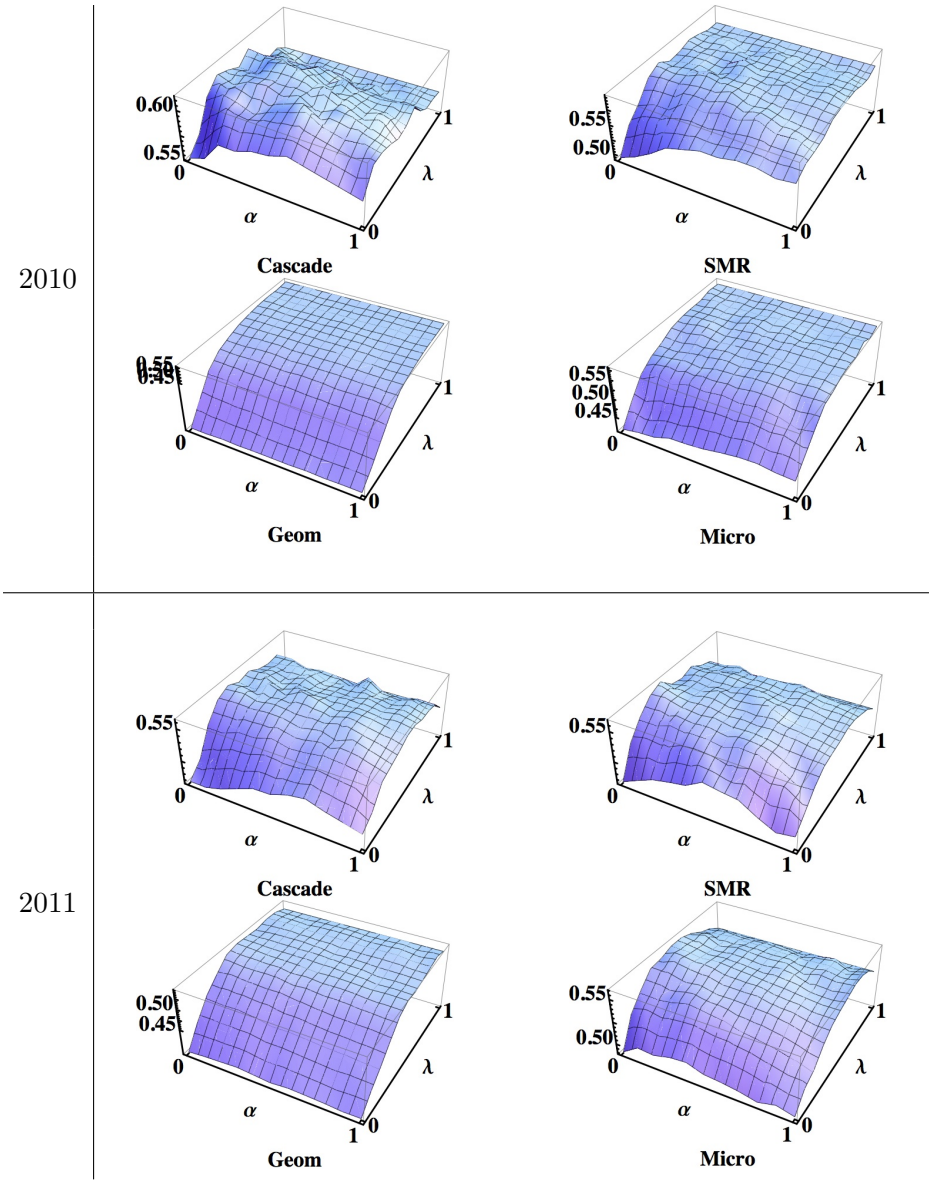


Figure A.1: Discriminative power at rank 5 using ERR discounting as a function of α and λ.

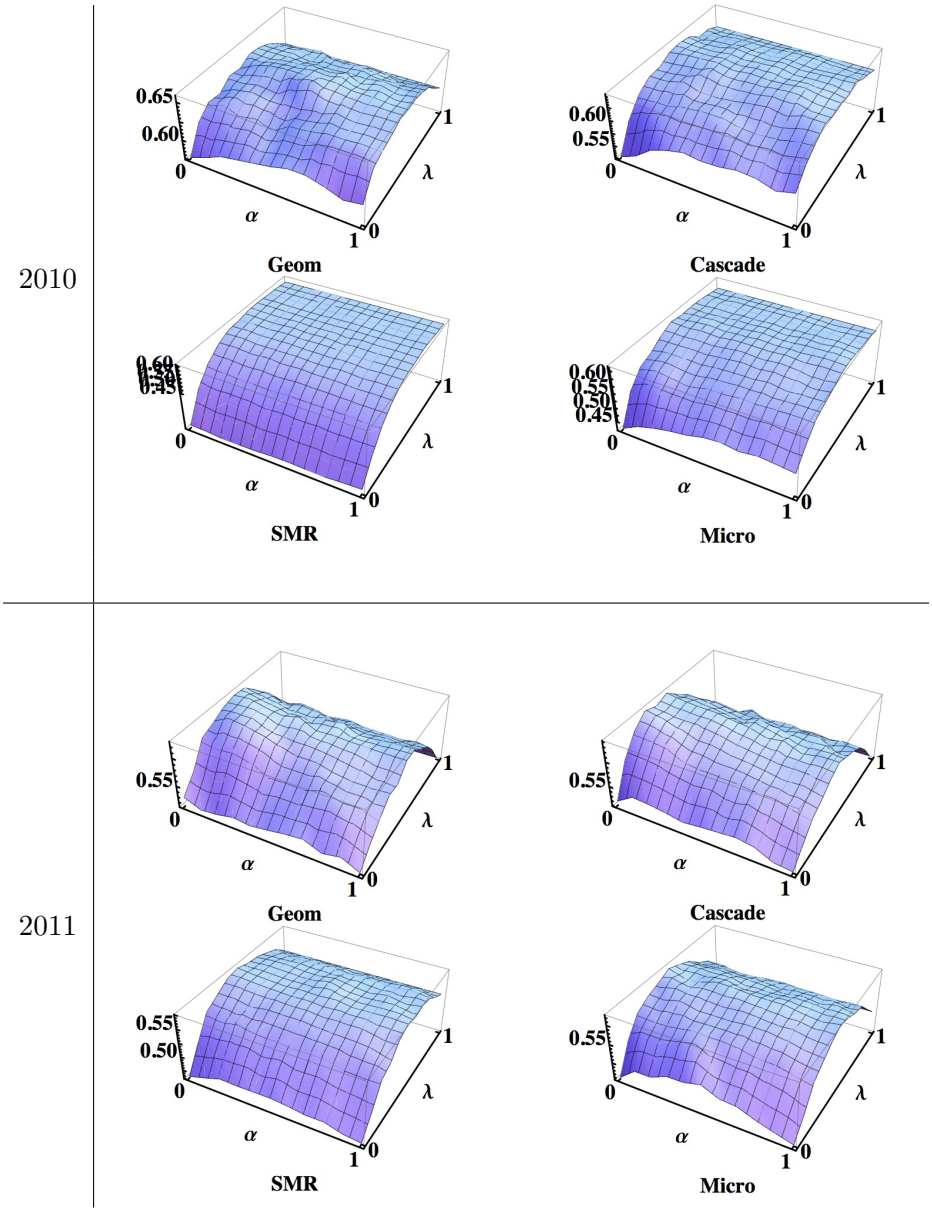


Figure A.2: Discriminative power at rank 10 using ERR discounting as a function of α and λ.

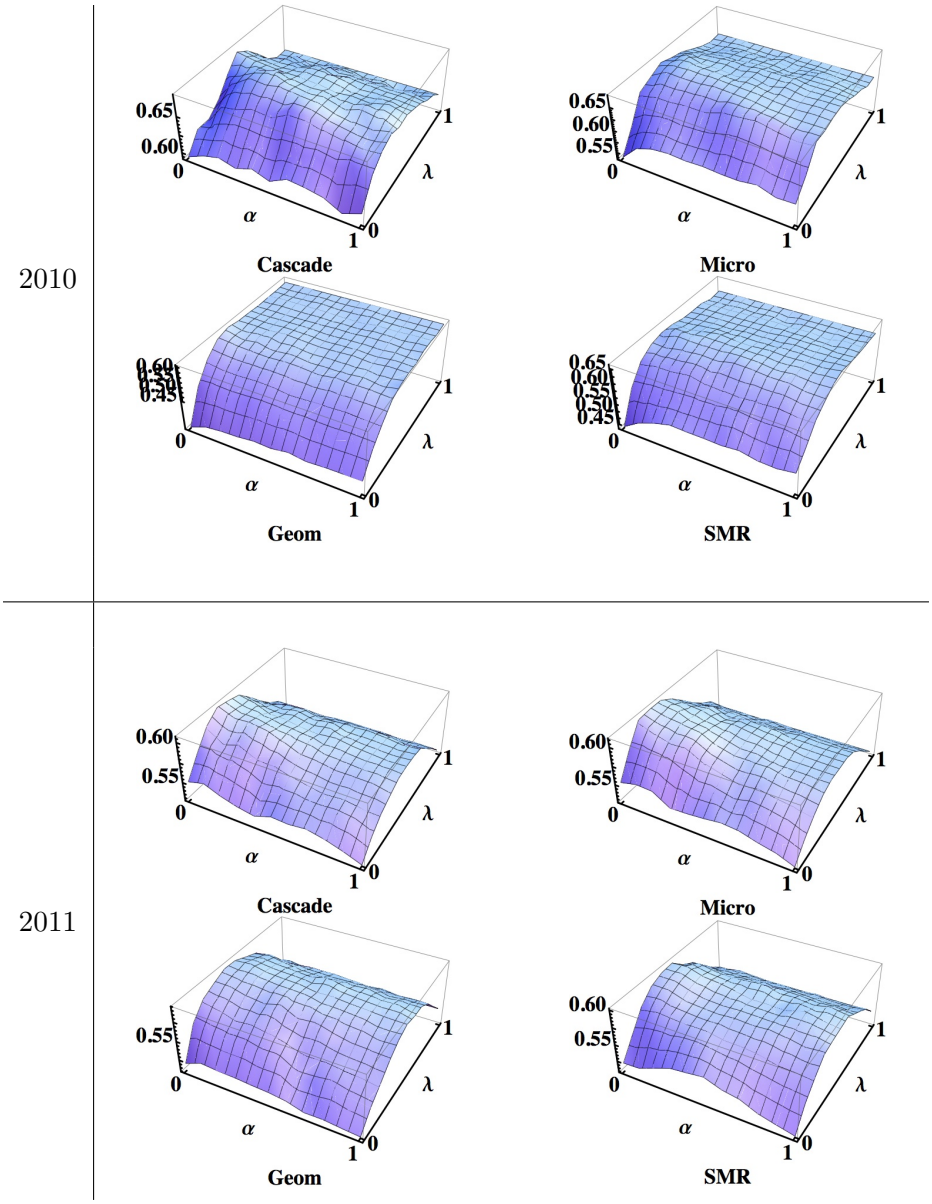


Figure A.3: Discriminative power at rank 20 using ERR discounting as a function of α and λ.

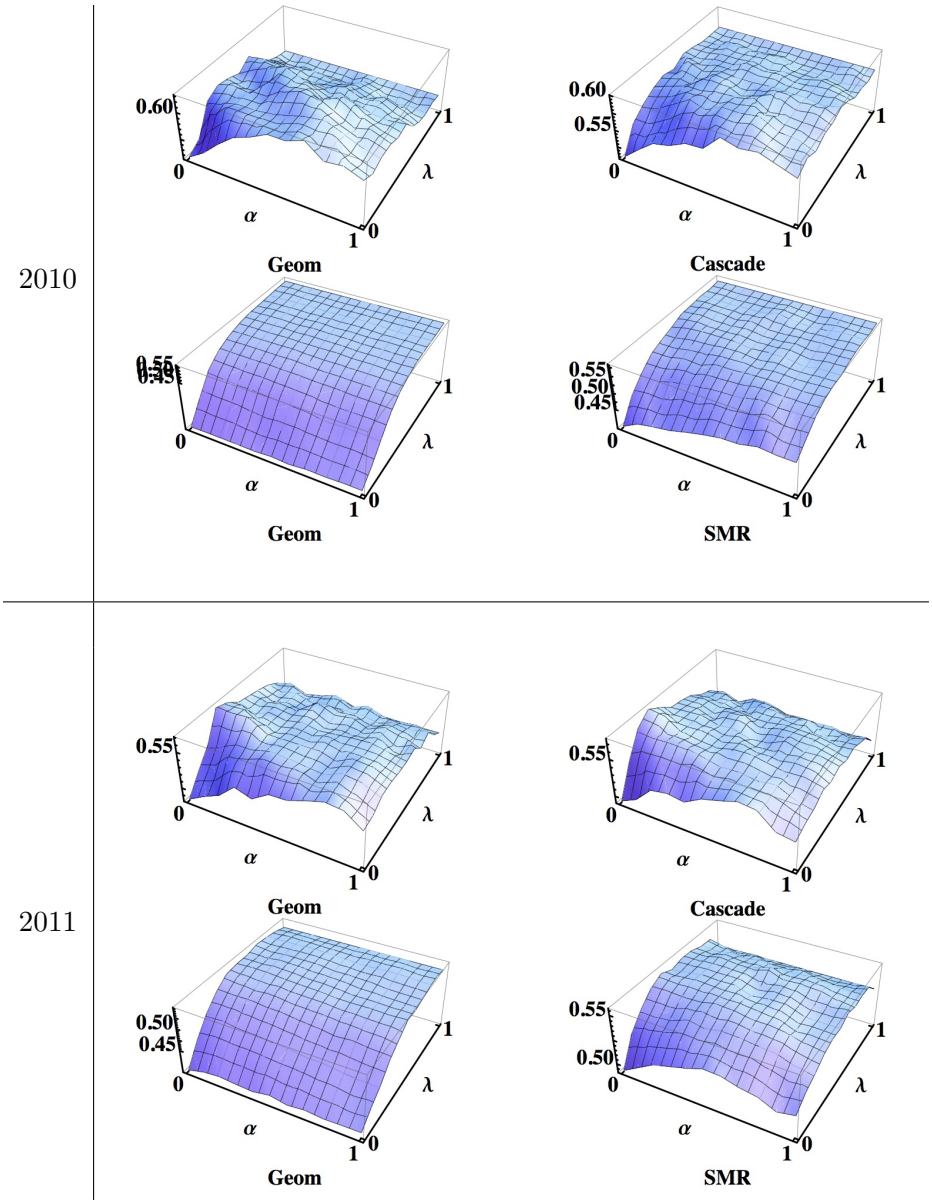


Figure A.4: Discriminative power at rank 5 using DCG discounting as a function of α and λ.

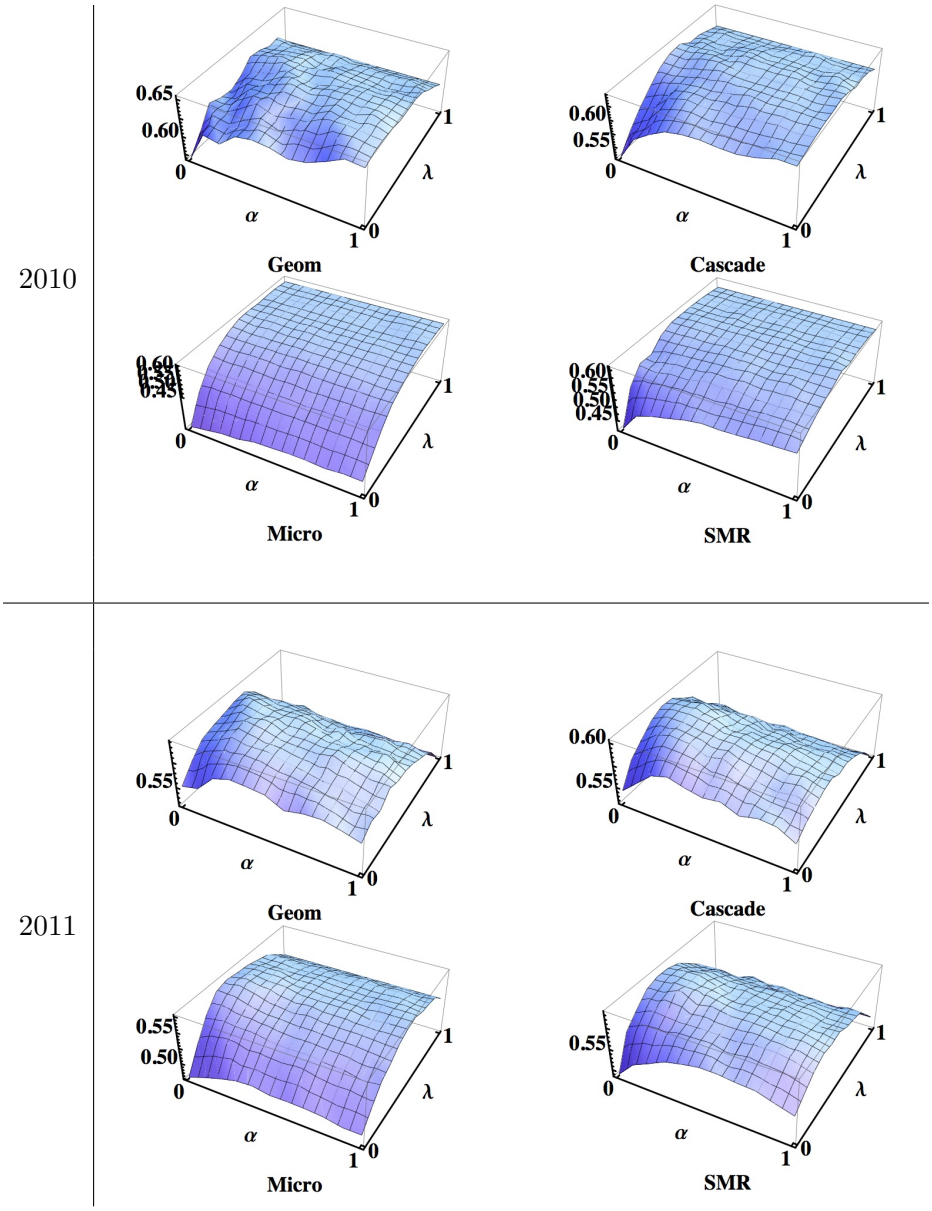


Figure A.5: Discriminative power at rank 10 using DCG discounting as a function of α and λ.

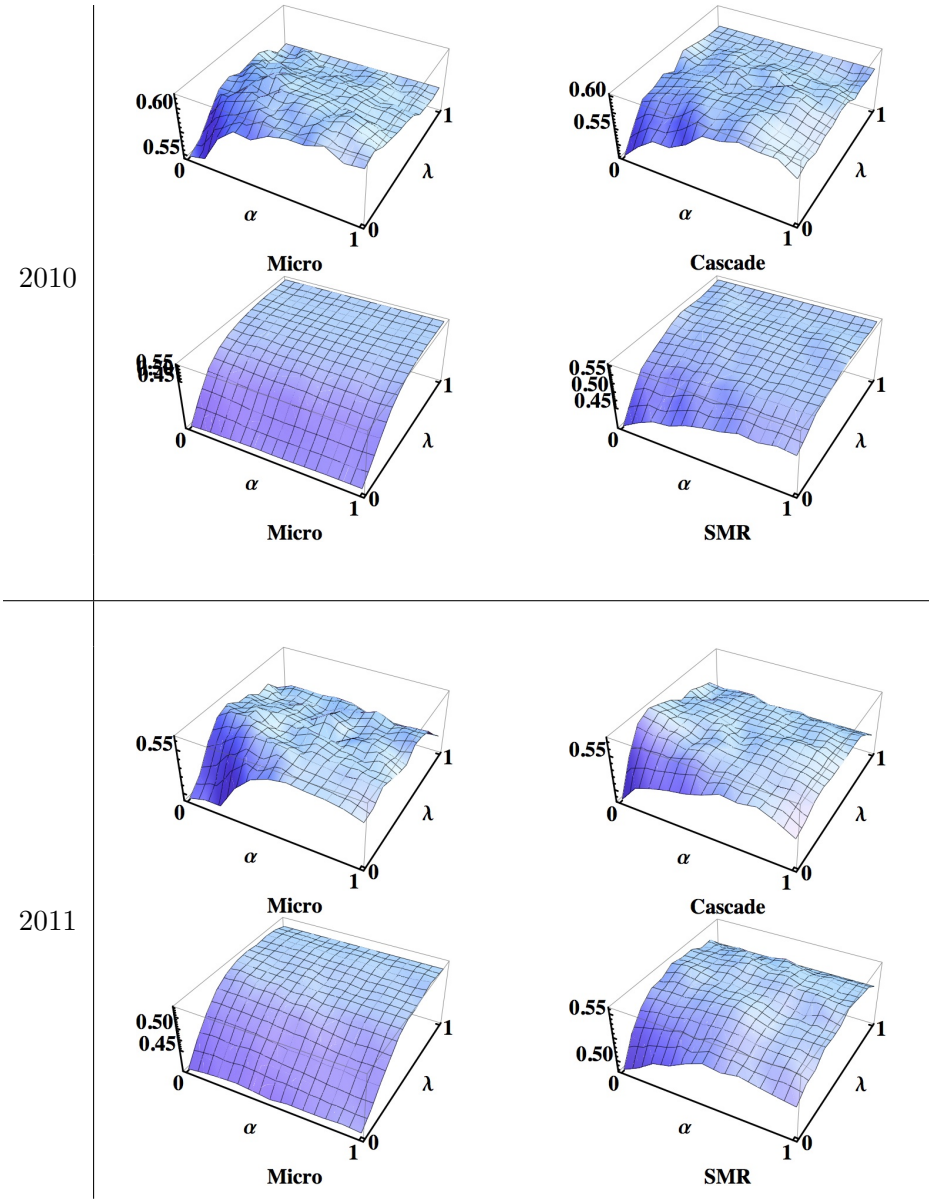


Figure A.6: Discriminative power at rank 5 using RBP discounting as a function of α and λ.

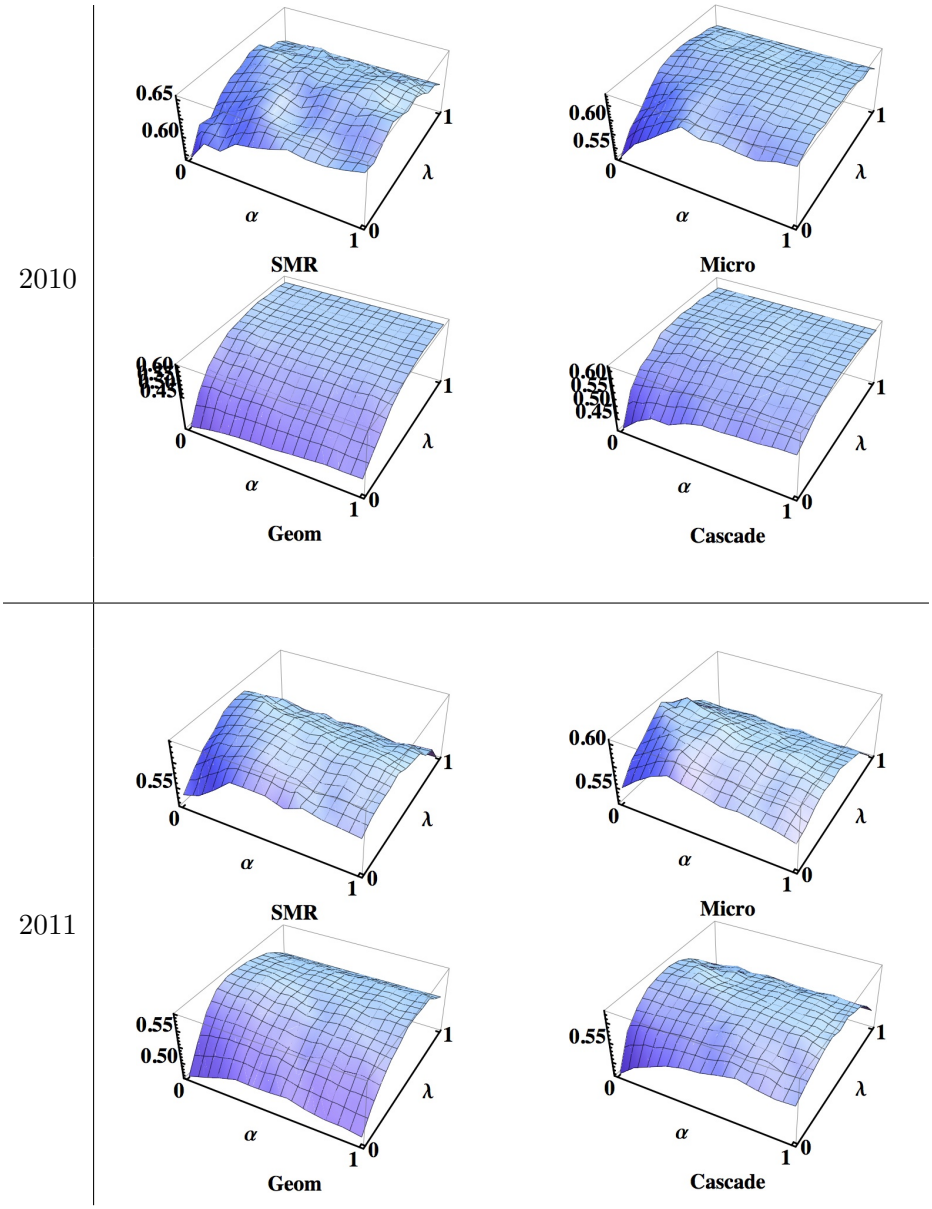


Figure A.7: Discriminative power at rank 10 using RBP discounting as a function of α and λ.

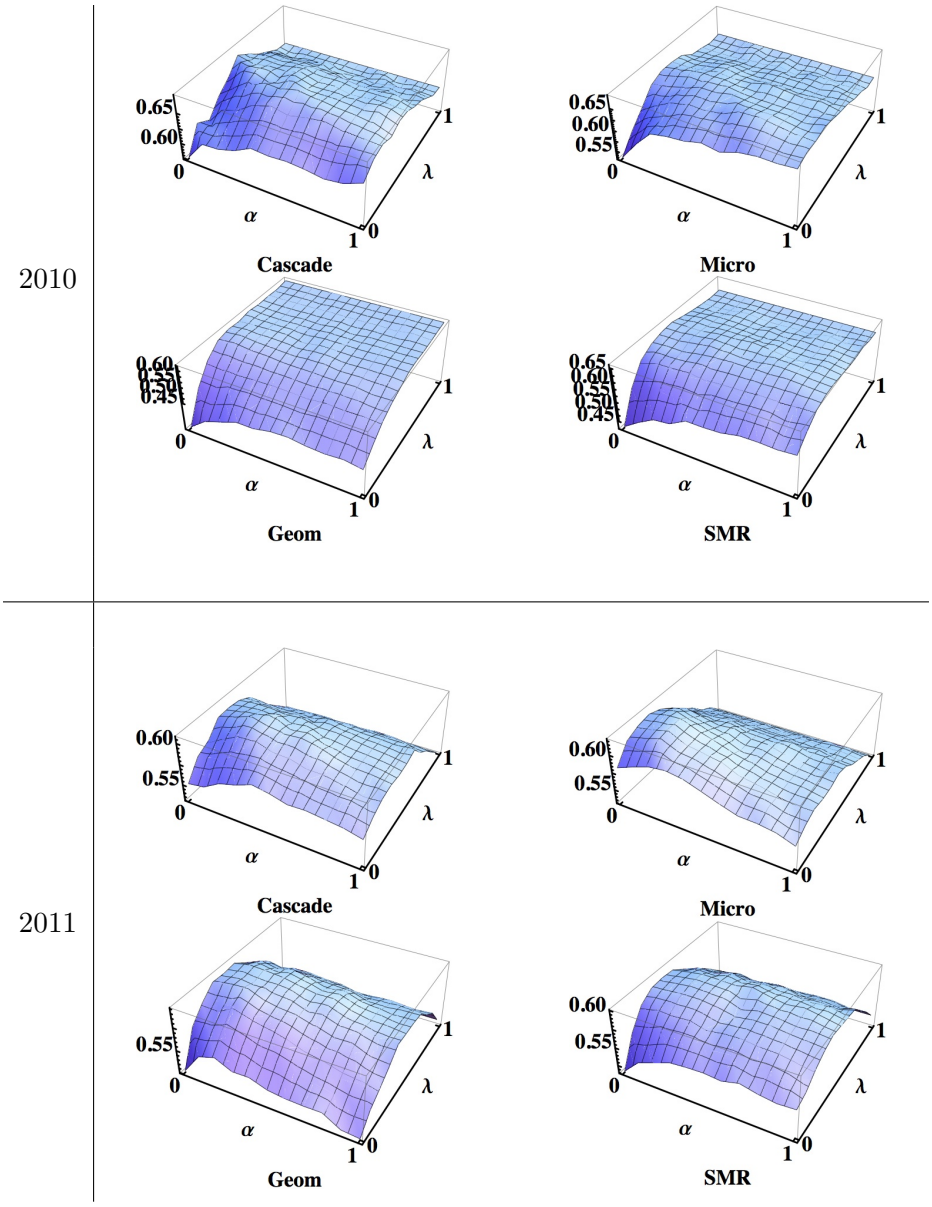


Figure A.8: Discriminative power at rank 20 using RBP discounting as a function of α and λ.

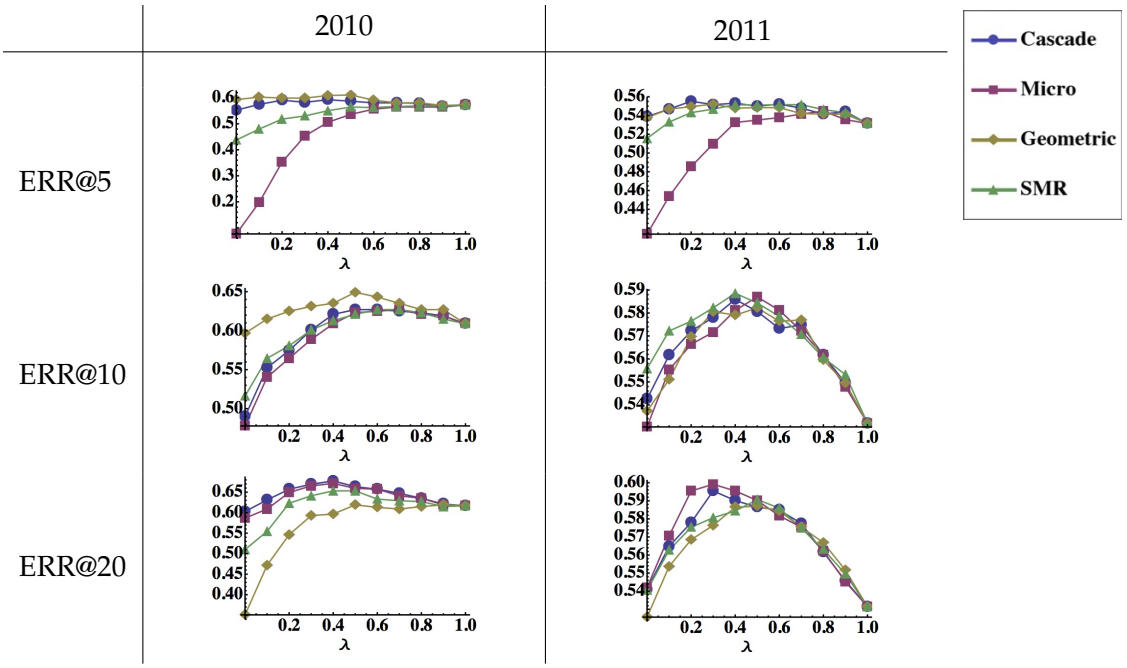


Figure A.9: Discriminative power of as a function of λ with ERR discounting. α is fixed at 0.3.

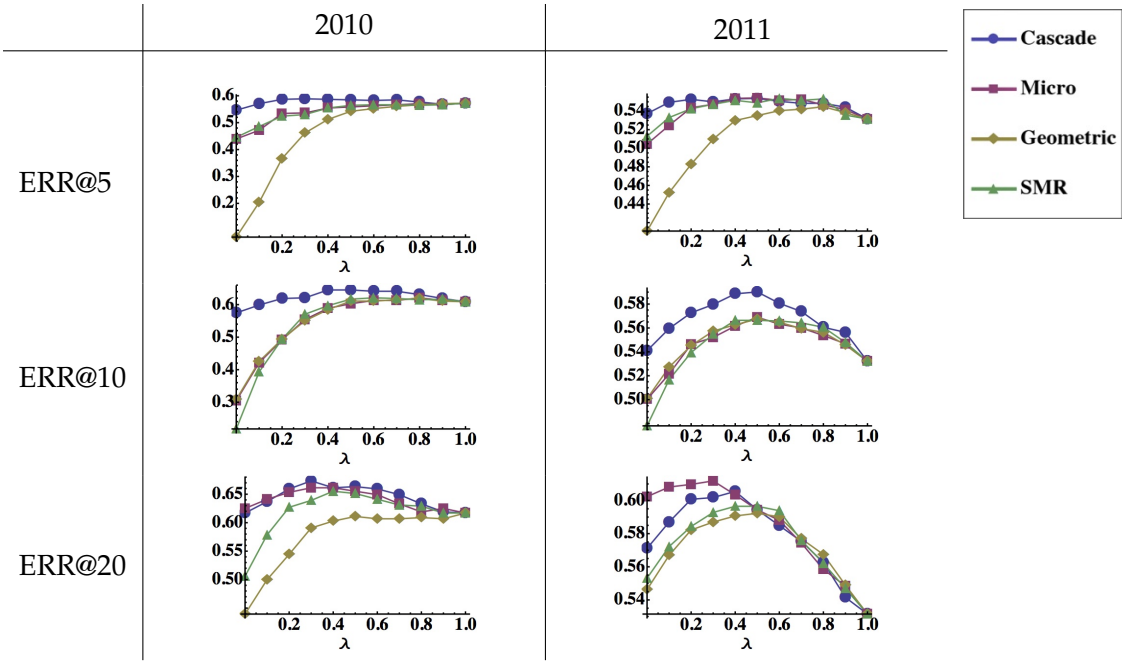


Figure A.10: Discriminative power of as a function of λ with RBP discounting. α is fixed at 0.3.

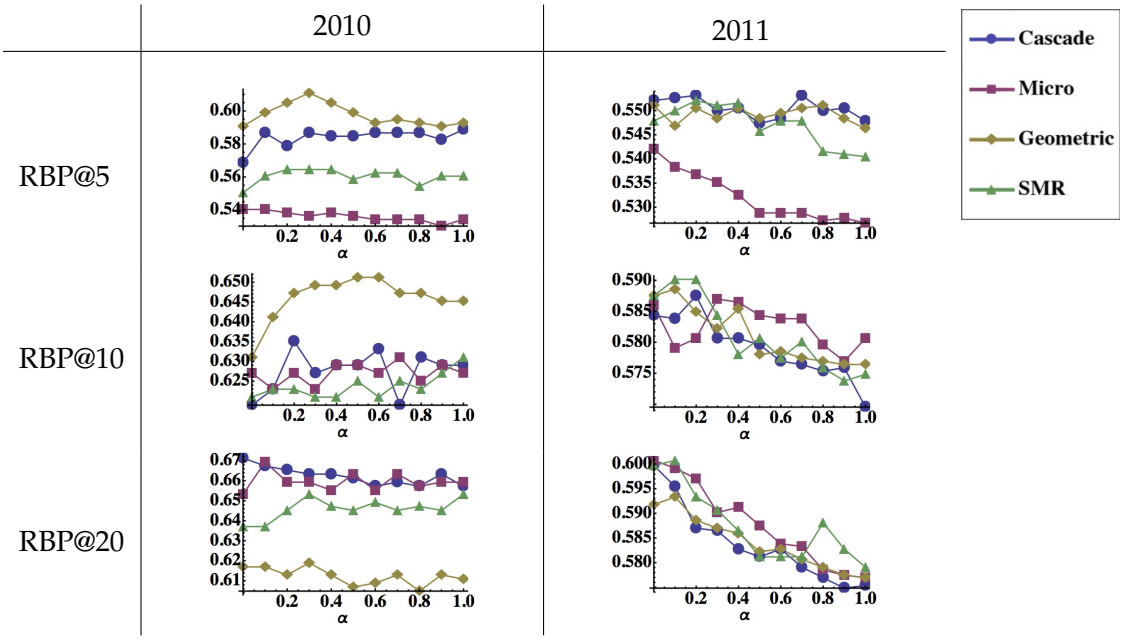


Figure A.11: Discriminative power as a function of α with ERR discounting. λ is fixed at 0.5.

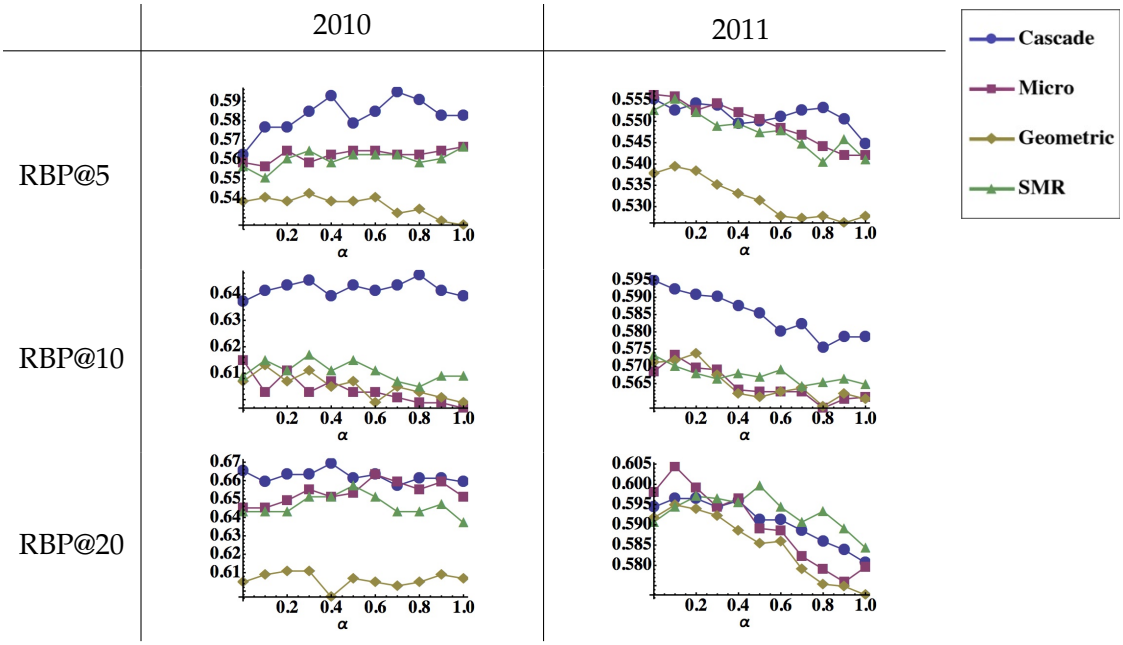


Figure A.12: Discriminative power as a function of α with RBP discounting. λ is fixed at 0.5.

A.2 Document Selection Sensitivity

In Section 4.1, we introduced the family of Intent-Aware cascade #-measures. α#-IA measures are defined as a linear combination of S-Recall and an intent aware cascade measure. For example,

$$\alpha\#\text{-nDCG-IA@}k = \lambda \times \text{S-Recall@}k + (1-\lambda)\sum_{i=1}^{M} w_i \times \alpha\text{-nDCG}_i\text{@}k. \qquad \text{(A.2)}$$

In Section 4.4, we analyzed the sensitivity of α#-IA measures in terms of document selection sensitivity (dss; Section 3.1), finding that choosing averaging methodologies that emphasize difficult topics and subtopics can greatly increase dss.

With respect to dss, there are five aspects of α#-IA measures that can be varied: the choice of discount function (Table 2.1), the α and λ parameters used to model a user's tolerance for redundancy and the weight given to S-Recall, respectively, the choice of topic average (Table 4.1), the choice of subtopic normalization (Table 4.2), and the rank at which evaluation is performed. In the main body of the text, we focused on DCG discounting at rank 20. In this appendix we present results for other choices of discount function and rank. As before, in all experiments, α and λ vary over the set $\{0, 0.1, 0.2, \ldots, 1\}$. When using RBP, β is set to 0.8.

Figures A.13 through A.20 show the dss of each evaluation measure for all values of α and λ, sorted by discount function and then by rank. We can compare the α#-IA measures to existing measures (with the exception of D# measures) by carefully considering these plots. For any subtopic average, setting $\lambda = 1$ (the far-right side in 3D plots) shows S-Recall. Using the cascade average and setting $\lambda = 0$ (the near-left side in 3D plots) shows α-nDCG. Using the micro average and setting $\lambda = \alpha = 0$ (the leftmost corner) shows nDCG-IA. Since the maximum is achieved by geometric subtopic averaging (geom), and not on the far-right side where $\lambda = 1$, we can conclude that the α#-IA measures can have significantly higher document selection sensitivity than existing measures. As before, we see that dss decreases with rank.

Figures A.21 and A.22 show the impact of λ as α is fixed. Figures A.23 and A.24 show the impact of α as λ is fixed. These figures show that:

1. geometric (geom) and arithmetic topic averaging are quite similar,
2. diversity difficulty topic weighting (DD) shows marked increases in selection sensitivity, as does geometric subtopic weighting (geom), and
3. subtopic miss rate weighting (smr) has higher selection sensitivity than subtopic

intent-weighted (micro) and cascade normalization.

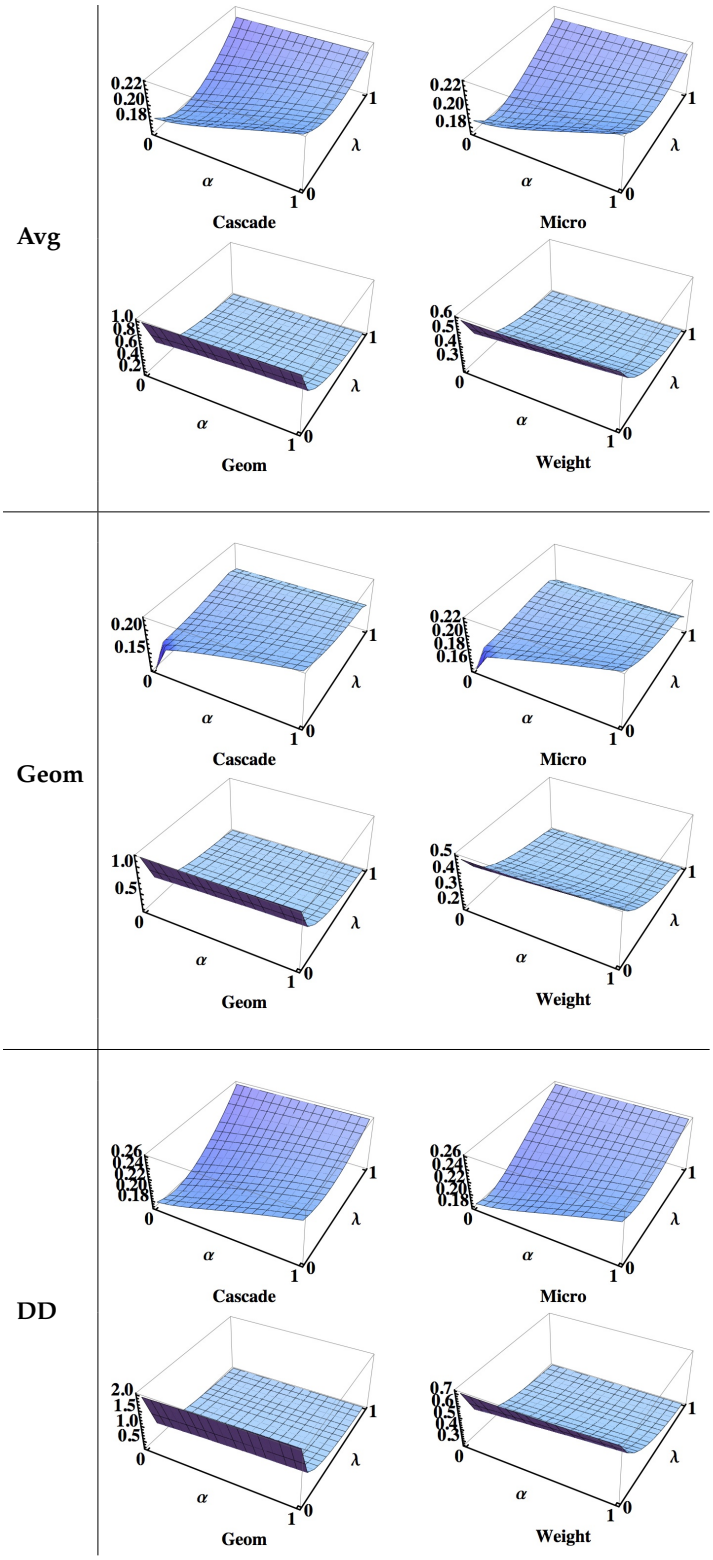


Figure A.13: Document selection sensitivity at rank 5 as a function of α and λ using ERR discounting.

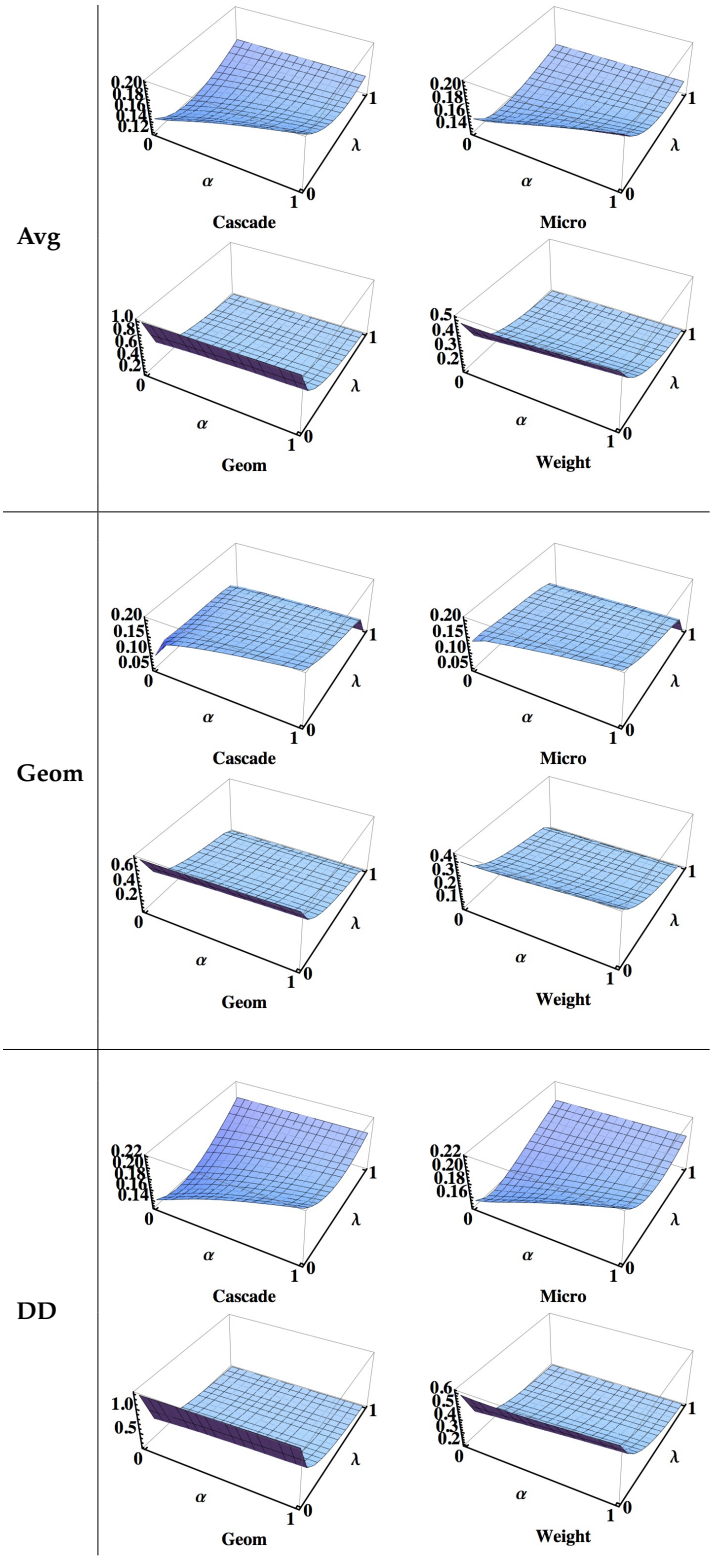


Figure A.14: Document selection sensitivity at rank 10 as a function of α and λ using ERR discounting.

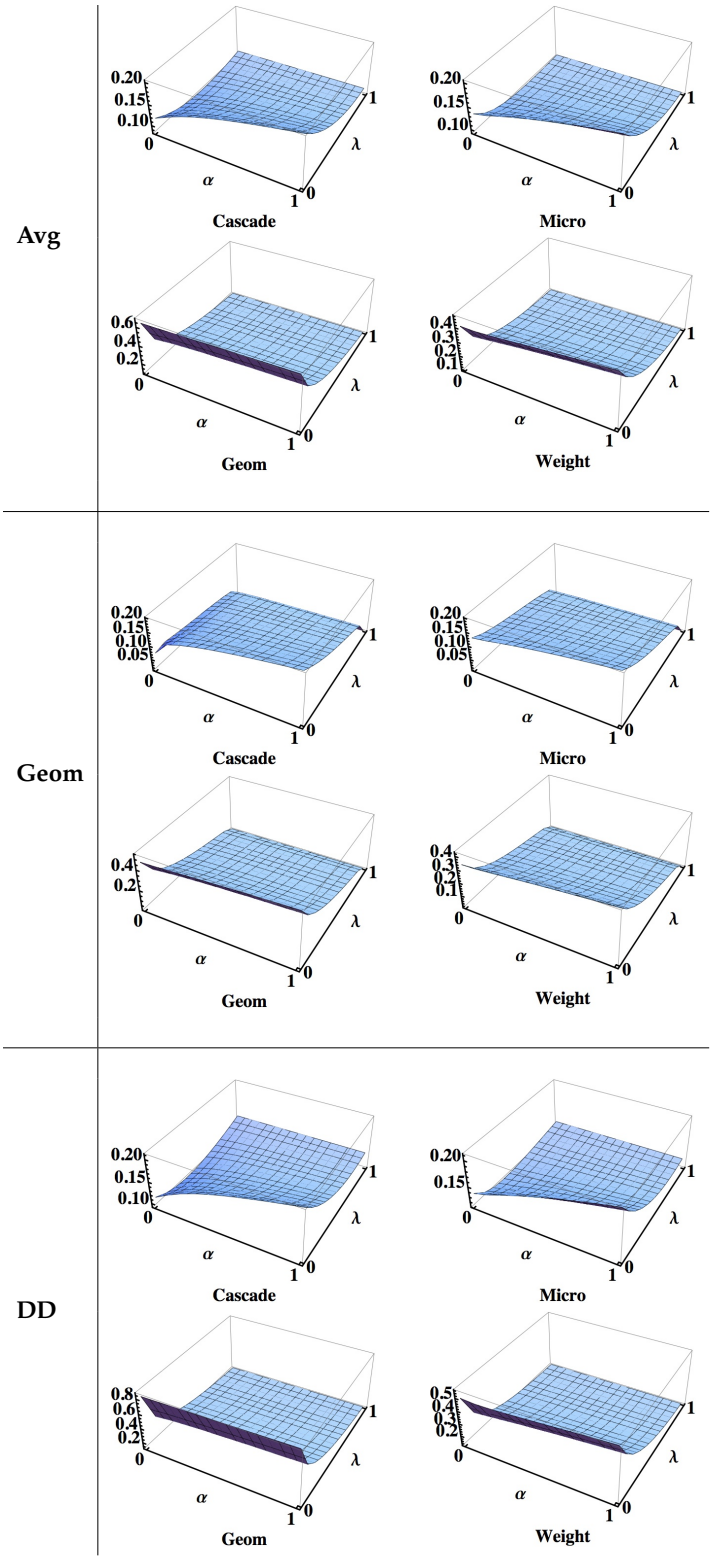


Figure A.15: Document selection sensitivity at rank 20 as a function of α and λ using ERR discounting.

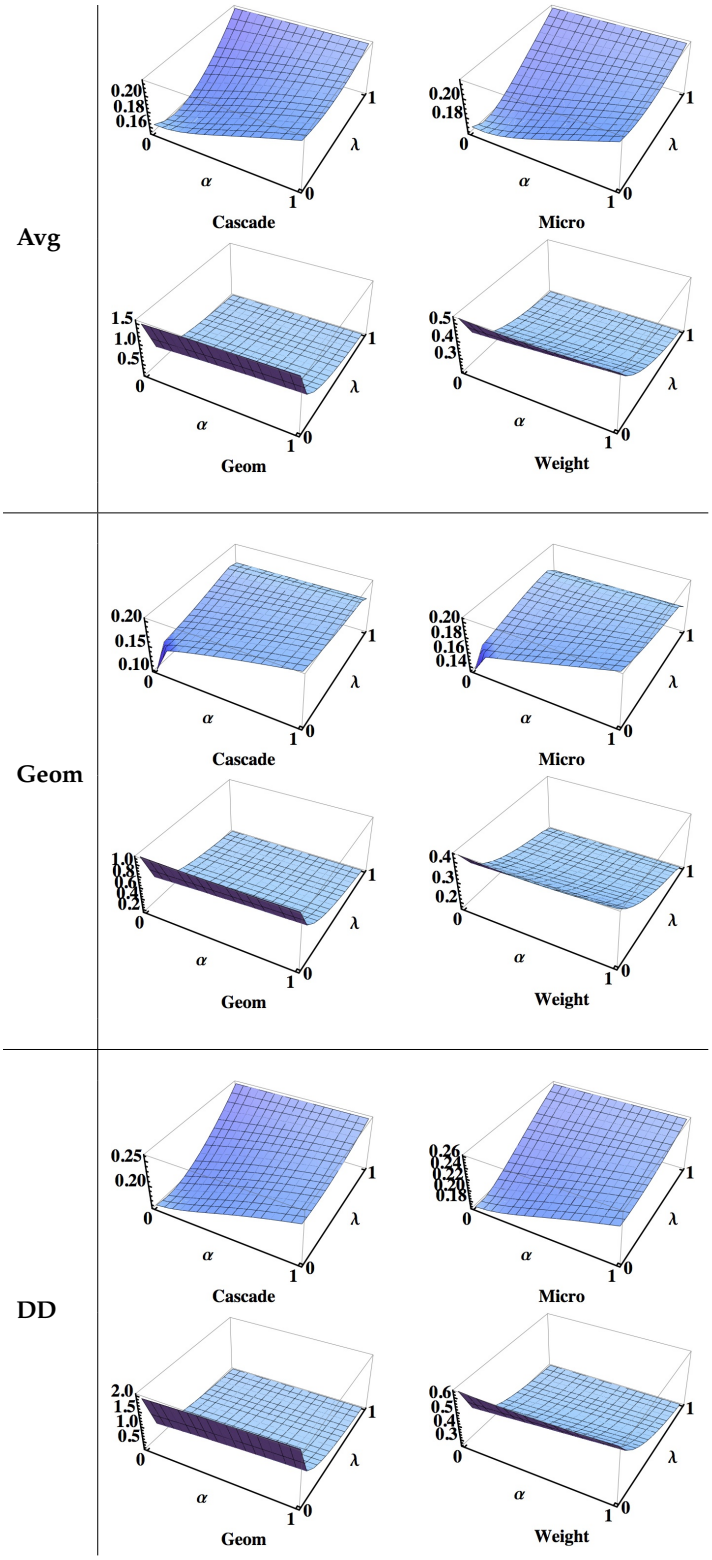


Figure A.16: Document selection sensitivity at rank 5 as a function of α and λ using DCG discounting.

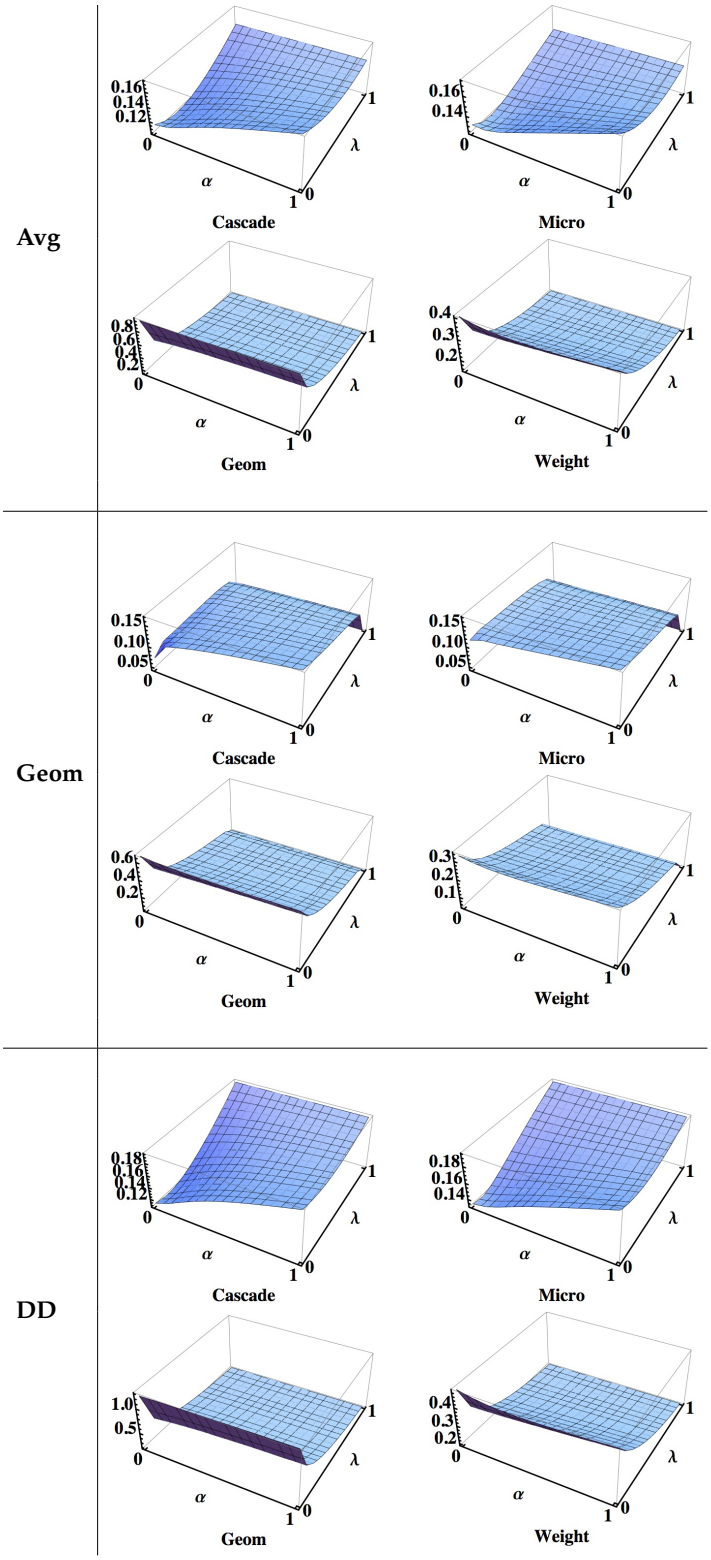


Figure A.17: Document selection sensitivity at rank 10 as a function of α and λ using DCG discounting.

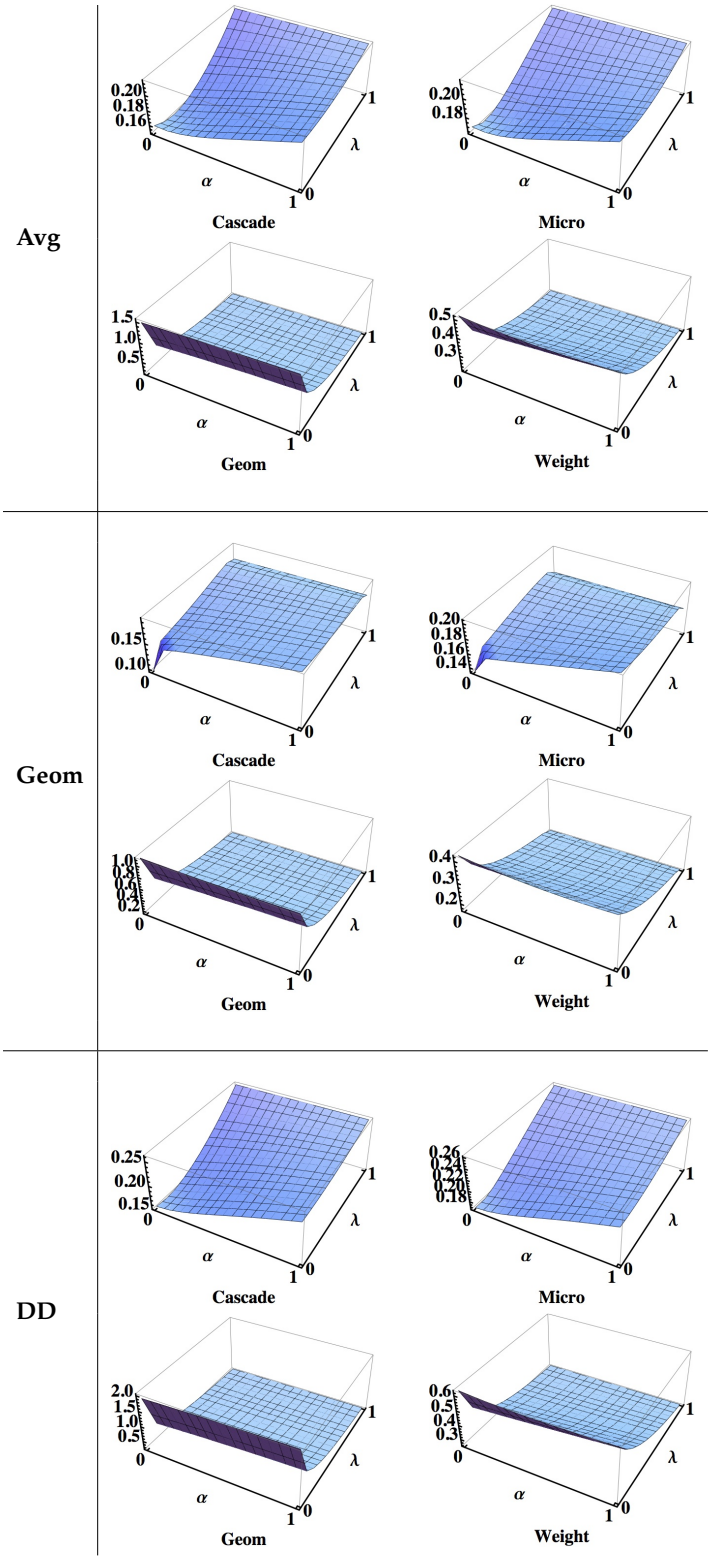


Figure A.18: Document selection sensitivity at rank 5 as a function of α and λ using RBP discounting.

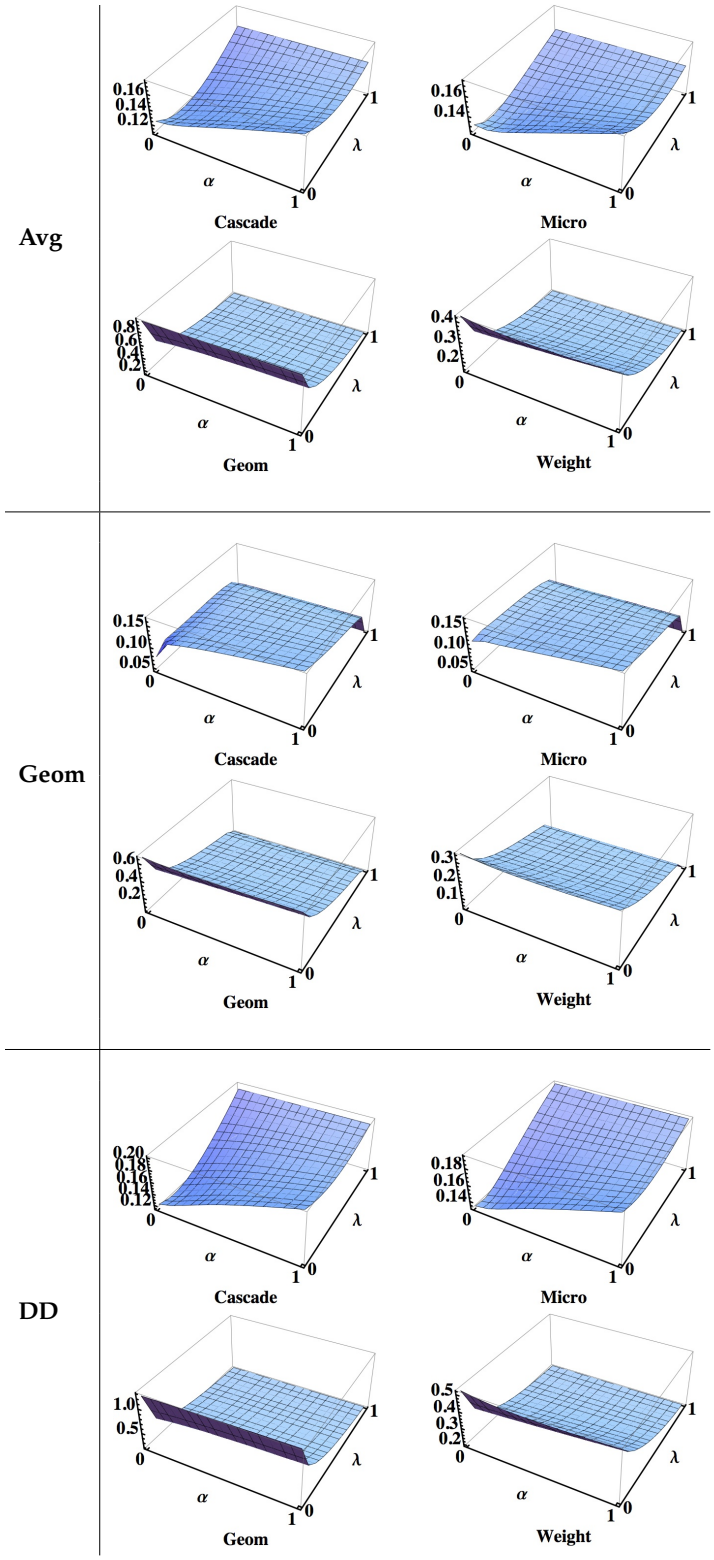


Figure A.19: Document selection sensitivity at rank 10 as a function of α and λ using RBP discounting.

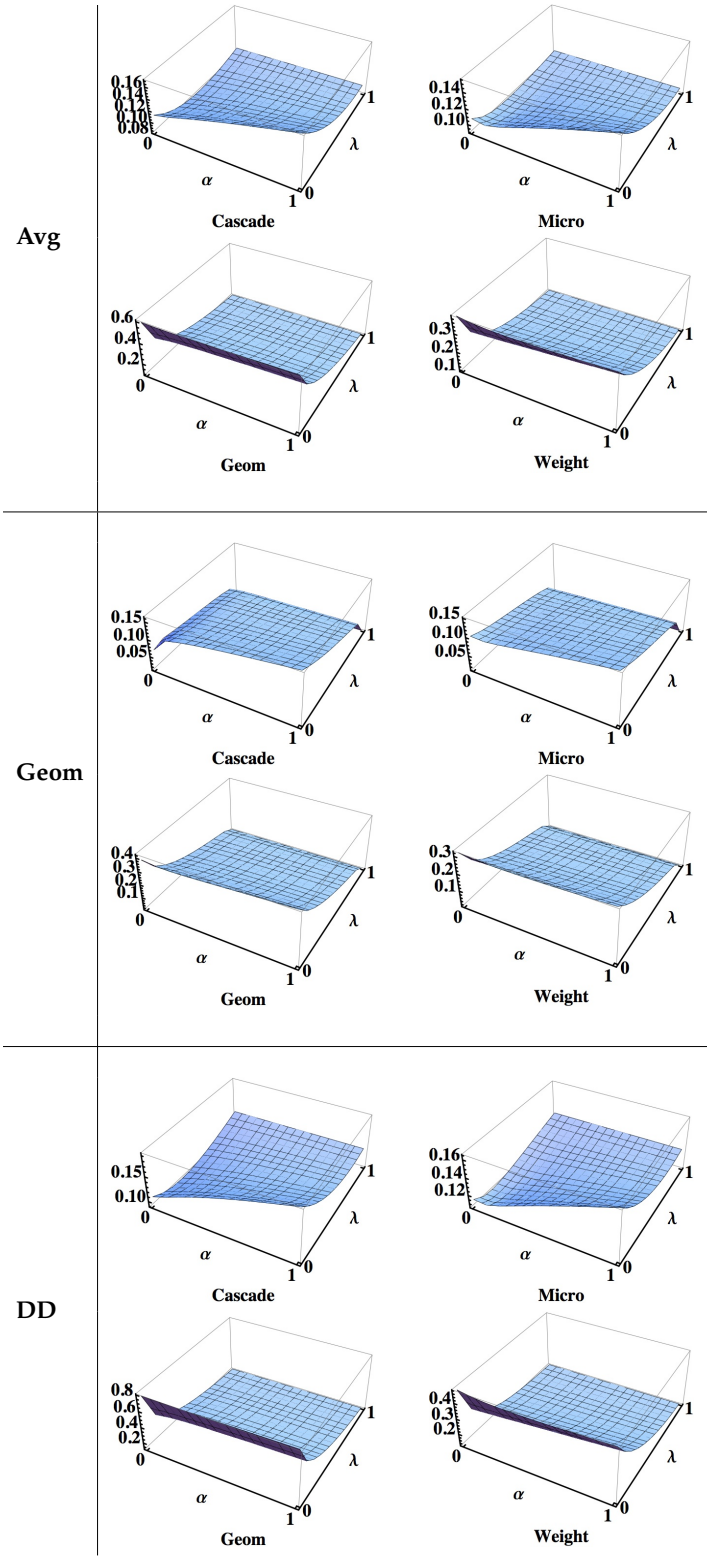


Figure A.20: Document selection sensitivity at rank 20 as a function of α and λ using RBP discounting.

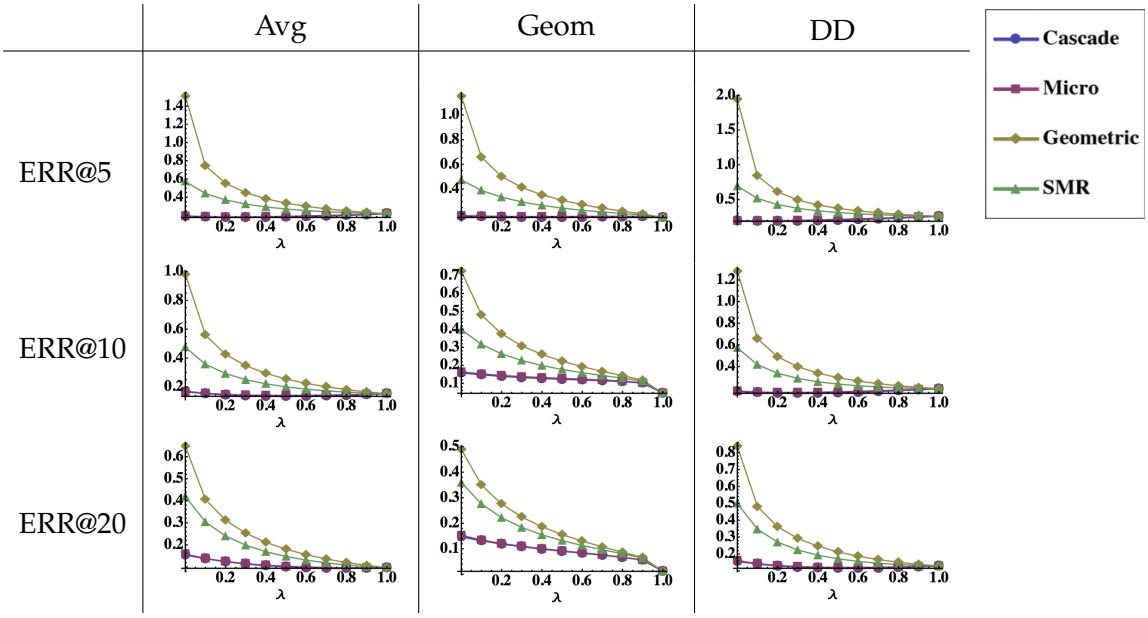


Figure A.21: Document Selection Sensitivity as a function of λ with ERR discounting. α is fixed at 0.3.

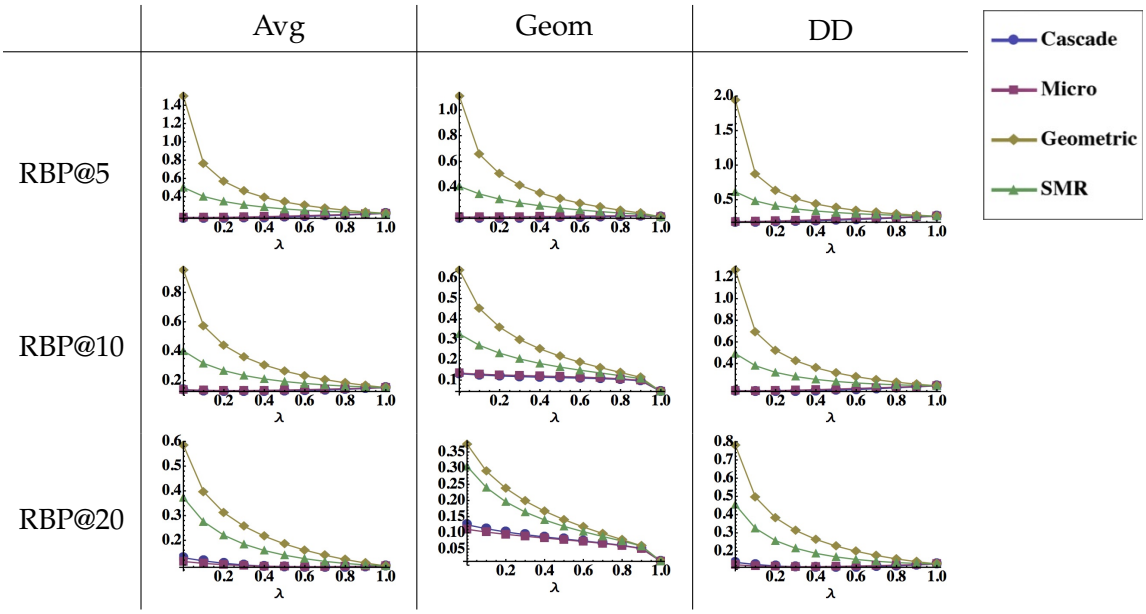


Figure A.22: Document Selection Sensitivity as a function of λ with RBP discounting. α is fixed at 0.3.

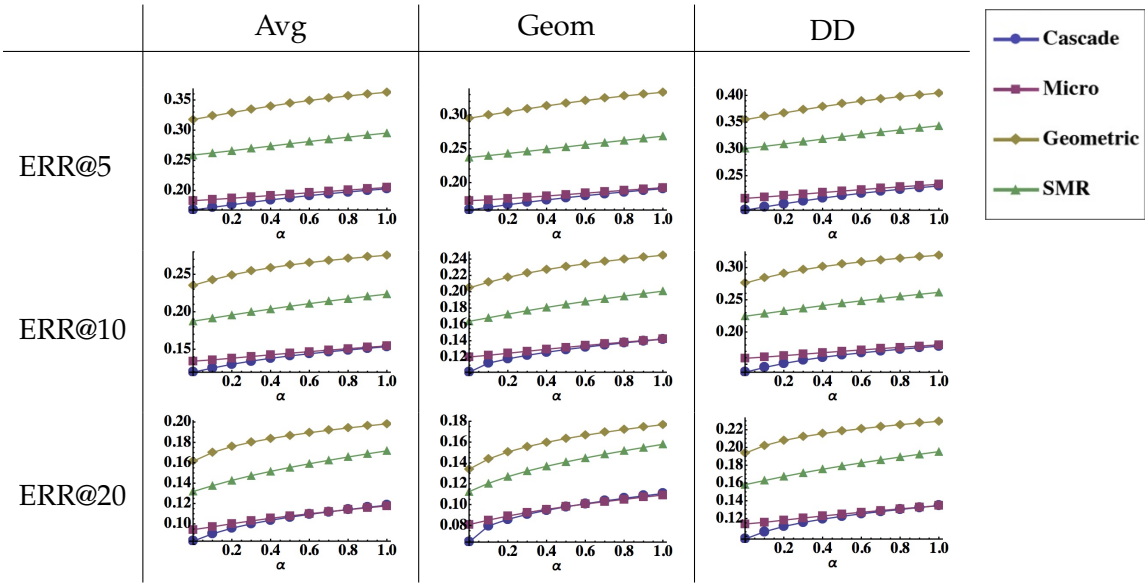


Figure A.23: Document Selection Sensitivity as a function of α with ERR discounting. λ is fixed at 0.5.

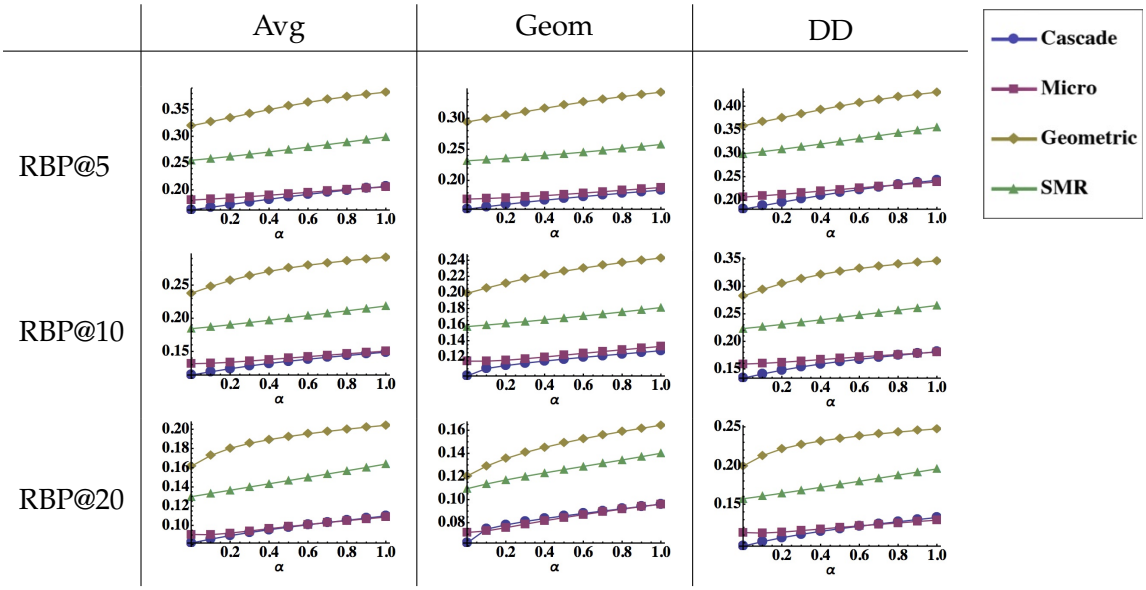


Figure A.24: Document Selection Sensitivity as a function of α with RBP discounting. λ is fixed at 0.5.

A.3 Information τ

In Section 5.1.3, we proved that the probabilistic formulation of Kendall's τ presented in Section 5.1.1 can be interpreted information-theoretically (Theorem 5.1). In Section 5.2, we defined this as *information* τ, which we showed can be used to define the rank correlation between two lists conditioned on a third. We demonstrated the utility of information τ by investigating the correlation between diversity measures. We found that, while diversity measures induce a highly correlated ranking of systems, most of this correlation disappears when the correlation due to the underlying performance measures is removed via conditioning.

We investigated three diversity measures: α-nDCG, D#-nDCG, and ERR-IA, and their underlying performance measures: nDCG and ERR. In Table 5.1, we showed the information τ between ERR-IA and both α-nDCG and D#-nDCG when averaged over all of the queries in the TREC 2010 and 2011 collections before and after the conditioning on the underlying performance measures. However, the variation due to topics in IR evaluation is generally quite large. Therefore, in Figure 5.2, we showed the impact of conditioning on a per-query basis to show that this effect holds for all queries. We presented the correlation between ERR-IA and D#-nDCG when conditioned on both ERR and nDCG, as these are the gold standard measures used at TREC and NTCIR and the results are typical. In this Section, we show the impact of conditioning between all combinations of measures shows in Table 5.1. See Figures A.25 through A.29. When visualized, the impact of conditioning is striking. The majority of queries are towards the right in each plot, showing that, prior to conditioning the rank correlation is high. Most queries have much larger values on the x-axis than on the y-axis, meaning that much of the correlation disappears when conditioned upon the underlying rank measure. Very few queries are above the diagonal, indicating that it is very rare for the correlation to be increased by conditioning.

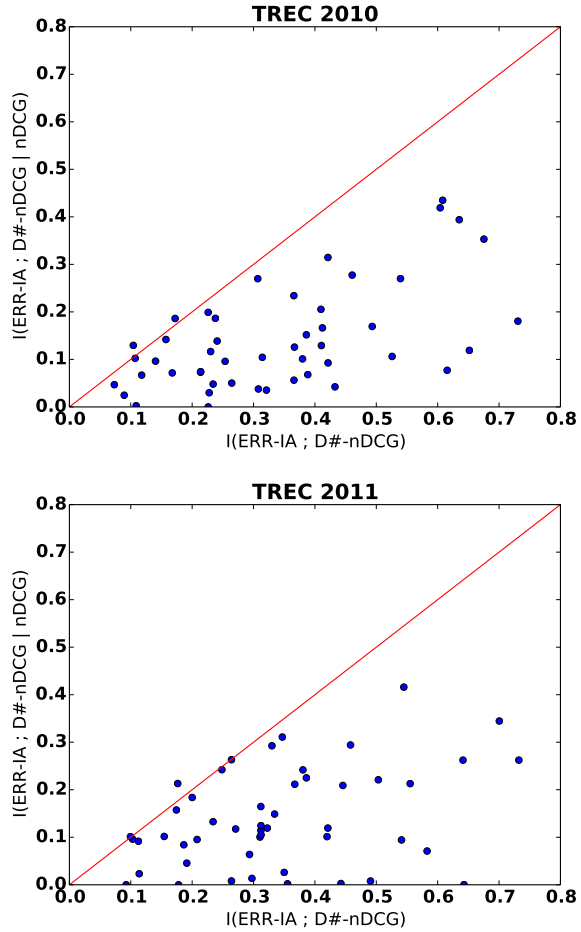


Figure A.25: Per-query information τ ERR-IA and D#-nDCG conditioned on nDCG.

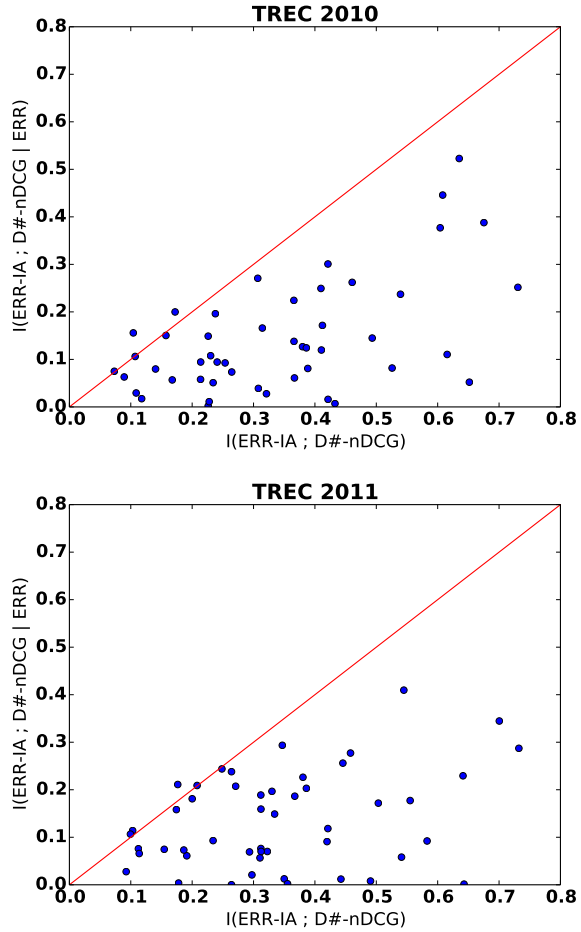


Figure A.26: Per-query information τ ERR-IA and D#-nDCG conditioned on ERR.

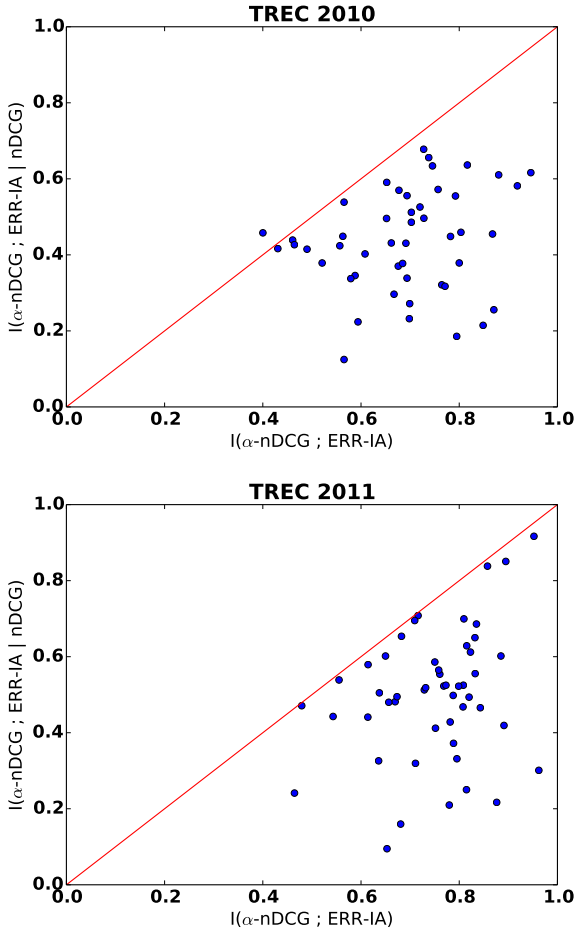


Figure A.27: Per-query information τ ERR-IA and α-nDCG conditioned on nDCG.

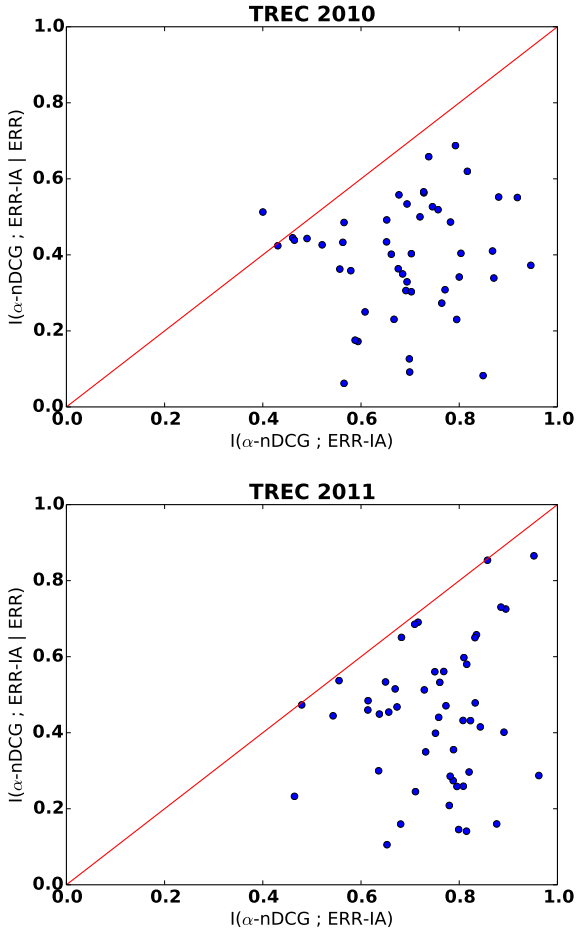


Figure A.28: Per-query information τ ERR-IA and α-nDCG conditioned on ERR.

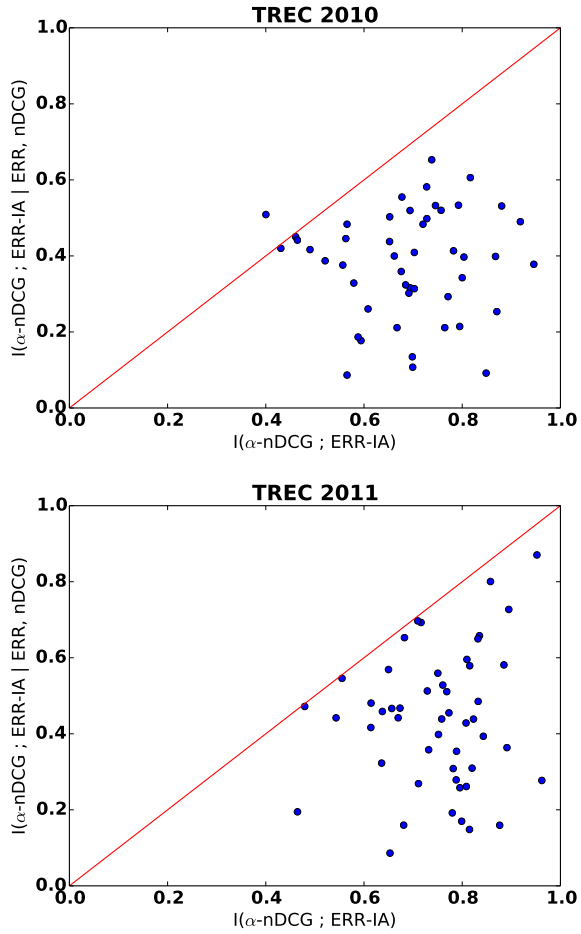


Figure A.29: Per-query information τ ERR-IA and α-nDCG conditioned on ERR and nDCG.

Bibliography

[1] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 5–14, New York, NY, USA, 2009. ACM.

[2] Gianni Amati. *Probability Models for Information Retrieval based on Divergence from Randomness*. PhD thesis, University of Glasgow, 2003.

[3] Einat Amitay, David Carmel, Ronny Lempel, and Aya Soffer. Scaling ir-system evaluation using term relevance sets. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 10–17, New York, NY, USA, 2004. ACM.

[4] Jaime Arguello, Fernando Diaz, Jamie Callan, and Ben Carterette. A methodology for evaluating aggregated search results. In *Proceedings of the 33rd European conference on Advances in information retrieval*, ECIR'11, pages 141–152, Berlin, Heidelberg, 2011. Springer-Verlag.

[5] Azin Ashkan and Charles L.A. Clarke. On the informativeness of cascade and intent-aware effectiveness measures. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pages 407–416, New York, NY, USA, 2011. ACM.

[6] Javed A. Aslam and Virgil Pavlu. Query hardness estimation using jensen-shannon divergence among multiple scoring functions. In *Proceedings of the 29th European conference on IR research*, ECIR'07, pages 198–209, Berlin, Heidelberg, 2007. Springer-Verlag.

[7] Javed A. Aslam and Virgil Pavlu. A practical sampling strategy for efficient retrieval evaluation. Technical report, College of Computer and Information Science, Northeastern University, Boston, Massachusetts, 2008.

[8] Javed A. Aslam, Virgil Pavlu, and Emine Yilmaz. A statistical method for system evaluation using incomplete judgments. In Susan Dumais, Efthimis N. Efthimiadis, David Hawking, and Kalervo Jarvelin, editors, *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 541–548. ACM Press, August 2006.

[9] Javed A. Aslam and Robert Savell. On the effectiveness of evaluating retrieval systems in the absence of relevance judgments. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, New York, NY, USA, 2003. ACM.

[10] Javed A. Aslam, Emine Yilmaz, and Virgiliu Pavlu. The maximum entropy method for analyzing retrieval measures. In Gary Marchionini, Alistair Moffat, John Tait, Ricardo Baeza-Yates, and Novio Ziviani, editors, *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 27–34. ACM Press, August 2005.

[11] David Banks, Paul Over, and Nien-Fan Zhang. Blind men and elephants: Six approaches to trec data. *Information Retrieval*, 1(1-2):7–34, 1999.

[12] Andrei Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, September 2002.

[13] Chris Buckley and Ellen M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, pages 33–40, New York, NY, USA, 2000. ACM.

[14] Chris Buckley and Ellen M. Voorhees. Retrieval evaluation with incomplete information. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 25–32, New York, NY, USA, 2004. ACM.

[15] Stefan Büttcher, Charles L. A. Clarke, Peter C. K. Yeung, and Ian Soboroff. Reliable information retrieval evaluation with incomplete and biased judgements. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 63–70, New York, NY, USA, 2007. ACM.

[16] Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of*

the 21st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '98, pages 335–336, New York, NY, USA, 1998. ACM.

[17] D. Carmel and E. Yom-Tov. *Estimating the Query Difficulty for Information Retrieval*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool, 2010.

[18] David Carmel, Elad Yom-Tov, Adam Darlow, and Dan Pelleg. What makes a query difficult? In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 390–397, New York, NY, USA, 2006. ACM.

[19] Ben Carterette. An analysis of NP-completeness in novelty and diversity ranking. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '09, pages 200–211, Berlin, Heidelberg, 2009. Springer-Verlag.

[20] Ben Carterette. System effectiveness, user models, and user utility: A conceptual framework for investigation. In *SIGIR '11*, New York, NY, USA, 2011. ACM.

[21] Ben Carterette, James Allan, and Ramesh Sitaraman. Minimal test collections for retrieval evaluation. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 268–275, New York, NY, USA, 2006. ACM.

[22] Ben Carterette, Virgil Pavlu, Evangelos Kanoulas, Javed A. Aslam, and James Allan. Evaluation over thousands of queries. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 651–658, New York, NY, USA, 2008. ACM.

[23] Praveen Chandar and Ben Carterette. Using preference judgments for novel document retrieval. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 861–870, New York, NY, USA, 2012. ACM.

[24] Praveen Chandar and Ben Carterette. Preference based evaluation measures for novelty and diversity. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 413–422, New York, NY, USA, 2013. ACM.

[25] Olivier Chapelle, Donald Metlzer, Ya Zhang, and Pierre Grinspan. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 621–630, New York, NY, USA, 2009. ACM.

[26] Harr Chen and David R. Karger. Less is more: probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 429–436, New York, NY, USA, 2006. ACM.

[27] Charles L. Clarke, Maheedhar Kolla, and Olga Vechtomova. An effectiveness measure for ambiguous and underspecified queries. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '09, pages 188–199, Berlin, Heidelberg, 2009. Springer-Verlag.

[28] Charles L. A. Clarke, Nick Craswell, and Ian Soboroff. Overview of the TREC 2009 Web Track. In *18th Text REtrieval Conference*, Gaithersburg, Maryland, 2009.

[29] Charles L. A. Clarke, Nick Craswell, Ian Soboroff, and Gordon V. Cormack. Overview of the TREC 2010 Web Track. In *19th Text REtrieval Conference*, Gaithersburg, Maryland, 2010.

[30] Charles L. A. Clarke, Nick Craswell, Ian Soboroff, and Ellen M. Voorhees. Overview of the TREC 2011 Web Track. In *20th Text REtrieval Conference*, Gaithersburg, Maryland, 2011.

[31] Charles L. A. Clarke, Nick Craswell, and Ellen M. Voorhees. Overview of the TREC 2012 Web Track. In *20th Text REtrieval Conference*, Gaithersburg, Maryland, 2012.

[32] Charles L.A. Clarke, Nick Craswell, Ian Soboroff, and Azin Ashkan. A comparative analysis of cascade measures for novelty and diversity. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 75–84, New York, NY, USA, 2011. ACM.

[33] Charles L.A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 659–666, New York, NY, USA, 2008. ACM.

[34] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 87–94, New York, NY, USA, 2008. ACM.

[35] Bruce Croft, Donald Metzler, and Trevor Strohman. *Search Engines: Information Retrieval in Practice*. Addison-Wesley Publishing Company, USA, 1st edition, 2009.

[36] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '02, pages 299–306, New York, NY, USA, 2002. ACM.

[37] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. Precision prediction based on ranked list coherence. *Inf. Retr.*, 9(6), December 2006.

[38] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, NY, 1993.

[39] Peter B. Golbus and Javed A. Aslam. A mutual information-based framework for the analysis of information retrieval systems. In *SIGIR '13*, New York, NY, USA, 2013. ACM.

[40] Peter B. Golbus, Javed A. Aslam, and Charles L.A. Clarke. Increasing evaluation sensitivity to diversity. *Information Retrieval*, 16(4), 2013.

[41] Peter B. Golbus, Virgil Pavlu, and Javed A. Aslam. What we talk about when we talk about diversity. In *Proceedings of Diversity in Document Retrieval 2012*, 2012.

[42] Teofilo F. Gonzalez. *Handbook of Approximation Algorithms and Metaheuristics (Chapman & Hall/Crc Computer & Information Science Series)*. Chapman & Hall/CRC, 2007.

[43] C. Hauff. *Predicting the Effectiveness of Queries and Retrieval Systems*. PhD thesis, University of Twente, Enschede, 2010.

[44] Ben He and Iadh Ounis. Inferring query performance using pre-retrieval predictors. In *In Proc. Symposium on String Processing and Information Retrieval*, pages 43–54. Springer Verlag, 2004.

[45] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, October 2002.

[46] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103, 1972.

[47] M. G. Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93, June 1938.

[48] Ravi Kumar and Sergei Vassilvitskii. Generalized distances between rankings. In *Proceedings of the 19th international conference on World wide web*, WWW '10, 2010.

[49] Yiyao Lu, Weiyi Meng, Liangcai Shu, Clement Yu, and King-Lup Liu. Evaluation of result merging strategies for metasearch engines. In AnneH.H. Ngu, Masaru Kitsuregawa, ErichJ. Neuhold, Jen-Yao Chung, and QuanZ. Sheng, editors, *Web Information Systems Engineering WISE 2005*, volume 3806 of *Lecture Notes in Computer Science*, pages 53–66. Springer Berlin Heidelberg, 2005.

[50] Alistair Moffat and Justin Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Trans. Inf. Syst.*, 27(1):2:1–2:27, December 2008.

[51] Mark Montague. *Metasearch: Data Fusion for Document Retrieval*. PhD thesis, Dartmouth College. Dept. of Computer Science, 2002.

[52] Mark Montague and Javed A. Aslam. Condorcet fusion for improved retrieval. In *Proceedings of the eleventh international conference on Informatio n and knowledge management*, CIKM '02, 2002.

[53] Josiane Mothe and Ludovic Tanguy. Linguistic features to predict query difficulty. In *In ACM SIGIR 2005 Workshop on Predicting Query Difficulty - Methods and Applications*, 2005.

[54] Michael J. Nelson. What are the important variables in the evaluation of information retrieval systems? In *Communication and Information in Context: Society, Technology, and the Professions*, 1997.

[55] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A high performance and scalable information retrieval platform. In *Proceedings of ACM SIGIR'06 Workshop on Open Source Information Retrieval (OSIR 2006)*, 2006.

[56] Virgil Pavlu, Shahzad Rajput, Peter B. Golbus, and Javed A. Aslam. Ir system evaluation using nugget-based test collections. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, pages 393–402. ACM Press, February 2012.

[57] Shahzad Rajput, Matthew Ekstrand-Abueg, Virgiliu Pavlu, and Javed A. Aslam. Constructing test collections by inferring document relevance via extracted relevant information. In *Proceedings of the 21st ACM Conference on Information and Knowledge Management*, pages 145–154. ACM Press, October 2012.

[58] S. E. Robertson. The probability ranking principle in IR. *Journal of Documentation*, 33(4):294–304, 1977.

[59] Stephen Robertson. On GMAP: and other transformations. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, CIKM '06, pages 78–83, New York, NY, USA, 2006. ACM.

[60] Tetsuya Sakai. Evaluating evaluation metrics based on the bootstrap. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, 2006.

[61] Tetsuya Sakai. Alternatives to Bpref. In *Proceedings of the 30th annual international ACM SIGIR conferenc e on Research and development in information retrieval*, SIGIR '07, 2007.

[62] Tetsuya Sakai. Evaluation with informational and navigational intents. In *Proceedings of the 21st World Wide Web Conference (WWW) 2012*, 2012.

[63] Tetsuya Sakai, Nick Craswell, Ruihua Song, Stephen Robertson, Zhicheng Dou, and Chin-Yew Lin. Simple evaluation metrics for diversified search results. In *The Third International Workshop on Evaluating Information Access (EVIA)*, 2010.

[64] Tetsuya Sakai and Ruihua Song. Evaluating diversified search results using per-intent graded relevance. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 1043–1052, New York, NY, USA, 2011. ACM.

[65] Gerard Salton. *Automatic Information Organization and Retrieval*. McGraw Hill Text, 1968.

[66] Mark Sanderson. Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375, 2010.

[67] Rodrygo L.T. Santos, Craig Macdonald, and Iadh Ounis. Intent-aware search result diversification. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 595–604, New York, NY, USA, 2011. ACM.

[68] RodrygoL.T. Santos, Craig Macdonald, and Iadh Ounis. On the role of novelty for search result diversification. *Information Retrieval*, 15:478–502, 2012.

[69] Markus Schulze. A new monotonic, clone-independent, reversal symmetric, and condorcet-consistent single-winner election method. *Social Choice and Welfare*, 36:267–303, 2011.

[70] Joseph A. Shaw and Edward A. Fox. Combination of multiple searches. In *The Second Text REtrieval Conference (TREC-2)*, pages 243–252, 1994.

[71] Anna Shtok, Oren Kurland, and David Carmel. Predicting query performance by query-drift estimation. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '09, pages 305–312, Berlin, Heidelberg, 2009. Springer-Verlag.

[72] Mark D. Smucker, Gabriella Kazai, and Matthew Lease. Overview of the TREC 2013 Crowdsourcing Track. In *21st Text REtrieval Conference*, Gaithersburg, Maryland, 2009.

[73] Mark D. Smucker, Gabriella Kazai, and Matthew Lease. Overview of the TREC 2013 Crowdsourcing Track. In *22th Text REtrieval Conference*, Gaithersburg, Maryland, 2009.

[74] Ian Soboroff. Dynamic test collections: Measuring search effectiveness on the live web. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 276–283, New York, NY, USA, 2006. ACM.

[75] Ian Soboroff, Charles Nicholas, and Patrick Cahan. Ranking retrieval systems without relevance judgments. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, New York, NY, USA, 2001. ACM.

[76] Ruihua Song, Min Zhang, Tetsuya Sakai, Makoto P. Kato, Yiqun Liu, Miho Sugimoto, Qinglei Wang, and Naoki Orii. Overview of the ntcir-9 intent task. In *Proceedings of the 9th NTCIR Workshop*, Tokyo, Japan, 2011.

[77] K SPARCK-JONES and CJ VANRIJSBERGEN. Information-retrieval test collections. *JOURNAL OF DOCUMENTATION*, 32(1):59–75, 1976.

[78] C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 1904.

[79] Jean Tague-sutcliffe and James Blustein. A statistical analysis of the trec-3 data. In *Overview of the Third Text REtrieval Conference (TREC-3*, pages 385–398, 1994.

[80] Vishwa Vinay, Ingemar J. Cox, Natasa Milic-Frayling, and Ken Wood. On ranking the effectiveness of searches. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 398–404, New York, NY, USA, 2006. ACM.

[81] Christopher C. Vogt. How much more is better? - characterizing the effects of adding more ir systems to a combination. In *In Content-Based Multimedia Information Access (RIAO*, 2000.

[82] Maksims N. Volkovs, Hugo Larochelle, and Richard S. Zemel. Learning to rank by aggregating expert preferences. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM '12, pages 843–851, New York, NY, USA, 2012. ACM.

[83] E. M. Voorhees and D. Harman. Overview of the eighth text retrieval conference (TREC-8). In *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, 2000.

[84] E. M. Voorhees and D. Harman. Overview of the ninth text retrieval conference (TREC-9). In *Proceedings of the Ninth Text REtrieval Conference (TREC-9)*, 2001.

[85] Ellen M. Voorhees and Chris Buckley. The effect of topic set size on retrieval experiment error. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 316–323, New York, NY, USA, 2002. ACM.

[86] William Webber, Alistair Moffat, and Justin Zobel. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst.*, 28(4), nov 2010.

[87] Emine Yilmaz and Javed A. Aslam. Estimating average precision with incomplete and imperfect judgments. In Philip S. Yu, Vassilis Tsotras, Edward Fox, and Bing Liu, editors, *Proceedings of the Fifteenth ACM International Conference*

on Information and Knowledge Management, pages 102–111. ACM Press, November 2006.

[88] Emine Yilmaz and Javed A. Aslam. Estimating average precision when judgments are incomplete. *Knowledge and Information Systems*, 16(2):173–211, August 2008.

[89] Emine Yilmaz, Javed A. Aslam, and Stephen Robertson. A new rank correlation coefficient for information retrieval. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, 2008.

[90] Emine Yilmaz, Evangelos Kanoulas, and Javed A. Aslam. A simple and efficient sampling method for estimating AP and NDCG. In Sung-Hyon Myaeng, Douglas W. Oard, Fabrizio Sebastiani, Tat-Seng Chua, and Mun-Kew Leong, editors, *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 603–610. ACM Press, July 2008.

[91] Elad Yom-Tov, Shai Fine, David Carmel, and Adam Darlow. Metasearch and federation using query difficulty prediction. In *In ACM SIGIR 2005 Workshop on Predicting Query Difficulty - Methods and Applications*, 2005.

[92] Budi Yuwono and Dik L. Lee. Server ranking for distributed text retrieval systems on the internet. In *In Proceedings of the Fifth International Conference on Database Systems for Advanced Applications*, 1997.

[93] Cheng Xiang Zhai, William W. Cohen, and John Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, pages 10–17, New York, NY, USA, 2003. ACM.

[94] Yuye Zhang, Laurence Park, and Alistair Moffat. Click-based evidence for decaying weight distributions in search eff ectiveness metrics. *Information Retrieval*, 13:46–69, 2010. 10.1007/s10791-009-9099-7.

[95] Yun Zhou and W. Bruce Croft. Ranking robustness: a novel framework to predict query performance. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 567–574, New York, NY, USA, 2006. ACM.

[96] Yun Zhou and W. Bruce Croft. Query performance prediction in web search environments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 543–550, New York, NY, USA, 2007. ACM.

[97] Justin Zobel. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 307–314, New York, NY, USA, 1998. ACM.

[98] Guido Zuccon, Leif Azzopardi, Dell Zhang, and Jun Wang. Top-k retrieval using facility location analysis. In *Proceedings of the 34th European conference on Advances in Information Retrieval*, ECIR'12, pages 305–316, Berlin, Heidelberg, 2012. Springer-Verlag.